
cisco Live!
ALL IN

*This session provides an overview of the programmability and automation features that are supported on the Cisco IOS XE Catalyst 9800 platforms.*

*An overview of the YANG based API's and the associated YANG Suite tooling will be used extensively throughout this session, in addition to gRPC and gNMI.*

*The Model Driven Telemetry capabilities will also be discussed and the example Docker container for collection and visualization will be demonstrated as well as example dashboards from Grafana for Client and AP visibility.*

*Let's not forget Guest Shell, EEM, the Python and NETCONF API, and other innovations around Zero Touch Provisioning that enable WLCs to be deployed, managed, and configured with ease at scale.*

The bridge to possible

# Programmability and Automation

## on Cisco Catalyst Wireless platforms

Jeremy Cohoe, Technical Marketing
@jeremycohoe

BRKEWN-2730

# Cisco Webex app

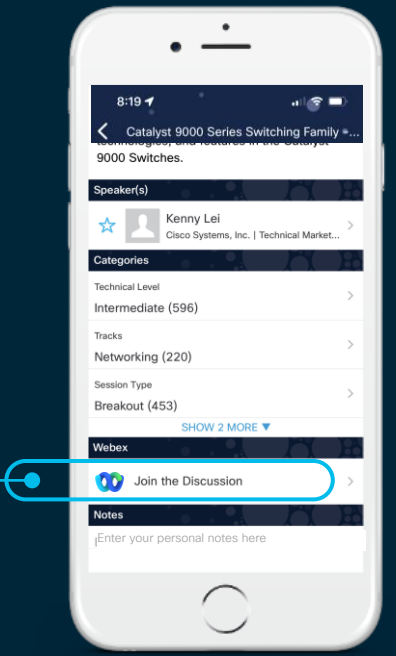
## Questions?

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/#BRKEWN-2730

# About Jeremy

WxT/jcohoe@cisco.com

FYI

- From Vancouver, BC, Canada
- Amateur Radio Operator, VA7NSA
- Canadian Forces Army – Signals Operator – 4 yrs
- UBC – Wireless Infrastructure – 7 yrs
- Cisco – Enterprise Networks - 5 yrs

# Agenda

- Introduction
- Day 1 WLC management with API's
  - YANG Suite, Terraform
- Day 2 Streaming Telemetry
  - TIG, gRPC Dial-Out & gNMI Dial-In
- Day N/0 Automation
  - Guest Shell, EEM, Python/NETCONF API
- Conclusion

GIF Demo
WLC in AWS

Live demos:
C9800-L ZTP

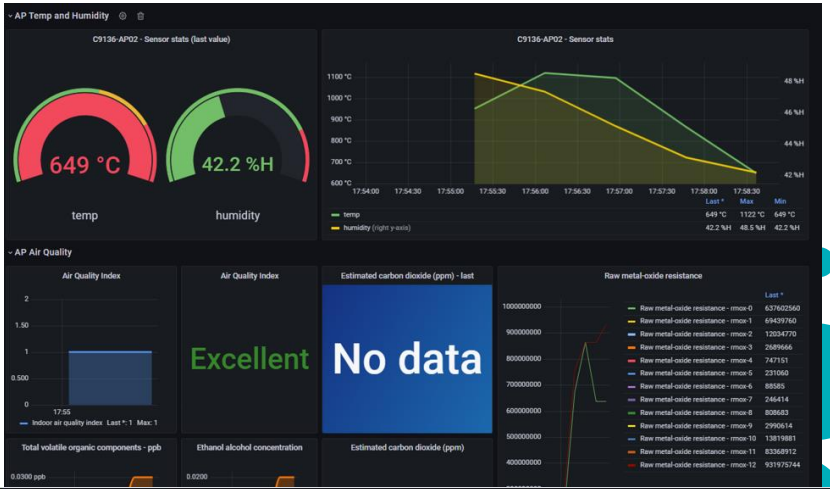
DEVWKS-2275
with Palmer
Supercharge
your Wireless
Programmability

# Demo workflow...

- Deploy C9800-CL WLC with Terraform in AWS
  - Using the default C9800-CL Amazon AMI (GCP, Azure, etc)
- Configure WLC with Terraform
  - Using RESTCONF + CLI2YANG payload
- WLC sends telemetry from AP to Grafana Dashboard
  - Visualized in Grafana

What is the current air quality in this breakout room ?

Let's plug in the AP9136 with built-in sensors and view the dashboard...



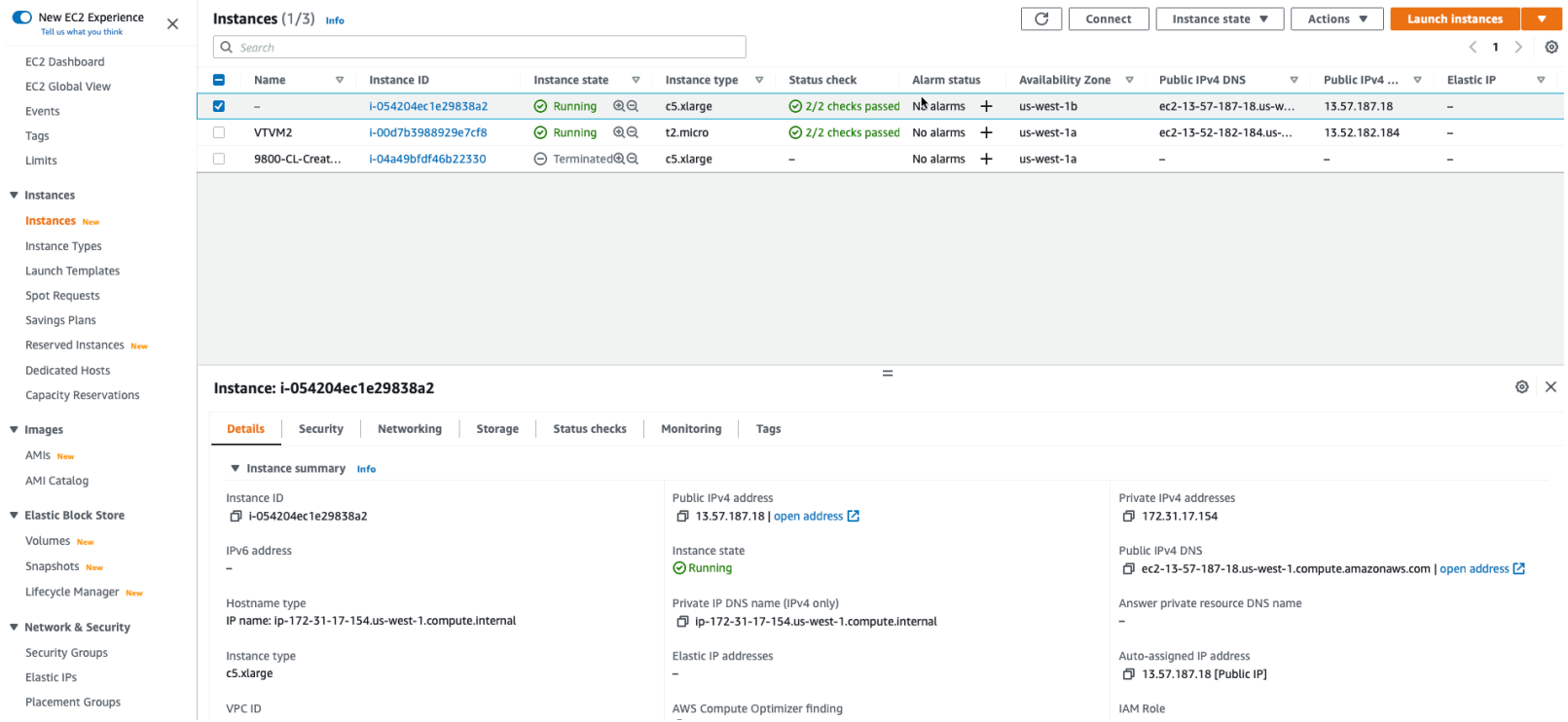


Demos

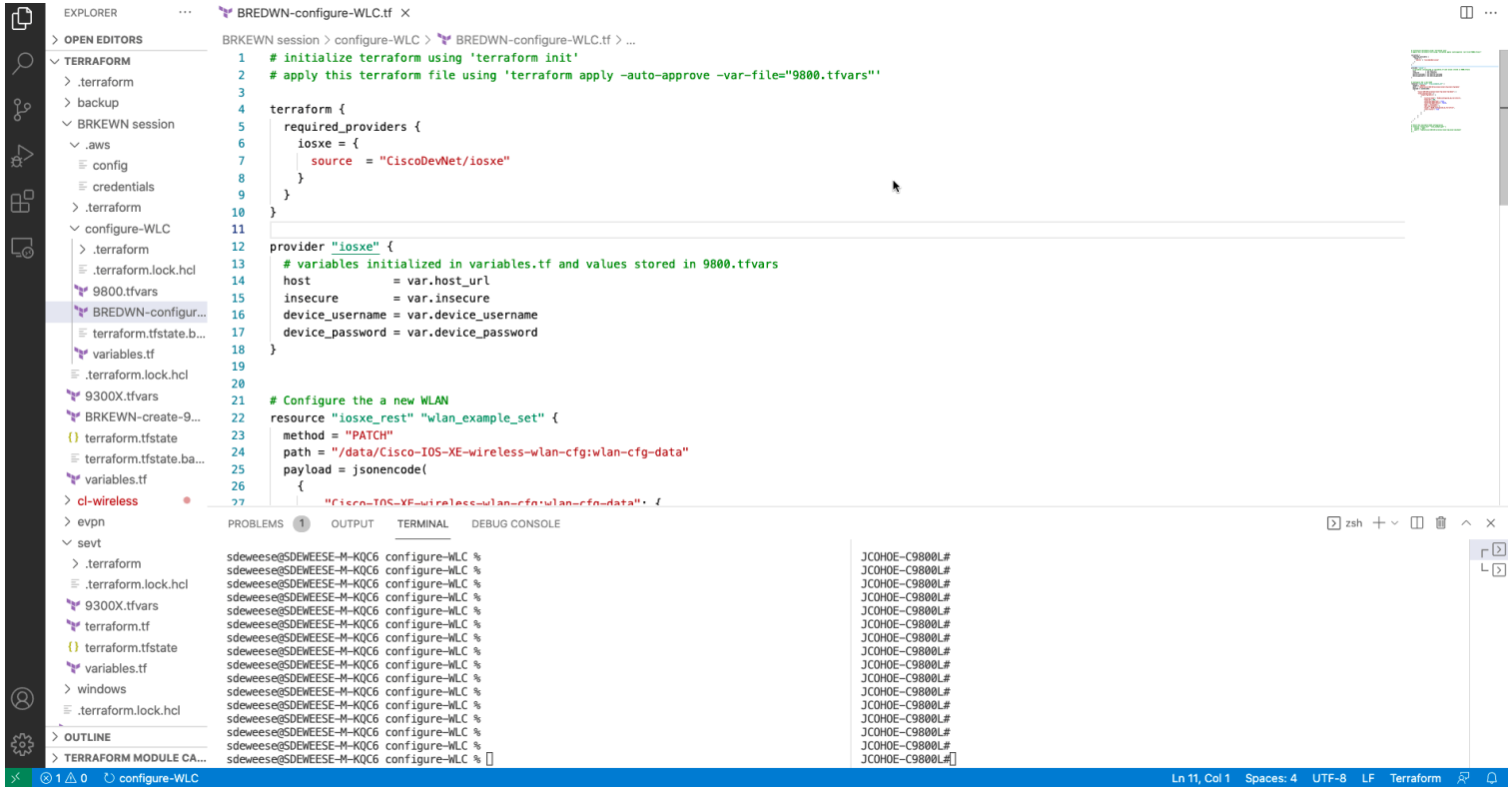
1. deploy controller w/ TF in cloud
2. Configure the new controller w/ TF
3. View cool dashboards (TIG stack)

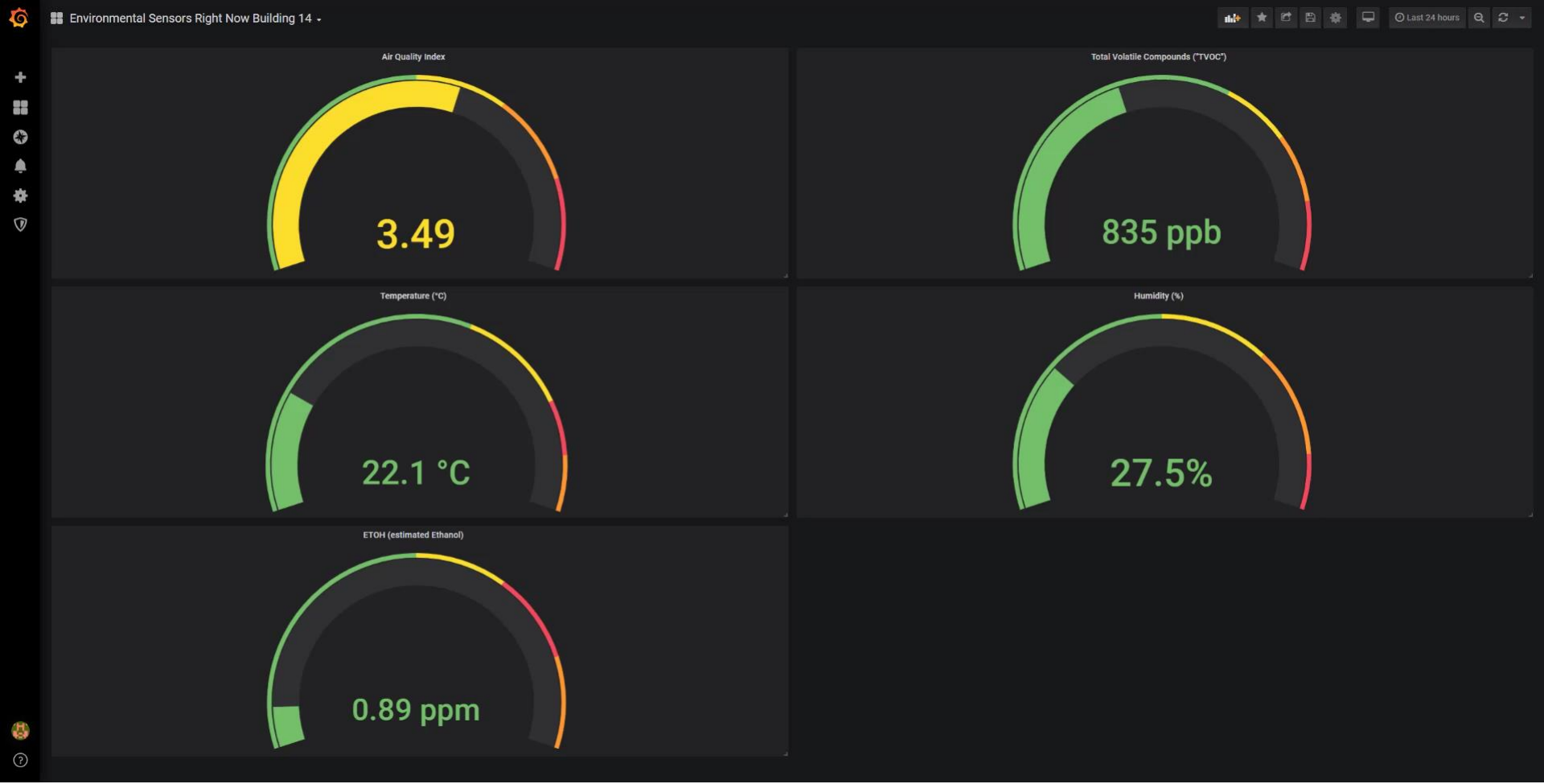
# Terraform Demo: Create a 9800 WLC in AWS

https://github.com/CiscoDevNet/terraform-provider-iosxe/tree/main/examples/tutorials/9800-and-AWS



# TF Demo – configure controller





# Cisco's Next-gen Wireless Stack

## Enabling next-generation mobility powered for Wi-Fi 6


Cisco Catalyst 9800
Wireless Controllers


Cisco Catalyst 9100
Access Points

Managed by

Cisco DNA Center

Translate business intent into network policy
and capture actionable insights

Digitized by

Cisco DNA Spaces

Digitize people, spaces and things

Resilient | Secure | Intelligent

# Cisco New Wi-Fi 6E Portfolio

MR and C series APs are not convertible

## One Product – Two personas

### CW9162

- 2x2 + 2x2 + 2x2
- 2.5 Gbps mGig
- Power Options: PoE, DC Power
- Scanning Radio
- IoT ready + Bluetooth 5.x
- Standard Bracket

### CW9164

- 2x2, 4x4, 4x4
- 2.5 Gbps mGig
- Power Options: PoE, DC Power
- Scanning Radio
- IoT Ready + Bluetooth 5.x
- Standard Bracket

### CW9166

- 4x4 + 4x4, 4x4 (XOR 5/6)
- 5 Gbps mGig
- Power Options: PoE, DC Power
- IoT ready + Bluetooth 5.x
- Scanning Radio
- Environmental Sensor
- Common XOR Architecture
- Standard Bracket

### MR57

- 4x4 + 4x4, 4x4 (XOR 5/6)
- Dual 5 Gbps mGig with failover
- Power Options: PoE, DC Power
- IoT ready + Bluetooth 5.x
- Scanning Radio
- XOR Architecture (High/Low band)
- Standard Bracket

### C9136

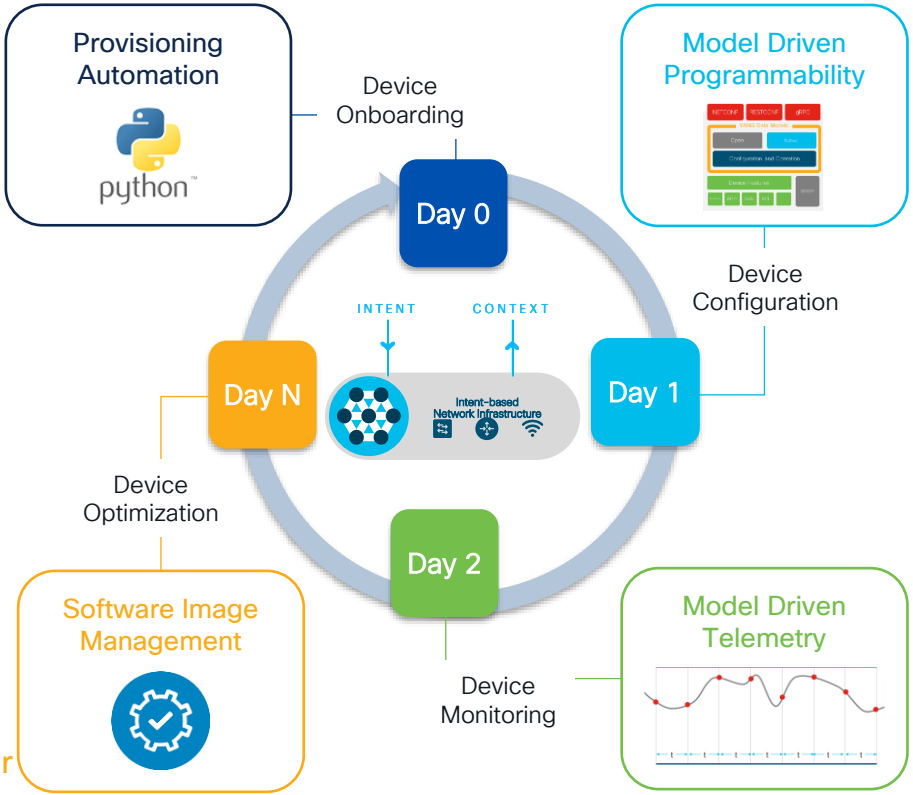
- 4x4 + 8x8 + 4x4 or 4x4+4x4+4x4+4x4
- Dual 5 Gbps mGig with failover
- Power Options: PoE, DC Power
- IoT ready + Bluetooth 5.x
- Scanning Radio
- Environmental Sensor
- XOR Architecture (macro/meso)
- Standard Bracket

# IOS XE Programmability & Automation Overview

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 with Python 3

Application Hosting with Docker

CLI to YANG

TIG_MDT container + GitHub examples

YANG On-Change support

gRPC Dial-Out + DNS + mTLS

gNMI/NETCONF Dial-In

Reference

# Programmable features on wireless platforms

| | EWC | C9800-CL | C9800-L | C9800-40/80 |
|---|---|---|---|---|
| ZTP / Guest Shell | N/A | N/A | ZTP | Yes |
| NETCONF | Yes | Yes | Yes | Yes |
| NETCON Dial-In MDT | Enabled | Enabled | Enabled | Enabled |
| RESTCONF | Yes | Yes | Yes | Yes |
| gNMI | N/A | Yes | Yes | Yes |
| gNMI Dial-In MDT | N/A | Enabled | Enabled | Enabled |
| gRPC Dial-Out MDT | N/A | Enabled | Enabled | Yes |
| gNOI cert.proto | N/A | Enabled | Enabled | Enabled |
| gNOI factory reset | N/A | N/A | N/A | Yes |

Source of truth: https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/prog/configuration/178/b_178_programmability_cg.html

*IOS XE Programmability provides a flexible and open API ecosystem that is available for integrations with 3rd party tooling and systems*

# Programmable Interfaces

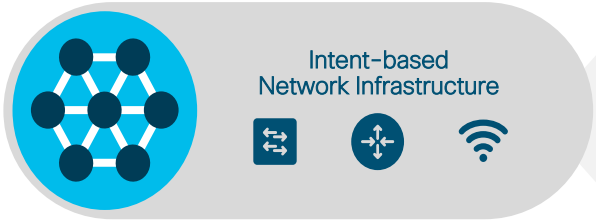
CLI

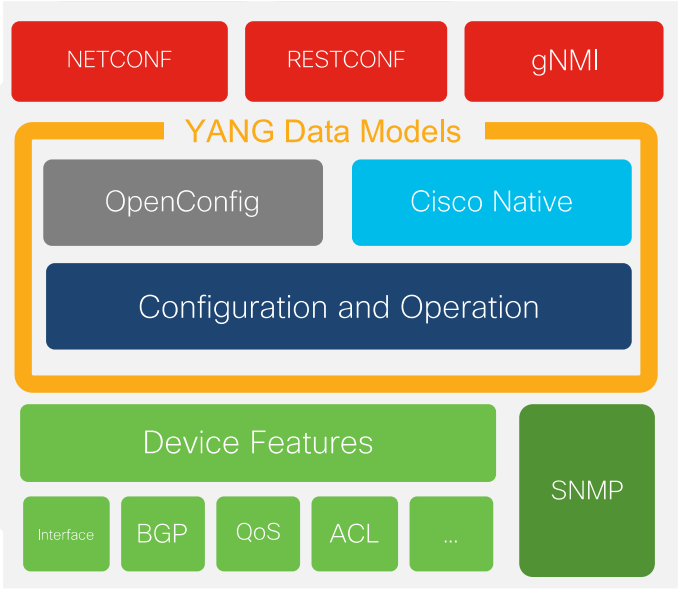
SNMP

WebUI

The NETCONF, RETCONF and gNMI are **programmatic** interfaces that provide **additional** methods for interfacing with the IOS XE device – Just like the CLI, SNMP, and WebUI is used for configuration changes and operational metrics so can the programmatic interfaces of NETCONF, RESTCONF and gNMI

YANG data models define the data that is available for configuration and streaming telemetry



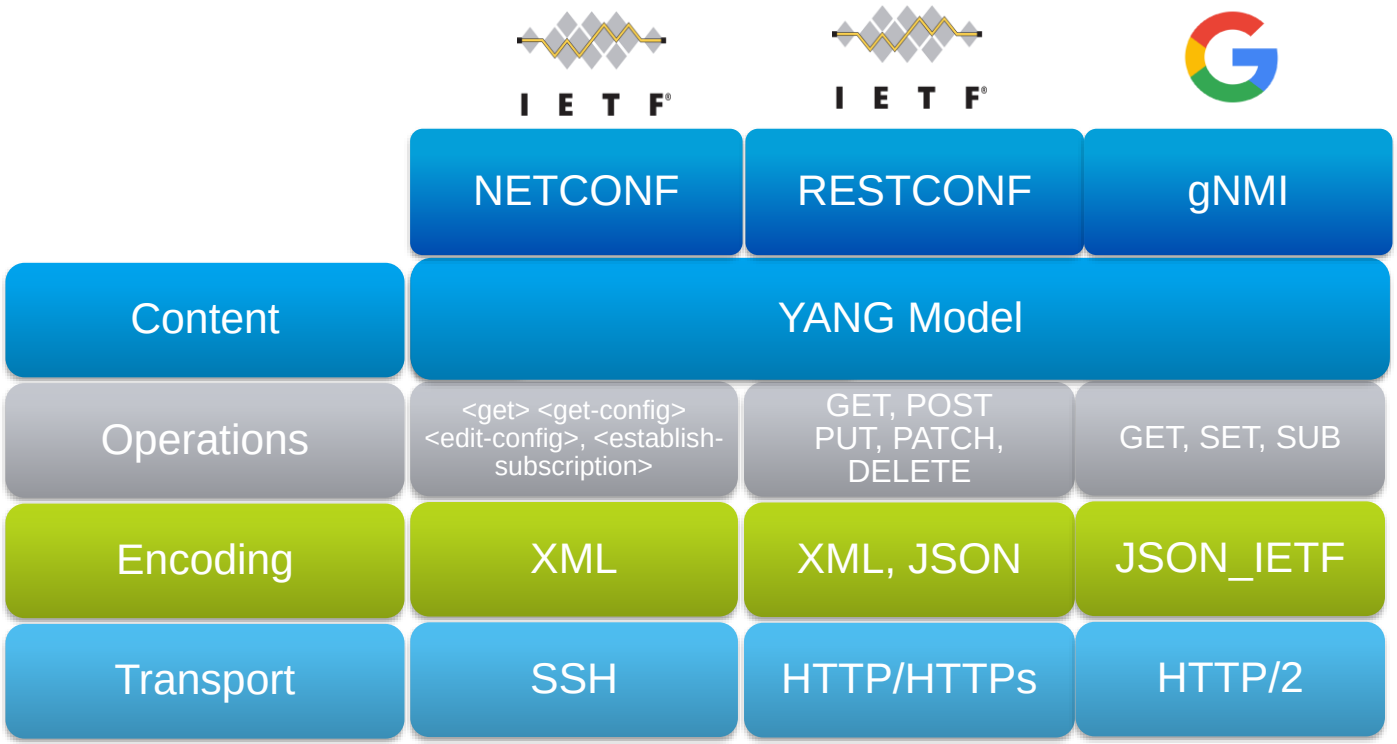

IETF

OPENCONFIG

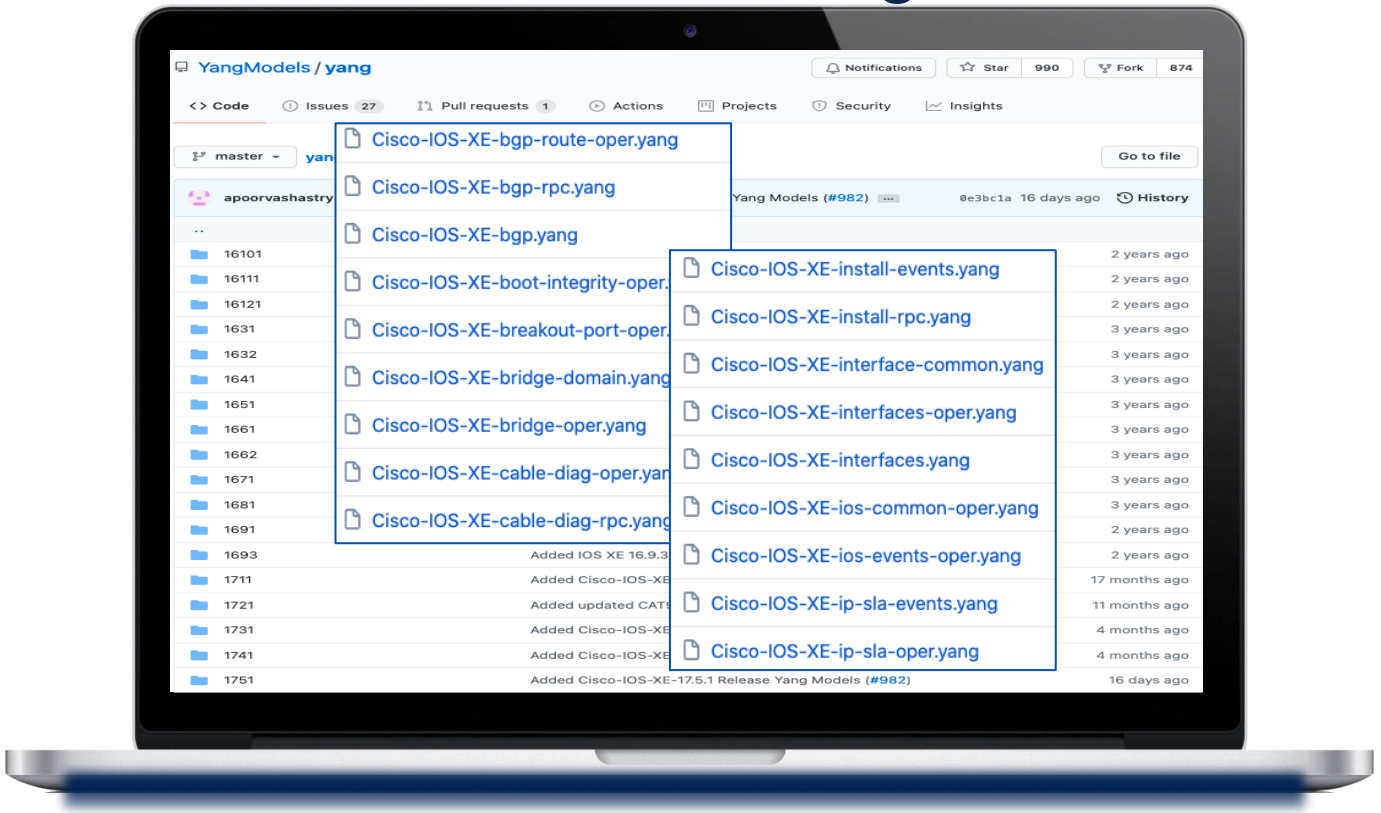
IEEE

CISCO

# API Interfaces



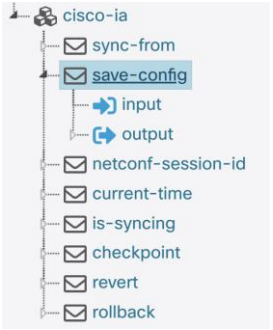
# IOS XE - YANG model coverage on GitHub

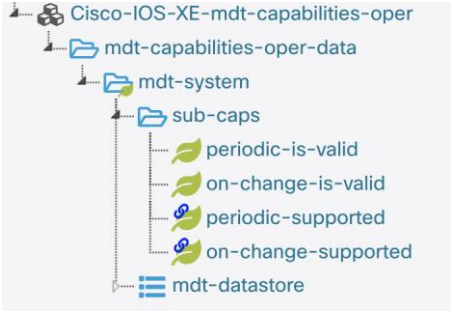



https://github.com/YangModels/yang/tree/master/vendor/cisco/xe

# Notable YANG models

| YANG Module | Description |
|---|---|
| Cisco-IOS-XE-native.YANG | Running-configuration |
| Cisco-IOS-XE-{feature}-oper.YANG | Feature specific operational data |
| Cisco-IOS-XE-{feature}-cfg.YANG | Feature specific configuration |
| Cisco-IOS-XE-RPC, cisco-ia | Actions for DHCP Renew, Save config |
| Cisco-IOS-XE-events-oper.yang | Event based telemetry notifications |
| Cisco-IOS-XE-mdt-capabilities-oper.yang | Telemetry capabilities |





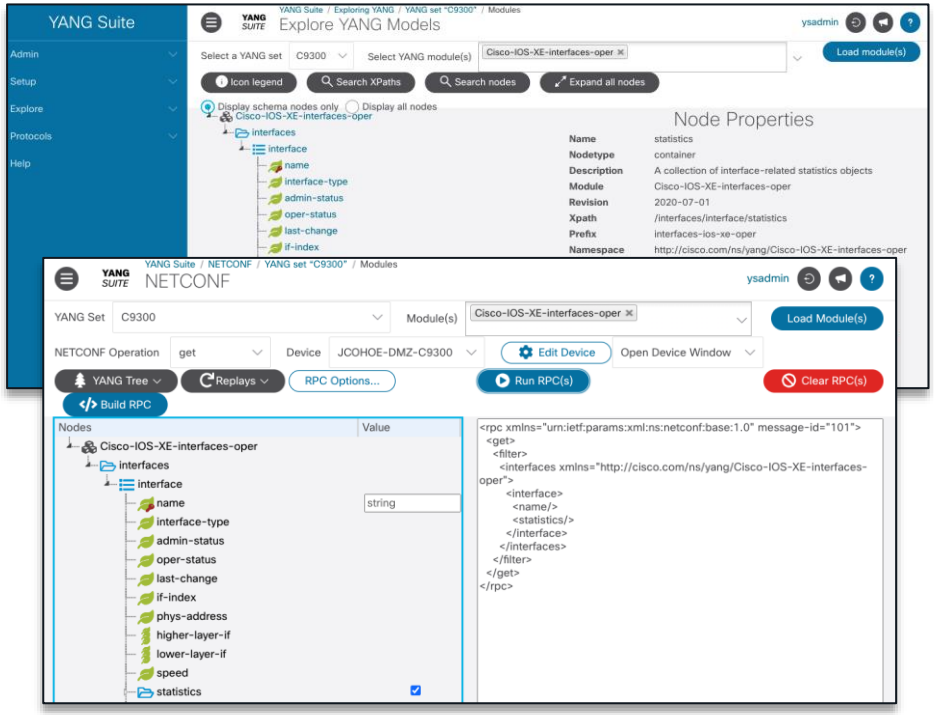
BLE YANG

# Cisco YANG Suite

YANG API Testing and Validation Environment

Construct and test YANG based APIs over NETCONF, RESTCONF, gRPC and gNMI

IOS XE / IOS XR / NX OS platforms




Now Generally Available !

DEVLIT-2787 with Story has more details about YANG Suite

developer.cisco.com/yangsuite

github.com/CiscoDevNet/yangsuite

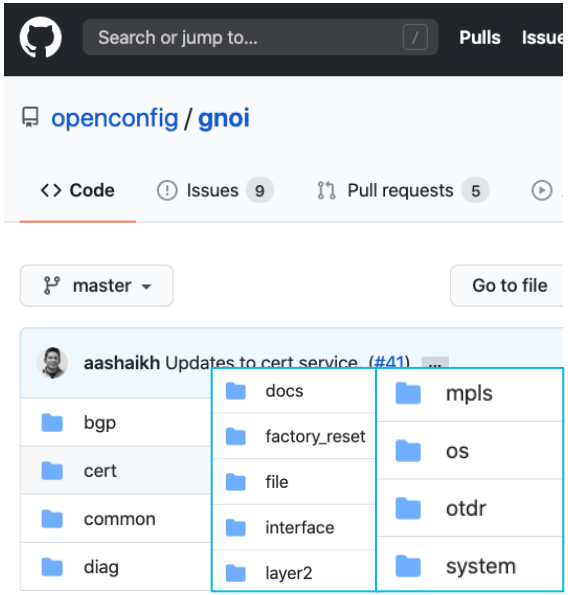
gNMI / gNOI

# gRPC Network Operations Interface

1. gRPC Network Operations Interface, or gNOI, is a set of gRPC-based microservices, used for executing operational commands on network devices
2. gNOI operations are executed against the gNMI API interface
3. gNOI is defined and implemented on a per proto basis
4. There are many protos defined - some are more mature and evolve and different pace

| Protobuf RPC | Use | Related CLI | Release |
|---|---|---|---|
| Cert.proto | TLS Certificate management | `crypto pki …` | 17.3 |
| Os.proto | Network Operating System management | `install add file …` | 17.5 |
| Reset.proto | Factory Reset and wipe | `factory-reset …` | 17.7 |
| File.proto | Not implemented | `copy, delete` | N/A |
| System.proto | Not implemented | `reload, set boot` | N/A |

https://github.com/openconfig/gnoi



# Model driven programmability comparison

Network architecture, security posture and policy, YANG data modules, tools and language preferences are some considerations when leveraging the various MDP interfaces

| | NETCONF | RESTCONF | gNMI |
|---|---|---|---|
| Minimum IOS XE Version | 16.6 | 16.7 | 16.8 |
| Recommended Version | 17.6 | 17.6 | 17.7 |
| Default Port | 830 | 443 | 9339 |
| Operations | <get>,<get-config>,<edit-config>,<establish-subscription> | GET, POST, PUT, PATCH, DELETE | GET, SET, SUBSCRIBE |
| Encoding | XML | XML or JSON | RFC7951 JSON_IETF |
| Security | SSH + PKI certificate or password | HTTPS user/pass | TLS certificate with user authentication |
| Transport Protocol | SSH | HTTPS | HTTP/2 |
| Telemetry Collector | Client | Server | Client |
| Tooling | YANG Suite, ncclient, Netconf-console | YANG Suite*, Postman, python | YANG Suite*, gnmic, gnmi_cli |
| Content | YANG | YANG | YANG + Protobuf |

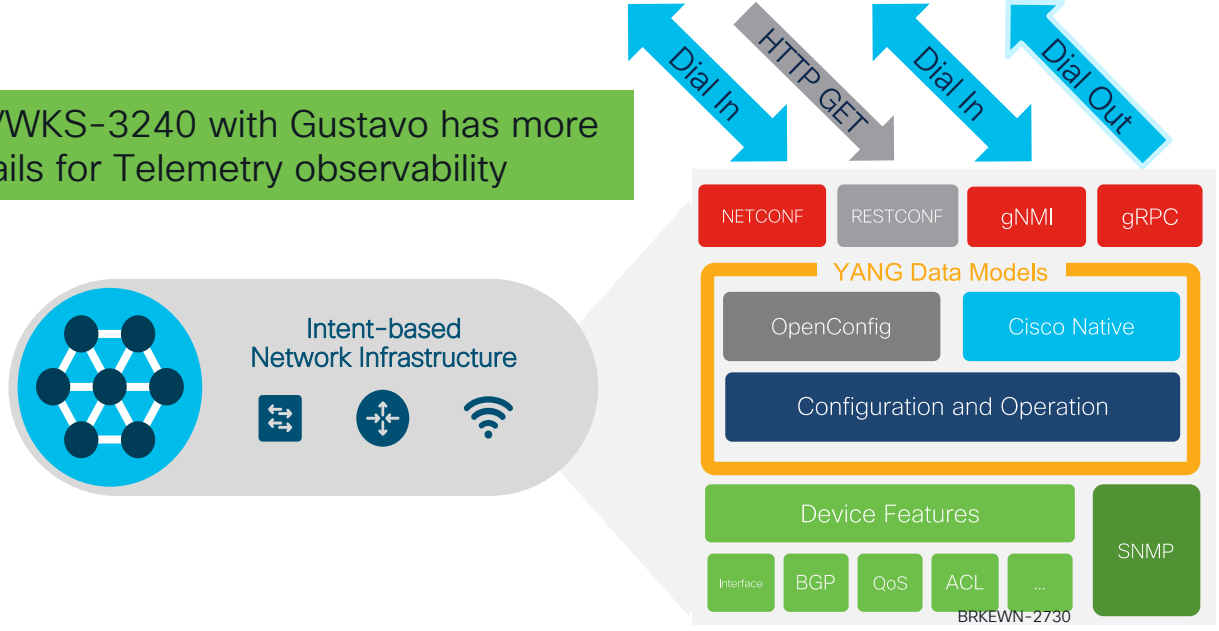
# Model Driven Telemetry Interfaces

Dial In: Collector establishes a connection to the device then subscribes to telemetry (pub/sub)

Dial Out: Telemetry is pushed from the device to the collector based off configuration (push)

## Publication / Subscription

DEVWKS-3240 with Gustavo has more details for Telemetry observability




XML, JSON and kvGPB encoding

Consistent YANG data models between interfaces

On-change event and time-based publication options

# Model Driven Telemetry Interface Comparison

| | NETCONF | gRPC | gNMI |
|---|---|---|---|
| Minimum IOS XE Version | 16.6 | 16.10 | 16.12 |
| Telemetry Direction | Dial-In,<br>IOS XE is server | Dial-Out<br>IOS XE is client | Dial-In<br>IOS XE is server |
| Configuration | Dynamic<br>per session | Static<br>per configuration | Dynamic<br>per session |
| Telemetry Collector | Client | Server | Client |
| Encoding | XML | KV GPB | JSON_IETF |
| Security | SSH + PKI<br>certificate or password | TLS or plain-text | TLS certificate<br>with user authentication |
| Transport Protocol | SSH | HTTP2 | HTTP2 |
| Data Models | YANG | YANG | YANG |

Network architecture, security posture and policy, YANG data modules, tools and language preferences are some considerations when leveraging the various MDT interfaces

DEVWKS-3270 with Jeremy has more details TIG telemetry collection

# IOS XE Model Driven Telemetry

CLI

...or with...

YANG

NSO

ANSIBLE

pyATS

gNMI Dial-In/Dynamic
NETCONF Dial-In

gRPC Dial-Out/Configured

**Collector/Receiver**
Decodes to text

splunk

telegraf

Cisco Crosswork
Network Automation

**Storage**
Time Series Database

elastic

InfluxDB

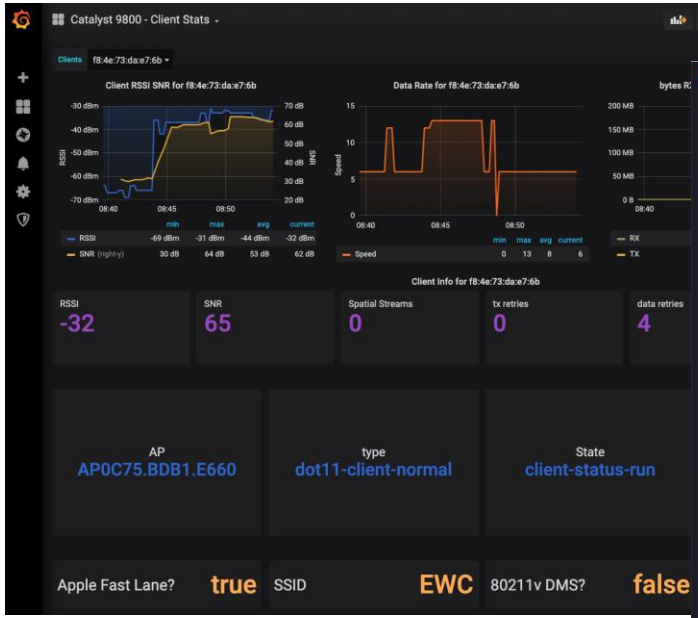
**Monitoring**
and Visualizations

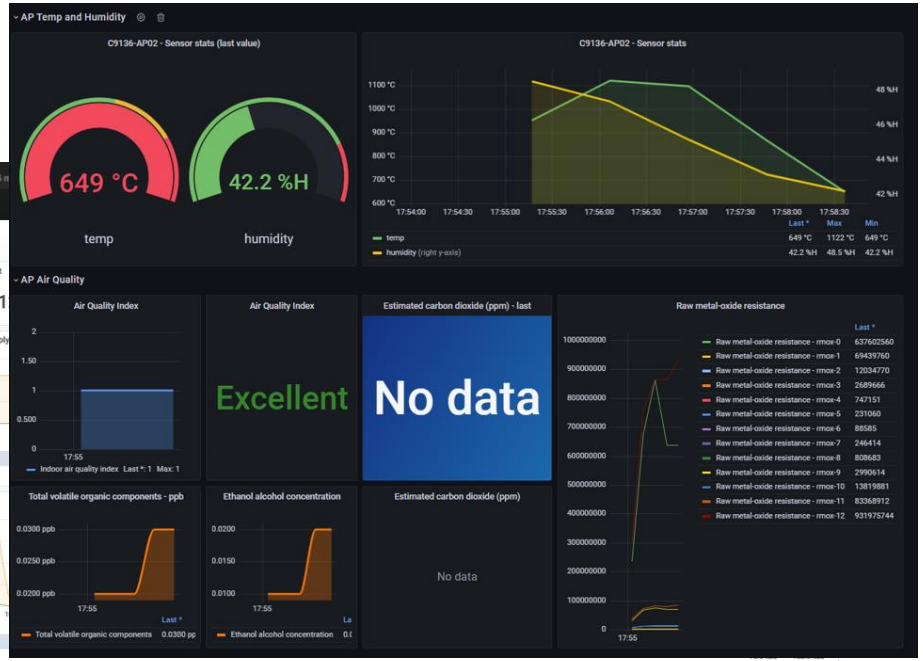
Grafana


https://hub.docker.com/r/jeremycohoe/tig_mdt https://github.com/jeremycohoe/cisco-ios-xe-mdt
https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/prog/configuration/172/b_172_programmability_cg.html

# Example dashboards

https://grafana.com/grafana/dashboards/13462
https://grafana.com/grafana/dashboards/12468



# Migrating from SNMP to gRPC Dial-Out Telemetry

What is the expected increase in CPU/Memory when using the gRPC Dial-Out telemetry interface, compared to SNMP ?

gRPC adds 2% for each telemetry collector
SNMP adds 6% and an additional 4% for each collector

Testbed:

Ubuntu Linux VM
Telegraf Tooling
SNMP + gRPC

C9300-48

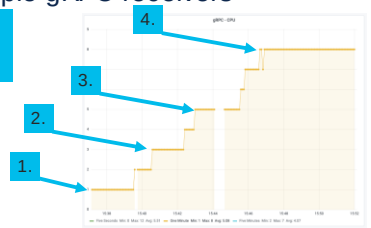
Spirent 48 port traffic generator

## CPU impact with multiple gRPC receivers

1. No Interface collection, CPU @ 1%
2. Add gRPC 1 receiver, CPU @ 3% +2
3. Add 2nd gRPC receiver, CPU @ 5% +2
4. Add 3rd gRPC receiver, CPU @ 8% +3

gRPC configuration

```
telemetry ietf subscription 1000
 encoding encode-kvgpb
 filter xpath /interfaces-ios-xe-oper:interfaces/interface
 source-address 128.107.223.252
 stream yang-push
 update-policy periodic 500
 receiver ip address 128.107.223.248 57500 protocol grpc-tcp

telemetry ietf subscription 1001
 encoding encode-kvgpb
 filter xpath /interfaces-ios-xe-oper:interfaces/interface
 source-address 128.107.223.252
 stream yang-push
 update-policy periodic 500
 receiver ip address 128.107.223.216 57500 protocol grpc-tcp

 telemetry ietf subscription 1002
 encoding encode-kvgpb
 filter xpath /interfaces-ios-xe-oper:interfaces/interface
 source-address 128.107.223.252
 stream yang-push
 update-policy periodic 500
 receiver ip address 128.107.223.236 57500 protocol grpc-tcp
```

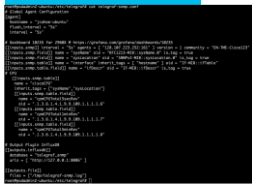


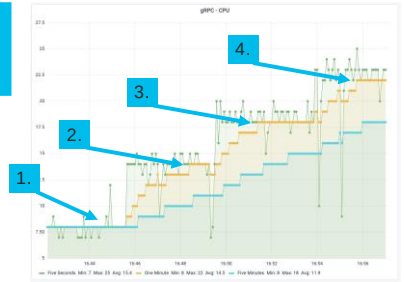

## CPU impact multiple SNMP pollers

Telegraf process, or:
$ snmpwalk -v 1 -c EN-TME-Cisco123 128.107.223.252 1.3.6.1.2.1.2.2

1. Baseline CPU @ 8 %
2. Add 1 SNMP CPU @ 14% +6
3. Add 2nd SNMP CPU @ 18% +4
4. Add 3rd SNMP CPU @ 22% +4

SNMP collection configuration



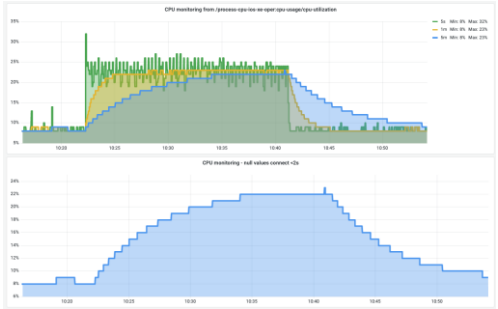


# Model Driven Telemetry: usage comparison

60-minute collection sample with 60-second update interval

| Interface | CPU Impact | PCAP file size/data size (MB) | Data byte Rate | Data bit rate | Average Packet Rate (sec) | Average Packet Size (bytes) |
|---|---|---|---|---|---|---|
| gNMI | +3% | 23 MB | 6 kBps | 53 kbps | 5 | 1180 |
| gRPC | +3% | 69 MB | 19 kBps | 155 kbps | 58 | 333 |
| NETCONF | +2% | 83 MB | 23 kBps | 185 kbps | 29 | 780 |
| RESTCONF | +4% | 200 MB | 35 kBps | 281 kbps | 37 | 945 |
| SNMP * | +6% | 120 / 87 | 24 kBps | 197 kbps | 90 | 273 |


Tested with C9300-48. Testing underway with 8x stack C9300-48



17 xpaths collected at 60 second update interval

/arp-ios-xe-oper:arp-data
/cdp-ios-xe-oper:cdp-neighbor-details
/environment-ios-xe-oper:environment-sensors
/if:interfaces-state
/interfaces-ios-xe-oper:interfaces/interface
/ios:native
/lldp-ios-xe-oper:lldp-entries
/matm-ios-xe-oper:matm-oper-data
/mdt-oper:mdt-oper-data/mdt-subscriptions
/memory-ios-xe-oper:memory-statistics/memory-statistic
/oc-if:interfaces/interface/state/counters
/oc-platform:components
/oc-sys:system
/platform-ios-xe-oper:components
/poe-ios-xe-oper:poe-oper-data/poe-switch
/process-cpu-ios-xe-oper:cpu-usage/cpu-utilization
/process-memory-ios-xe-oper:memory-usage-processes

* SNMP collection of interfaces (IF-MIB) only

# Day 1: YANG Programmatic Interfaces

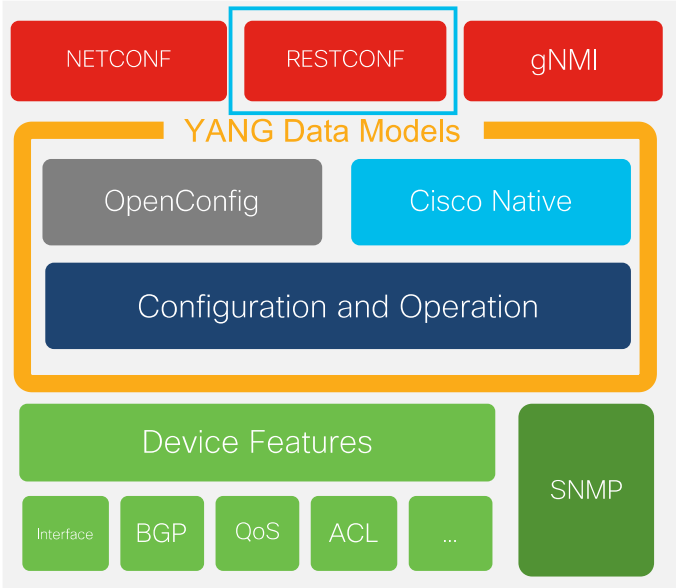
## Terraform
## CLI-to-YANG
## YANG Suite

# Terraform is...

Open-source 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



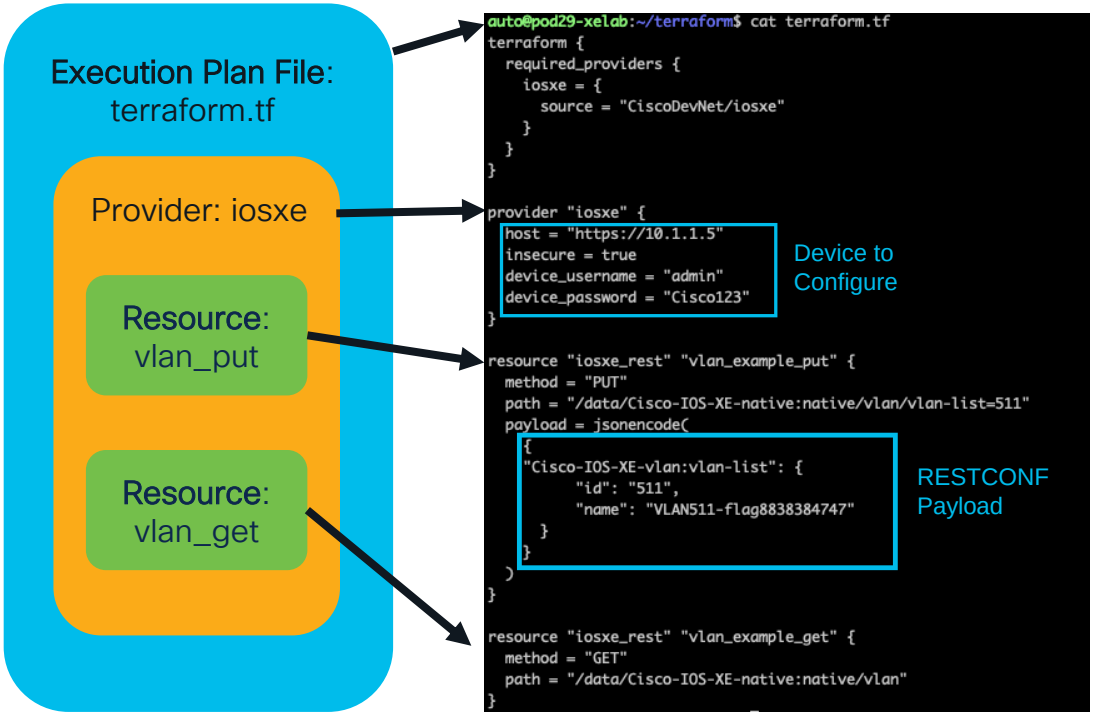
# Terraform terminology

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



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

# 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
```

```
JCOHOE-C9840#show run | format restconf-json
{
 "data": {
  "Cisco-IOS-XE-native:native": {
    "version": "17.8",
    "boot-start-marker": [null],
    "boot": {
      "system": {
        "bootfile": {
          "filename-list-ordered-by-user": [
            {
              "filename": "bootflash:packages.conf"
```

```
JCOHOE-C9840#show run | format netconf-xml
<config xmlns="http://tail-f.com/ns/config/1.0">
  <gnmi-cfg-data xmlns="http://cisco.com/ns/yang/Cisco-IOS-XE-gnmi-cfg">
    <config>
      <enable>true</enable>
      <service>true</service>
    </config>
  </gnmi-cfg-data>
  <mdt-config-data xmlns="http://cisco.com/ns/yang/Cisco-IOS-XE-mdt-cfg">
    <mdt-subscription>
      <subscription-id>1010</subscription-id>
      <base>
        <stream>yang-push</stream>
        <encoding>encode-kvgpb</encoding>
        <source-address>10.85.134.83</source-address>
```

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

# YANG 1.0 to 1.1 transition – YANG advertisement

Planning for YANG 1.0 RFC6020 to RFC7950 YANG 1.1 in IOS XE 17.10 is underway

Legacy YANG 1.0 capabilities exchange and NETCONF "hello" message will soon be unsupported

```
(v) auto@pod24-xelab:~/ncc$
(v) auto@pod24-xelab:~/ncc$ netconf-console --host c9300 --port 830 --user admin --password Cisco123 --hello
/home/auto/ncc/v/lib/python3.8/site-packages/paramiko/kex_ecdh_nist.py:39: CryptographyDeprecationWarning: encode_point has been deprecated on EllipticCurvePublicNumbers and will be
compressed and uncompressed point encoding.
  m.add_string(self.Q_C.public_numbers().encode_point())
/home/auto/ncc/v/lib/python3.8/site-packages/paramiko/kex_ecdh_nist.py:91: CryptographyDeprecationWarning: Support for unsafe construction of public numbers from encoded data will be
  self.Q_S = ec.EllipticCurvePublicNumbers.from_encoded_point(
/home/auto/ncc/v/lib/python3.8/site-packages/paramiko/kex_ecdh_nist.py:103: CryptographyDeprecationWarning: encode_point has been deprecated on EllipticCurvePublicNumbers and will be
n compressed and uncompressed point encoding.
  hm.add_string(self.Q_C.public_numbers().encode_point())
<?xml version="1.0" encoding="UTF-8"?>
<nc:hello xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
  <nc:capabilities>
    <nc:capability>urn:ietf:params:netconf:base:1.0</nc:capability>
    <nc:capability>urn:ietf:params:netconf:base:1.1</nc:capability>
    <nc:capability>urn:ietf:params:netconf:capability:writable-running:1.0</nc:capability>
    <nc:capability>urn:ietf:params:netconf:capability:rollback-on-error:1.0</nc:capability>
    <nc:capability>urn:ietf:params:netconf:capability:validate:1.0</nc:capability>
    <nc:capability>urn:ietf:params:netconf:capability:validate:1.1</nc:capability>
    <nc:capability>urn:ietf:params:netconf:capability:xpath:1.0</nc:capability>
    <nc:capability>urn:ietf:params:netconf:capability:notification:1.0</nc:capability>
    <nc:capability>urn:ietf:params:netconf:capability:interleave:1.0</nc:capability>
    <nc:capability>urn:ietf:params:netconf:capability:with-defaults:1.0?basic-mode=explicit&amp;also-supported=report-all-tagged,report-all</nc:capability>
    <nc:capability>urn:ietf:params:netconf:capability:yang-library:1.0?revision=2016-06-21&amp;module-set-id=4702f66e9d4acb0c47ccf523dcafba47</nc:capability>
    <nc:capability>http://tail-f.com/ns/netconf/actions/1.0</nc:capability>
    <nc:capability>http://cisco.com/ns/cisco-xe-ietf-ip-deviation?module=cisco-xe-ietf-ip-deviation&amp;revision=2016-08-10</nc:capability>
    <nc:capability>http://cisco.com/ns/cisco-xe-ietf-ipv4-unicast-routing-deviation?module=cisco-xe-ietf-ipv4-unicast-routing-deviation&amp;revision=2015-09-11</nc:capability>
    <nc:capability>http://cisco.com/ns/cisco-xe-ietf-ipv6-unicast-routing-deviation?module=cisco-xe-ietf-ipv6-unicast-routing-deviation&amp;revision=2015-09-11</nc:capability>
    <nc:capability>http://cisco.com/ns/cisco-xe-ietf-ospf-deviation?module=cisco-xe-ietf-ospf-deviation&amp;revision=2018-02-09</nc:capability>
    <nc:capability>http://cisco.com/ns/cisco-xe-ietf-routing-deviation?module=cisco-xe-ietf-routing-deviation&amp;revision=2016-07-09</nc:capability>
    <nc:capability>http://cisco.com/ns/cisco-xe-openconfig-acl-deviation?module=cisco-xe-openconfig-acl-deviation&amp;revision=2017-08-25</nc:capability>
    <nc:capability>http://cisco.com/ns/cisco-xe-openconfig-aft-deviation?module=cisco-xe-openconfig-aft-deviation&amp;revision=2018-12-05</nc:capability>
    <nc:capability>http://cisco.com/ns/cisco-xe-openconfig-isis-deviation?module=cisco-xe-openconfig-isis-deviation&amp;revision=2018-12-05</nc:capability>
    <nc:capability>http://cisco.com/ns/cisco-xe-openconfig-lldp-deviation?module=cisco-xe-openconfig-lldp-deviation&amp;revision=2018-07-25</nc:capability>
    <nc:capability>http://cisco.com/ns/cisco-xe-openconfig-mpls-deviation?module=cisco-xe-openconfig-mpls-deviation&amp;revision=2019-06-27</nc:capability>
    <nc:capability>http://cisco.com/ns/cisco-xe-openconfig-segment-routing-deviation?module=cisco-xe-openconfig-segment-routing-deviation&amp;revision=2018-12-05</nc:capability>
    <nc:capability>http://cisco.com/ns/cisco-xe-openconfig-system-management-deviation?module=cisco-xe-openconfig-system-management-deviation&amp;revision=2019-07-01</nc:capability>
```

YANG 1.1 example: "ietf-yang-library" to retrieve supported YANG modules

```
Sending:

#275
<nc:rpc xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="urn:uuid:b966c2ff-a59b-46a3-aa31-cb1cc5e97e44">
  <nc:get>
    <nc:filter>
      <modules-state xmlns="urn:ietf:params:xml:ns:yang:ietf-yang-library"/>
    </nc:filter>
  </nc:get>
</nc:rpc>

##

Received message from host

<?xml version="1.0" ?>
<rpc-reply message-id="urn:uuid:b966c2ff-a59b-46a3-aa31-cb1cc5e97e44" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" xmlns:nc="urn:ietf:para
  <data>
    <modules-state xmlns="urn:ietf:params:xml:ns:yang:ietf-yang-library">
      <module-set-id>4702f66e9d4acb0c47ccf523dcafba47</module-set-id>
      <module>
        <name>BGP4-MIB</name>
        <revision>1994-05-05</revision>
        <schema>http://localhost:9938/restconf/tailf/modules/BGP4-MIB/1994-05-05</schema>
        <namespace>urn:ietf:params:xml:ns:yang:smiv2:BGP4-MIB</namespace>
        <conformance-type>implement</conformance-type>
      </module>
      <module>
        <name>BRIDGE-MIB</name>
        <revision>2005-09-19</revision>
        <schema>http://localhost:9938/restconf/tailf/modules/BRIDGE-MIB/2005-09-19</schema>
        <namespace>urn:ietf:params:xml:ns:yang:smiv2:BRIDGE-MIB</namespace>
        <conformance-type>implement</conformance-type>
      </module>
```

If the desired application previously parsed the NETCONF "hello" message to retrieve the supported YANG models, the parsing must be modified to reflect how version 1.1 advertises via "ietf-yang-library" instead of the NETCONF "hello" message.

# Innovations in wireless YANG

## Cisco IOS XE wireless mesh exec RPC

## Access Point Oper Data: YANG improvements

| Container/leaf name | TYPE | xpath |
|---|---|---|
| SSID name | string | /acce |
| SSID state | Boolean | |
| number of associated clients | | /acce |
| BSSID | | /acce |


## On Change telemetry: AP oper


| Gather Point |
|---|
| /access-point-oper-dat data/phy-ht-cfg/cfg-da |
| /access-point-oper-dat |
| /access-point-oper-dat |
| access-point-oper-data |

## 30s Periodic telemetry: AP and Client oper

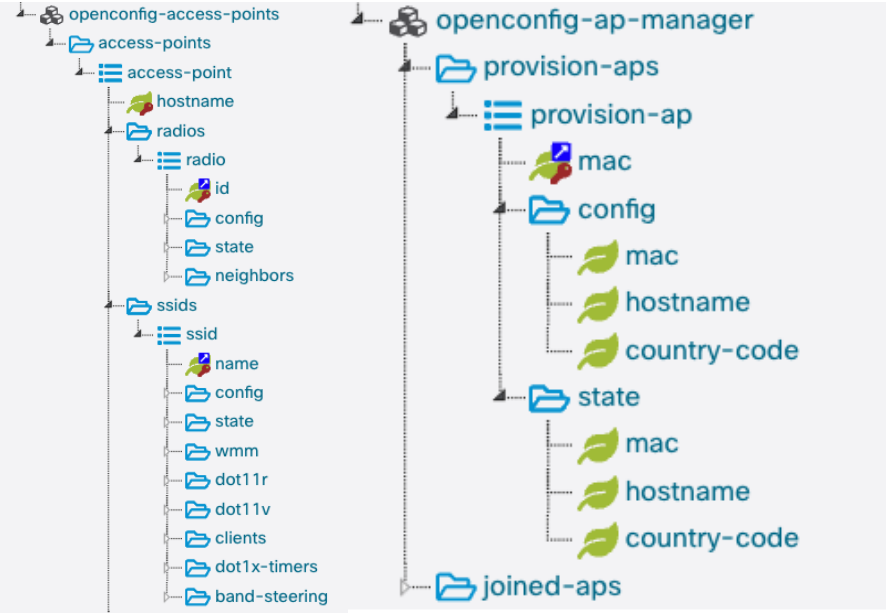

| Gather Point | Xpath |
|---|---|
| /ap-global-oper-data/ap-join-stats | /ap-global-oper-data/ap-join-stats/wtp-mac |
| | /ap-global-oper-data/ap-join-stats/ap-join-info/ap-ethernet-mac |
| | /ap-global-oper-data/ap-join-stats/ap-join-info/ap-name |
| | /ap-global-oper-data/ap-join-stats/ap-join-info/ap-ip-addr |
| | /ap-global-oper-data/ap-join-stats/ap-join-info/is-joined |
| | /ap-global-oper-data/ap-join-stats/ap-join-info/last-error-type |
| | /ap-global-oper-data/ap-join-stats/ap-disconnect-reason |
| /client-oper-data/traffic-stats | /client-oper-data/traffic-stats/rx-group-counter |

# OC wireless: YANG model end of support

17.9 will be the last release supporting the OC wifi YANG

https://github.com/openconfig/public/tree/master/release/models/wifi

- The YANG for OpenConfig Wireless including OpenConfig-access-points and OpenConfig-ap-manager YANG are no longer being supported after 17.9
- The constructs within OC Wireless support only flex deployments and direct AP management. There is no modelled concept of any CAPWAP tunnels or centralized controller infrastructure
- All or nothing: OC Wifi leverages a hostname centric view and does not use the MAC address. All config and operations must be via OC-wifi.YANG as the traditional YANG/CLI uses MAC centric view
- Most deployment are controller based/local mode, so the OC Wifi model is not applicable and not usable for most deployments
- OC Model version drift: the initial version implemented of 0.1.0 or 0.2.0 is not current with GitHub version of 1.0.0 so there are many mapping gaps making it even less usable



# Day 2:

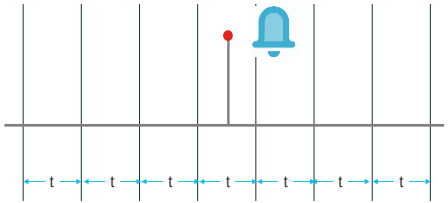
# Model Driven Telemetry

# Telemetry publication options

Operational telemetry periodic and on-change update of AP status (joined/not joined) and client traffic stats

- Periodic + On-Change MDT
- AP-oper YANG
- Mesh exec RPC's

On-Change



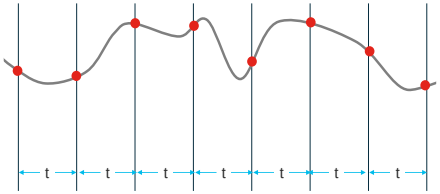

Feature Model "On-Change" Notifications

Event Notifications (failed login, optic fault, etc.)

State and Configuration

AP Oper / CAPWAP / radio / etc

Periodic




Feature Model "Periodic" Notifications

Time based publication

Minimum interval 100 centiseconds (1s)

AP Join Stats / Traffic Stats

# Streaming Telemetry support for 9840/9880

gNMI Dial-In and gRPC Dial-Out for the following KPI's and YANG

Cisco-IOS-XE-wireless-access-point-oper.yang
Cisco-IOS-XE-wireless-client-oper.yang
Cisco-IOS-XE-wireless-rrm-oper.yang

Recommended telemetry collection intervals with one or two collectors is shown on the next slides and described in more detail in the Programmability Configuration Guide.

Per AP Radio Stats – keyed by AP MAC + Slot ID:
channel, channel width, total/tx/rx channel utilization, rx noise channel utilization, noise floor, frequency, tx power, neighbor table

Per AP Per Radio Per SSID stats – keyed by AP MAC, slot ID, ap-vap-id, wlan_id:
SSID name/state/hidden/VLAN, number of associated clients, BSSID

Client Counters – keyed by client MAC address
AP/SSID name, phy rate, connection-mode, rssi, frequency, IPv4/IPv6 address, tx retries, data tx/rx-bytes+pkts, broadcast packet count, channel-support

AP System – keyed by AP MAC
Hostname, joined-controller, software version, 802.3 speed, IPv4/IPv6 address, power lldp, serial, uptime, state, CPU, memory

BSSID Counters – keyed by AP MAC + Slot ID + WLAN ID
Tx-retries-data, tx-retries-subframe, rx-data-dist, rx-mcs, tx-data-dist, tx-mcs

Neighbor details – keyed by AP MAC + Slot ID + BSSID
Ssid, rssi, channel, primary-channel, last-update-rcvd

# Innovations in wireless Telemetry

Source: https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/prog/configuration/178/b_178_programmability_cg/m_178_prog_ietf_telemetry.html

## Wireless Telemetry Full Scale

Six SSIDs at Scale Phase 1

| Gathering Point | Records |
|---|---|
| Joined | 2,000 |
| AAA | 2,000 |
| Radio | 4,000 |
| Client RF | 30,000 |
| Client CNTR | 30,000 |
| Client CONN | 30,000 |
| BSSID | 24,000 |
| Neighbor | 288,000 |

## Wireless Telemetry Full Scale

Four SSIDs at Scale Phase 1

| Gathering Point | Records | Recommende Interval (Seco One Collector |
|---|---|---|
| Joined | 2,000 | 30 |
| AAA | 2,000 | 30 |
| Radio | 4,000 | 30 |
| Client RF | 30,000 | 30 |
| Client CNTR | 30,000 | 30 |
| Client CONN | 30,000 | 60 |
| BSSID | 16,000 | 90 |
| Neighbor | 192,000 | 180 |

## Wireless Telemetry Reduced Scale

Six SSIDs at Scale Phase 1

| Gathering Point | Records | Recommended Interval (Seconds) One Collector | Recommended Interval (Seconds) Two Collectors |
|---|---|---|---|
| Joined | 1,000 | 30 | 30 |
| AAA | 1,000 | 30 | 30 |
| Radio | 2,000 | 30 | 30 |
| Client RF | 15,000 | 30 | 30 |
| Client CNTR | 15,000 | 30 | 30 |
| Client CONN | 15,000 | 30 | 30 |
| BSSID | 12,000 | 120 | 120 |
| Neighbor | 144,000 | 180 | 180 |


30 seconds is recommended periodic update interval for wireless metrics

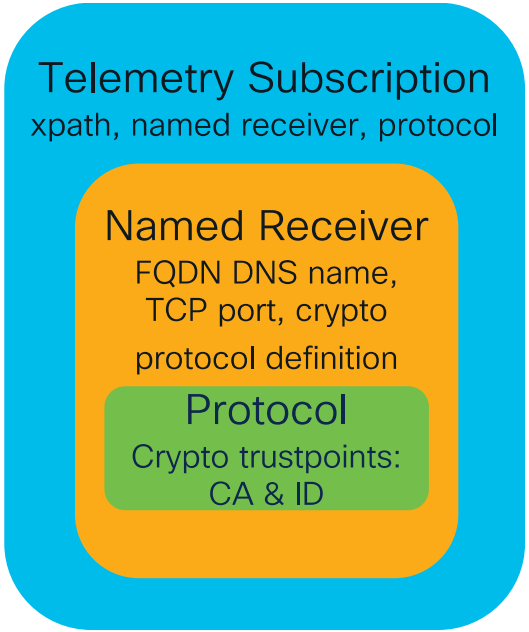
# gRPC Dial-Out with DNS + mTLS

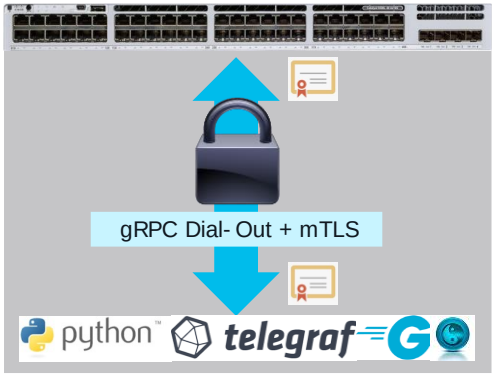
**Zero Trust Telemetry:** mTLS for gRPC Dial-Out by defining the telemetry configuration with the CA and ID crypto trustpoints

```
telemetry ietf subscription 1010
 encoding encode-kvgpb
filter xpath /wireless-ble-ltx-oper:ble-ltx-oper-data/ble-ltx-ap-streaming
 source-address 10.85.134.83
 stream yang-push
 update-policy periodic 6000
 receiver-type protocol
 receiver name yangsuite

telemetry receiver protocol yangsuite
 host name yangsuite-telemetry.cisco.com 57501
 protocol grpc-tls profile mtlsyangsuite

telemetry protocol grpc profile mtlsyangsuite
  ca-trustpoint myCA
  id-trustpoint myID
```

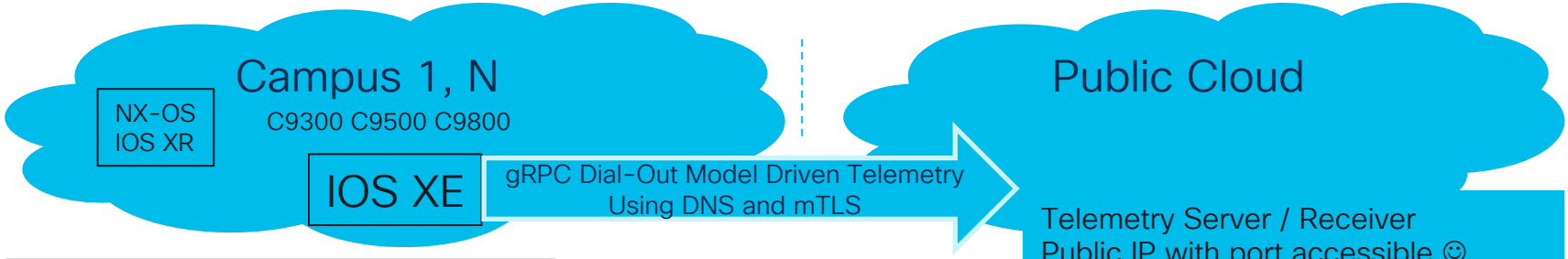
Crypto trustpoints (myCA and myID) and certifications are defined before telemetry configuration using:

- gNOI cert.proto
- Cisco-IOS-XE-crypto-rpc.YANG
- "crypto pki import" CLI



# Case Study: Telemetry in production




```
C9300#show run | i domain / i name-server
ip name-server 208.67.222.222
ip domain lookup

C9300#show run | sec telemetry
telemetry ietf subscription 101
 encoding encode-kvgpb
 filter xpath /access-point-oper-data/capwap-data
 receiver-type protocol
 source-address 10.4.20.188
 source-vrf Mgmt-intf
 stream yang-push
 update-policy periodic 6000
 receiver name yangsuite

telemetry receiver protocol yangsuite
 host name yangsuite-telemetry.cisco.com 443
 protocol grpc-tls profile yangsuite

telemetry protocol grpc profile yangsuite
 ca-trustpoint CAforMDTserver
 id-trustpoint CAforWLCclient
```

DNS Load Balancing is used to distribute the connections across a group of servers

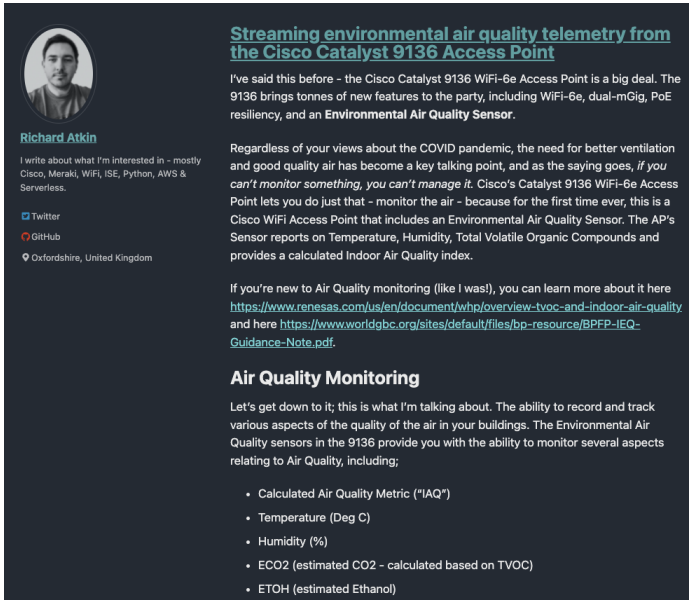
```
dig +short cisco.com mx
30 aer-mx-01.cisco.com.
10 alln-mx-01.cisco.com.
20 rcdn-mx-01.cisco.com.
```

DNS Geo-location uses the geographic location of the client and the server resource

```
lab8-co-wlc-1#show telemetry connection all
Telemetry connections

Index Peer Address               Port  VRF Source Address             State      State Description
----- -------------------------- ----- --- -------------------------- ---------- --------------------
49182 52.                          443 0   10.                        Active     Connection up
```

# Partner telemetry examples

https://richardatkin.com/post/2022/03/22/Streaming-environmental-air-quality-data-from-the-Cisco-Catalyst-9136-WiFi6e-Access-Point.html



Richard Atkin

I write about what I'm interested in - mostly Cisco, Meraki, WiFi, ISE, Python, AWS & Serverless.

Twitter
GitHub
Oxfordshire, United Kingdom

## Streaming environmental air quality telemetry from the Cisco Catalyst 9136 Access Point

I've said this before - the Cisco Catalyst 9136 WiFi-6e Access Point is a big deal. The 9136 brings tonnes of new features to the party, including WiFi-6e, dual-mGig, PoE resiliency, and an **Environmental Air Quality Sensor**.

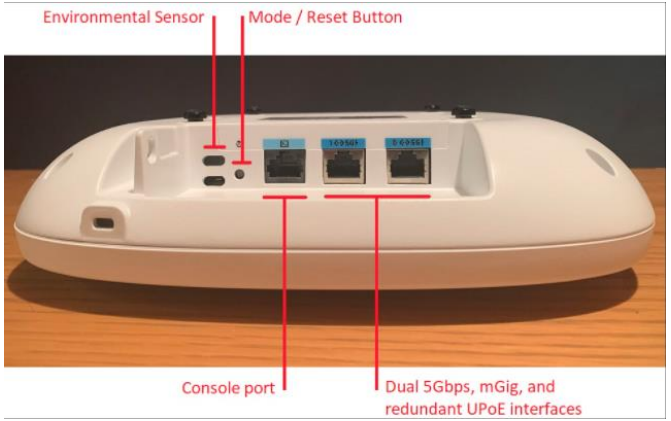
Regardless of your views about the COVID pandemic, the need for better ventilation and good quality air has become a key talking point, and as the saying goes, *if you can't monitor something, you can't manage it.* Cisco's Catalyst 9136 WiFi-6e Access Point lets you do just that – monitor the air - because for the first time ever, this is a Cisco WiFi Access Point that includes an Environmental Air Quality Sensor. The AP's Sensor reports on Temperature, Humidity, Total Volatile Organic Compounds and provides a calculated Indoor Air Quality index.

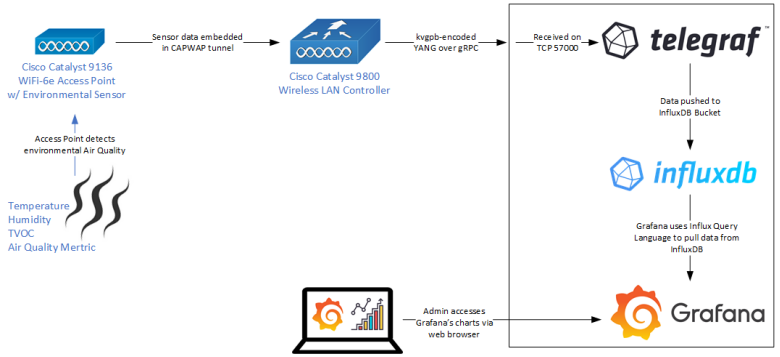
If you're new to Air Quality monitoring (like I was!), you can learn more about it here https://www.renesas.com/us/en/document/whp/overview-tvoc-and-indoor-air-quality and here https://www.worldgbc.org/sites/default/files/bp-resource/BPFP-IEQ-Guidance-Note.pdf.

### Air Quality Monitoring

Let's get down to it; this is what I'm talking about. The ability to record and track various aspects of the quality of the air in your buildings. The Environmental Air Quality sensors in the 9136 provide you with the ability to monitor several aspects relating to Air Quality, including;

- Calculated Air Quality Metric ("IAQ")
- Temperature (Deg C)
- Humidity (%)
- ECO2 (estimated CO2 - calculated based on TVOC)
- ETOH (estimated Ethanol)



# Day N – On-Box Automation

# ZTP, Guest Shell, EEM, Python & NETCONF API

# gNOI

# gNXI

| | C9800-40<br>C9800-80 | C9800-L | C9800-CL | EWC |
|---|---|---|---|---|
| os.proto | Roadmap | Roadmap | Roadmap | Roadmap |
| reset.proto | Supported | Roadmap | Roadmap | Roadmap |
| cert.proto | Enabled | Enabled | Roadmap | Roadmap |

The ‘gnmi’ CLI has been renamed to the ‘gnxi’ CLI in 17.3 as more protocols are “ratified” in the Github/openconfig and Github/gnxi organizations and implemented by vendors

```
C9300#show gnxi state detail
Settings
========
  Server: Disabled
  Server port: 50052
  Secure server: Enabled
  Secure server port: 9339
  Secure client authentication: Disabled
  Secure trustpoint: gnxi-cert
  Secure client trustpoint:
  Secure password authentication: Disabled
```

```
GNMI
====
  Admin state: Enabled
  Oper status: Up
  State: Provisioned

  gRPC Server
  -----------
    Admin state: Enabled
    Oper status: Up

  Configuration service
  ---------------------
    Admin state: Enabled
    Oper status: Up

  Telemetry service
  -----------------
    Admin state: Enabled
    Oper status: Up
```

```
GNOI
====

  Cert Management service
  -----------------------
    Admin state: Enabled
    Oper status: Up
```

**gNMI Clients:**

- gNMI Capabilities
- gNMI Get
- gNMI Set
- gNMI Subscribe

**gNOI Clients**

- gNOI Cert
- gNOI OS
- gNOI Reset

**gNxI Tools**

- gNMI - gRPC Network Management Interface
- gNOI - gRPC Network Operations Interface

https://github.com/google/gnxi

# ZTP Overview

1. When an IOS XE device boots and no configuration is present, the device will issue a DHCP request on the management port and on the front panel port.

2. If the DHCP response contains option 67 then ZTP is initiated and the device will retrieve and execute the python script from within the Guest Shell

3. Guest Shell is started and networking is automatically configured

Cisco Blogs

Cisco Blog > Developer

Developer

## Automate Device Provisioning with Cisco IOS XE Zero Touch Provisioning

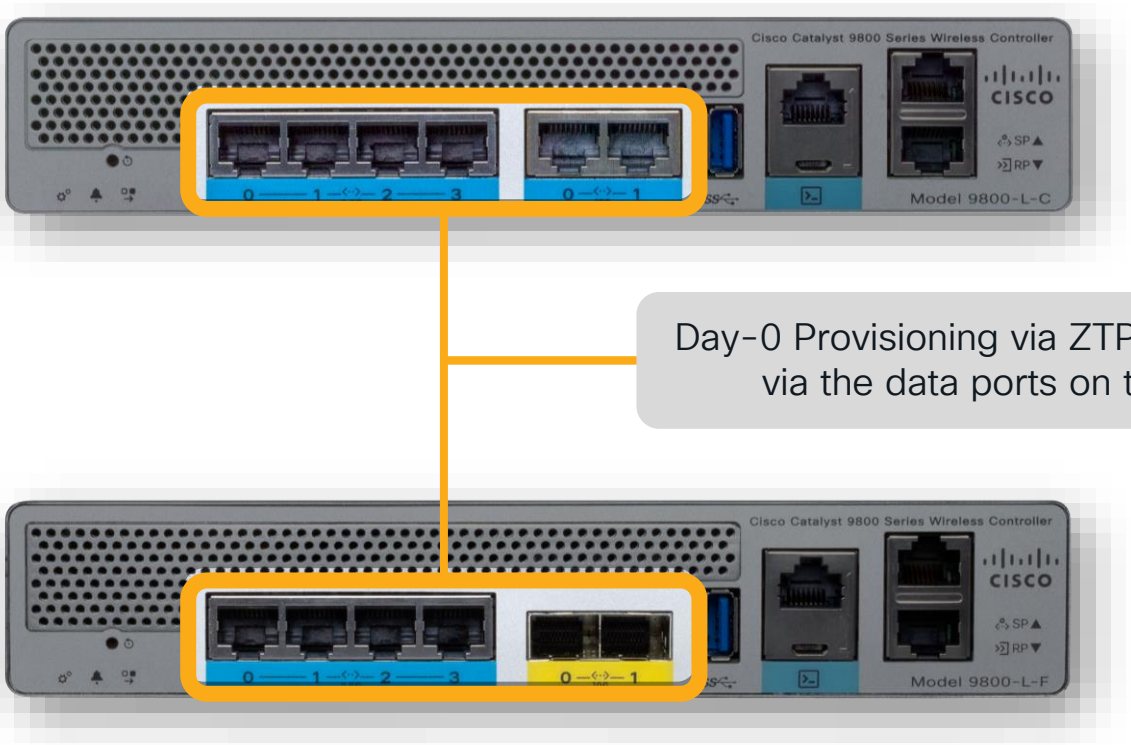
Jeremy Cohoe
April 15, 2019 - 1 Comment

When new hardware is ordered and it arrives on site, it's an exciting time. New hardware! New software! ... But new challenges too! But the age-old challenge of getting new devices on the network doesn't need to be one of them. Sitting in the lab pre-provisioning devices is no longer required if you're using Cisco IOS XE, because of features like Cisco Network Plug-n-Play (PnP) and Zero Touch Provisioning (ZTP). PnP is the premium solution made possible with Cisco DNA Center, while Zero Touch Provisioning (ZTP) is for the do-it-yourself customers who don't mind investing more time in configuring and maintaining the infrastructure required to bootstrap devices. IOS XE runs on the enterprise hardware and

https://www.youtube.com/watch?v=EAXnftG6odg

https://blogs.cisco.com/developer/device-provisioning-with-ios-xe-zero-touch-provisioning

# ZTP Support on Data Port on 9800-L




NEW

Day-0 Provisioning via ZTP now supported via the data ports on the 9800-L

# ZTP timing improvements for C9800

- Improvements in 17.3 reduce boot time and ZTP execution time, including starting of the Guest Shell faster and executing the python script sooner

- The previous Python **file size limitation of 8k has been removed** – allows for more customization and flexibility during day 0
- Larger Python files are becoming more common as additional logic, loops, and intelligence are built into scripts as needed

# ZTP MSDC customer use case – workflow

- Customer problem statement: How can we reduce the manual work that takes time which increases costs and risk when deploying networks at various locations globally ?
- Zero Touch Provisioning and automated onboarding solution for multiple devices in multiple locations
  - (retail, warehouse, corporate branch and campus)
- DHCP connectivity required and new devices are upgraded, configured, and available for management without any manual interaction – 100's – 1000's of devices deployed and configured with zero touch

IOS XE will use the option 67 for ZTP first then the option 150 for Auto Install

Auto Install is the "backup" for ZTP

Day 0 workflow

## option 67 for ZTP python script

Python3 script + Guest Shell
Software upgrade + EEM
Download and apply pre-generated config

## option 150 for AutoInstall CLI config

Auto Install CLI config
Set basic device config

# ZTP MSDC customer use case - details

**DHCP server configuration with both options 67 and 150 configured**

```
option bootfile-name "http://10.85.134.66/ztp.py";
option tftp-server-name "10.85.134.66";
```

**option 67 for ZTP.py with Guest Shell and python3**

Define software images, version, and checksum
Define HTTP server and version to use
Check if software upgrade is required
    Install and cleanup with EEM applets
Download config file for the device type from HTTP
    One pre-generated config file for each type
    Apply with "copy device.cfg running-config"
Logs the output to flash/ztp.log

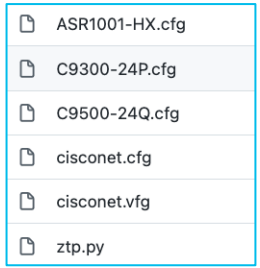
**option 150 for cisconet.cfg for DHCP Auto Install**

Hostname: autoinstall_config
Same featureset is configured

Pre-generated config files
ASR1001-HX.cfg
Hostname: AR1001-HX-default

C9300.cfg
Hostname: C9300_default

C9500.cfg
Hostname: C9500_default




**Configured and enabled features**

Console password, Enable AAA + credentials, crypto, SSH, netconf-yang, DHCP on Gi0/0 and/or Gi0, set defaults for TFTP block size, LLDP, MST, logging, IOX, line settings

https://github.com/jeremycohoe/IOSXE-Zero-Touch-Provisioning
https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/prog/configuration/177/b_177_programmability_cg/m_177_prog_ztp.html
https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/fundamentals/configuration/xe-16-6/fundamentals-xe-16-6-book/cf-auto-install.html

# 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!

# 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**

# 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.

# Cisco DEVNET

Start Now

Videos and Tutorials

Sandbox Learning Lab

Automation and Code Exchange

Learning and Certifications

Community and Study Groups

developer.cisco.com

# 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

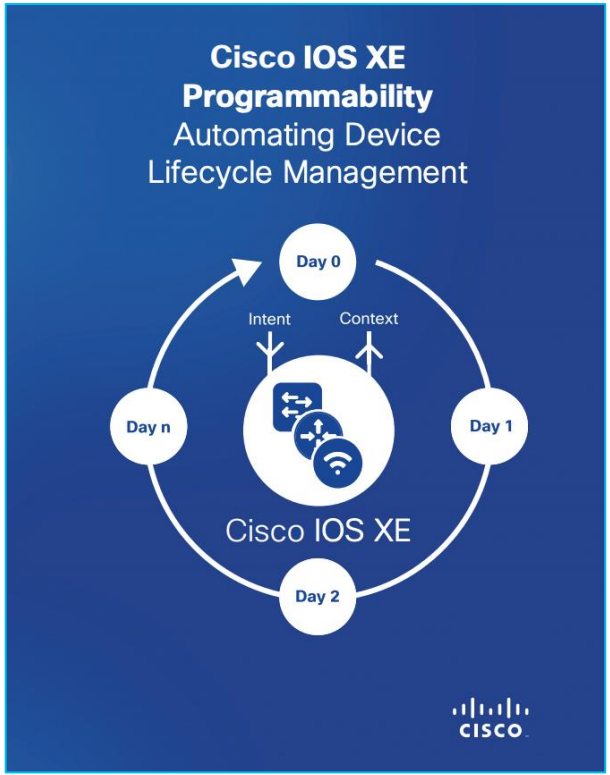
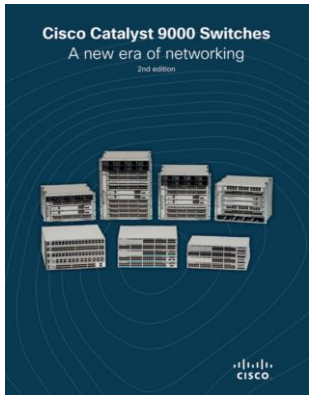
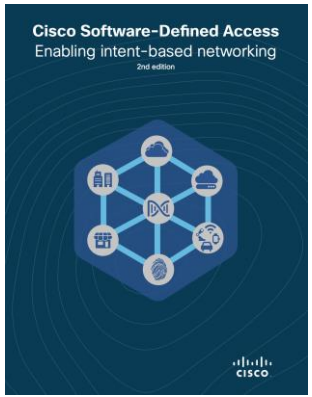



Table of Contents

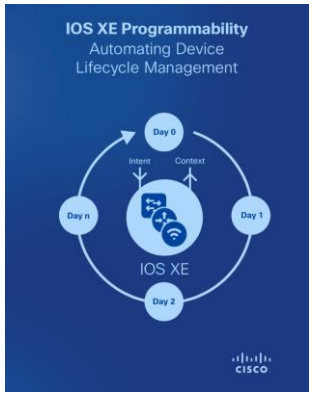
# Enterprise Networks booksprints

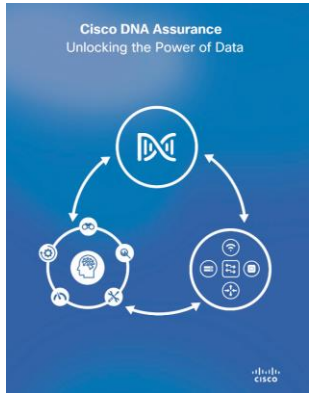
http://cs.co/cat9000book
http://cs.co/sdabook
http://cs.co/wirelessbook
http://cs.co/programmabilitybook
http://cs.co/assurancebook
http://cs.co/sdwanbook

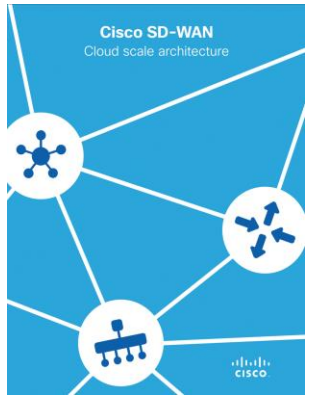






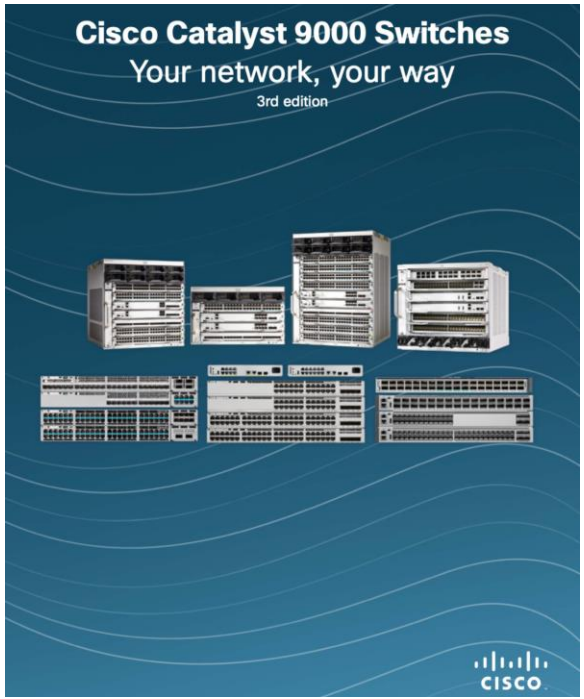



# Cisco Catalyst 9000 Switches – 3rd edition

Cisco Whitepaper: https://www.cisco.com/c/en/us/products/collateral/switches/catalyst-9000/nb-06-cat9k-ebook-cte-en.html
Direct ebook download: https://www.cisco.com/c/dam/en/us/products/collateral/switches/catalyst-9000/nb-06-cat9k-ebook-cte-en.pdf

Cisco Whitepaper




ebook PDF

# Programmability Configuration Guide



**Programmability Configuration Guide, Cisco IOS XE Cupertino 17.8.x**

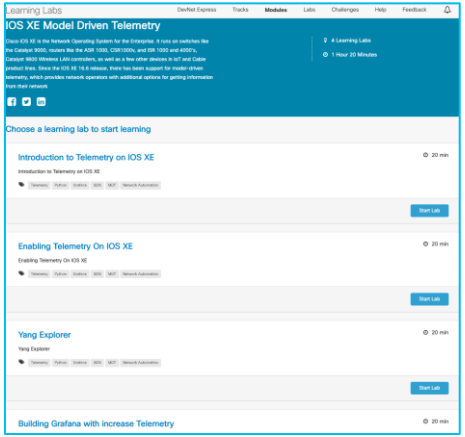
**First Published:** 2022-04-09

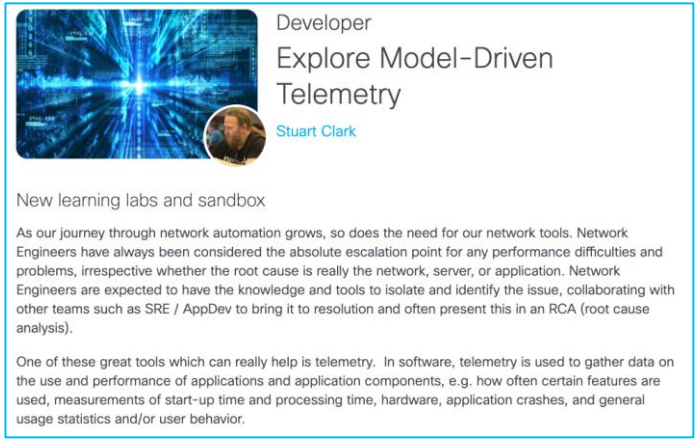
**Last Modified:** 2022-06-01

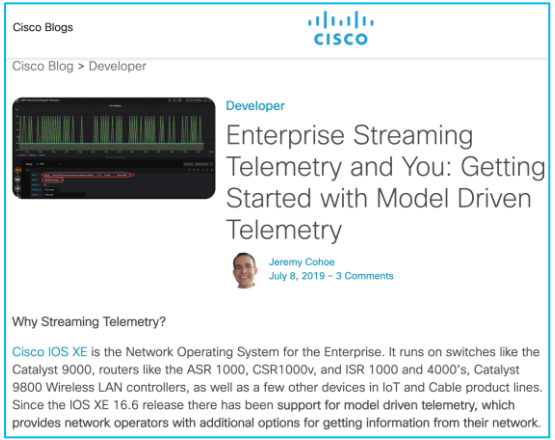
https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/prog/configuration/178/b_178_programmability_cg.html

# Learning Lab and Blog: IOS XE MDT

https://developer.cisco.com/learning/modules/iosxe_telemetry
https://blogs.cisco.com/developer/model-driven-telemetry-sandbox
https://blogs.cisco.com/developer/getting-started-with-model-driven-telemetry
https://youtu.be/QwwZakkWBng



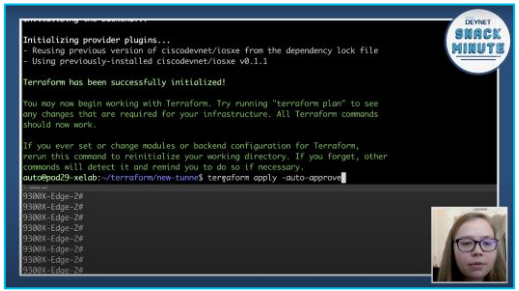



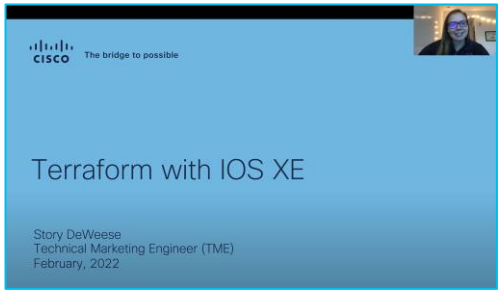
# Terraform blog and resources

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

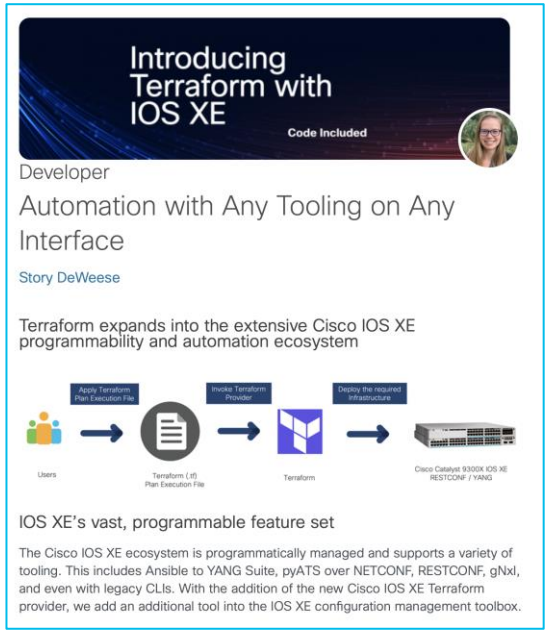
https://github.com/CiscoDevNet/terraform-provider-iosxe/
https://registry.terraform.io/search/providers?namespace=CiscoDevNet




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




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


# Thank you


cisco Live!
ALL IN