Open IOS XE

Python support (on-box)

Q

How is a switch protected against errant scripts?

A

Python scripts are executed in a Linux container with built-in security. Only a limited amount of CPU can be used by the Linux container (guestshell). There is Linux user namespace protection for the Python guestshell so that the box cannot be hacked or compromised.

Q

What is Python programmability?

A

Python programmability provides users with the ability to control devices running the Cisco IOS® XE 16.5+ operating system by writing Python code that makes use of APIs. It has multiple use cases, such as:

- Interactive Python prompts
- Running Python scripts
- Zero-Touch Provisioning (ZTP)
- Cisco IOS Embedded event manager

Q

What version of Python is on the switch by default?

A

Python 2.7.11.

Q

Can I upgrade to Python 3?

A

Yes; there are two methods you can use to upgrade to Python 3 on the Cisco® Catalyst® 9000 platform.

- Using a standalone/self-contained installation tarball

  Refer to https://wiki.cisco.com/display/GRIFF/Python3+installation for more details.
- Using 'yum install' for python3 and 'pip install' for the Command-Line Interface (CLI) module

In situations where the user can provide network access from the device (and hence from guestshell) to an external repository, the user may choose to use 'yum install' to install Python 3. The CLI module needs to be installed separately using 'pip install'.

Application hosting

Q

A

Which switching platforms support application hosting?

Application hosting is supported on all Cisco Catalyst 9000 family platforms starting with the 16.5.1a release. It is supported on routing platforms running open Cisco IOS XE such as the Cisco 4000 Series Integrated Services Routers (ISRs) as well.

Q

A

Can applications running on the device compromise the performance or the security of the device?

No. Applications run in either a virtual machine or a Linux container and are isolated from the main operating system. Moreover, only a portion of the system resources (RAM, CPU, and storage) are allocated for a given application.

Q

A

Can different applications be hosted on the same device?

Yes, different VMs and Linux containers can run simultaneously on the devices, and different applications can run in the same VM or container. The only limitation is the resources (RAM, CPU, storage) available for application hosting on the device.

Q

A

Is automated application lifecycle management possible?

Yes. An application's lifecycle, from initial deployment through ongoing change management and application retirement, can be managed using Cisco Fog Director through a visual web environment or integrated with existing management systems through APIs.

The other option is to download the package locally to the device and use the new open Cisco IOS XE app-hosting CLIs.

Q

A

How is application hosting different from guestshell?

Guestshell is just an example of application hosting. It is built into the open Cisco IOS XE image and is meant to be an environment for Python scripting only. There is no need to install any image; just enable both Cisco IOx and guestshell.

Q

A

What is Cisco IOx?

Cisco IOx is Cisco's implementation of "fog computing." It enables hosting of applications on the devices. For more information, visit: https://developer.cisco.com/site/iox/

Q

A

Do hosted applications have access to front panel ports?

No, only to the management port for now. Access to front panel ports will be added in a future release.

Puppet

Q

A

What is Puppet?

Puppet is a Configuration Management Tool (CMT) used for centralizing and automating configuration management. Traditionally, Puppet has used an agent-based architecture, requiring a software agent to be installed on the device being managed. However, Puppet now supports direct NETCONF integration to Cisco IOS XE devices.

Q

A

Why would a customer use a configuration management tool?

CMTs automate the provisioning and configuration of network devices. If a customer has a large number of devices to configure, it is easier to use a CMT than to configure devices manually using a CLI. Additionally, CMTs such as Puppet usually operate on a declarative model, which means that users specify their intent (what they want) rather than giving specific instructions (how to do it), leaving implementation details to the tool.

Q **Why would a customer choose Puppet versus one of the other configuration management tools?**

A Most likely because they are already using Puppet and don't want to use another tool.

Q **How is configuration specified for Puppet?**

A Puppet has a text file called a manifest that has details of what configuration should be pushed to what devices.

Q **How does the agentless architecture work?**

A Puppet uses NETCONF to communicate with the switch. Therefore, any Cisco Catalyst switch capable of NETCONF will in theory be supported by Puppet.

Chef

Q **What is Chef?**

A Chef is a Configuration Management Tool (CMT) used for centralizing and automating configuration management. Traditionally, Chef has used an agent-based architecture, requiring a software agent to be installed on the device being managed.

Q **Why would a customer use a configuration management tool?**

A CMTs automate the provisioning and configuration of network devices. If a customer has a large number of devices to configure, it is easier to use a CMT than to configure devices by hand using a CLI. Additionally, CMTs such as Chef usually operate on a declarative model, which means that users specify their intent (what they want) rather than giving specific instructions (how to do it), leaving implementation details to the tool.

Q **Why would a customer choose Chef versus one of the other configuration management tools?**

A Most likely because they are already using Chef and don't want to use another tool.

Q **How is configuration specified for Chef?**

A Chef has a text file called a cookbook that has details on what configuration should be pushed to what devices.

Q **What is required for a switch to use Chef?**

A Chef requires a software agent on the device under management. This requires guestshell running on the switch to host the Chef agent. However, there are no plans to support Chef agents on Cisco Catalyst platforms at this time.

Ansible

Q **What is Ansible?**

A Ansible is a Configuration Management Tool (CMT) used for centralizing and automating configuration management. Ansible uses an agentless architecture, which means that it does not require a software agent to be installed on the devices under management.

Q **Why would a customer use a configuration management tool?**

A CMTs automate the provisioning and configuration of network devices. If a customer has a large number of devices to configure, it is easier to use a CMT than to configure devices by hand using a CLI. Additionally, CMTs such as Ansible usually operate on a declarative model, which means that users specify their intent (what they want) rather than giving specific instructions (how to do it), leaving implementation details to the tool.

Why would a customer choose Ansible versus one of the other configuration management tools?

A Ansible has been popular because of its agentless architecture. Customers might also choose it because they are already using Ansible and don't want to use another tool.

How is configuration specified for Ansible?

A Ansible has a text file called a playbook that has details of what configuration should be pushed to what devices.

What is required for a switch to use Ansible?

A Ansible has a powerful CLI templating engine, which uses the Jinja2 language. CLI templating allows variables to be placed into a CLI configuration, which Ansible then replaces during playbook execution. Version 2.3 introduces support for NETCONF as well as persistent connections. Ansible is working on developing modules that use the NETCONF interface to provide declarative-type syntax, which is superior to CLI templating. Ansible is very committed to developing its network automation capabilities, and expects that by developing its network automation capabilities, and expects that by version 2.5 it will be a robust network management solution.

OpenFlow 1.3

What is OpenFlow?

A OpenFlow is a programmable network protocol designed to manage and direct traffic among routers and switches. It is a communications protocol that gives access to the forwarding plane of a network switch or router over the network. OpenFlow enables network controllers to determine the path of network packets across a network of switches. An OpenFlow switch separates the data path from the control path.

What version of OpenFlow is supported on Cisco Catalyst switches?

A A subset of OpenFlow 1.3, supporting the Faucet controller, is supported on Cisco Catalyst switches.

What is the Faucet controller?

A Faucet is an OpenFlow controller for multitable OpenFlow 1.3 switches (including optional table features). It implements Layer 2 switching, VLANs, Access Control Lists (ACLs), and Layer 3 IPv4 and IPv6 routing, static and via Border Gateway Protocol (BGP). The OpenFlow switch is deployed as a drop-in replacement for a Layer 2/Layer 3 switch in the network to enable extra Software-Defined Networking (SDN)-based functionality.

Are any controllers other than Faucet supported?

A Any OpenFlow controller should work, but only Faucet-related features are supported.

How secure is the connection between controller and switch?

A Transport Layer Security (TLS) can be used to secure the connection between controller and switch. TLS is a protocol that provides data encryption and authentication between applications and servers in scenarios in which the data is being sent across an insecure network.

Cisco Network Plug and Play

What is Plug and Play?

A Cisco Network Plug and Play is a day-0 bootstrap solution that provides a simple, secure, unified, and integrated offering for enterprise network customers to ease new branch or campus device rollouts or for provisioning updates to an existing network. Simply plug a device (such as a router, switch, or access point)

into the network. It discovers the controller and the automated process begins. A few minutes later, the device is upgraded and/or configured and operational.

Q

A

Does Network Plug and Play require APIC-EM?

Yes, Network Plug and Play is an application that runs on the Cisco Application Policy Infrastructure Controller Enterprise Module (APIC-EM). It is designed to allow you to connect a new device to the network given a config file and image file, without manual intervention.

Q

A

How do Network Plug and Play-enabled devices discover the APIC-EM?

Devices can automatically discover the APIC-EM through Dynamic Host Configuration Protocol (DHCP), DNS, a proxy server, or the cloud through Plug and Play Connect.

Q

A

Can Network Plug and Play-enabled devices be configured behind a DMZ?

Yes; this involves configuring a generic HTTP proxy or a VPN link to the network operations center so that the Cisco Plug and Play IOS Agent in devices at remote sites can communicate with the Cisco Network Plug and Play application. Refer to the Solution Guide for Cisco Network Plug and Play for details on setting up an HTTP proxy:

https://www.cisco.com/c/en/us/td/docs/solutions/Enterprise/Plug-and-Play/solution/guidexml/b_pnp-solution-guide.html#con_134707

Q

A

Does Network Plug and Play support authentication?

Yes, Cisco network devices support secure device identification and authentication using a Secure Unique Device Identifier (SUDI) certificate that is factory installed in the device hardware. The device sends this SUDI certificate to the APIC-EM during the SSL handshake. You can specify that the APIC-EM must validate the SUDI certificate to authenticate the device. For a list of devices that support SUDI authentication, go to:

https://www.cisco.com/c/en/us/td/docs/solutions/Enterprise/Plug-and-Play/release/notes/pnp-release-notes14.html#pgfId-151372

Zero-Touch Provisioning

Q

A

What is Zero-Touch Provisioning (ZTP)?

ZTP is an “open” bootstrap interface to automate provisioning of Cisco IOS XE devices. After the OS boots, the network device being provisioned receives a DHCP address with an option pointing it to a script server. The device then downloads a script from the server that, in turn, can be used to download a new OS image or device configuration. The script, image, and configuration files can all reside on the same server.

Q

A

How does ZTP differ from Network Plug and Play?

ZTP provides “open” bootstrap interfaces that allow customers to integrate automated provisioning into existing workflows. Network Plug and Play is a Cisco turnkey solution that requires the customer to first provision the APIC-EM in their environment.

Q

A

Why would a customer choose ZTP versus Network Plug and Play?

Network Plug and Play offers a more secure provisioning method and the most robust feature set. However, it is supported only on Cisco IOS XE devices. ZTP is ideal for customers who want to support a heterogeneous, multivendor network environment, or a network environment running multiple Cisco operating systems (Cisco IOS XE, IOS XR, and NX-OS).

PXE

Q

What is Preboot eXecution Environment (PXE)?

A

PXE is a day-0 provisioning technology that provides an open network bootstrap interface on a network device. PXE is an industry standard that has been used for servers for years.

Q

How does PXE differ from Network Plug and Play and other day-0 technologies?

A

Network Plug and Play and ZTP require the device to boot to a fully functional image, and then they begin the provisioning process. PXE begins the day-0 provisioning process from the bootloader, without loading an image.

Q

Why would anyone use PXE versus other day-0 technologies?

A

PXE is preferred by a small subset of customers who are already using PXE for server management.

NETCONF

Q

What is NETCONF?

A

NETCONF or Network Configuration Protocol (RFC 6241) is a network management protocol developed and standardized by the Internet Engineering Task Force (IETF). The NETCONF protocol defines a simple mechanism through which a network device can be managed, configuration data can be retrieved, and new configuration data can be uploaded and manipulated. The NETCONF protocol uses Remote Procedure Calls (RPCs) for its paradigm. A client encodes an RPC in XML and sends it to a server using a secure, connection-oriented session (such as Secure Shell Protocol [SSH]). The server responds with a reply encoded in XML. The key part of this mechanism is the request, and both the request and the response are fully described in an agreed-upon communication model, meaning that both parties understand the syntax that is being exchanged.

Please refer to the link below (RFC 6241) for more information on NETCONF: https://tools.ietf.org/html/rfc6241

Q

How is NETCONF related to YANG?

A

The YANG data modeling language (RFC 6020) has been developed for specifying NETCONF data models and protocol operations.

Q

What protocol operations are available with NETCONF?

A

The NETCONF protocol provides a small set of low-level operations to manage device configurations and retrieve device state information. The base protocol provides operations to retrieve, configure, copy, and delete configuration datastores. Additional operations are provided, based on the capabilities advertised by the device.

The base protocol includes the following operations:

- get
- get-config
- edit-config
- copy-config
- delete-config
- lock
- unlock
- close-session
- kill-session

Is the Cisco IOS XE NETCONF protocol open and standards compliant?

A

Yes, Cisco IOS XE NETCONF is compliant with the following standards:

RFC 6241: Network Configuration Protocol (NETCONF)

RFC 6242: Using the NETCONF Protocol over Secure Shell (SSH)

RFC 6243: With-defaults Capability for NETCONF

RFC 6020: YANG – A Data Modeling Language for the Network Configuration Protocol (NETCONF)

RFC 6022: YANG Module for NETCONF Monitoring

RFC 6991: Common YANG Data Types

RFC 5277: NETCONF Event Notifications

draft-ietf-netconf-rfc5277-bis: Notification

RFC 6470: Network Configuration Protocol (NETCONF) Base Notifications

RFC 7950: The YANG 1.1 Data Modeling language

RCF 7951: JSON Encoding of Data Modeled with YANG

RFC 7223: A YANG Data Model for Interface Management

RFC 7277: A YANG Data Model for IP Management

RFC 7224: IANA Interface Type YANG Module

Does Cisco IOS XE support NETCONF Candidate Configuration Capability?

A

Not at this time. NETCONF Candidate Configuration Capability is a Cisco IOS XE roadmap item. Applications can use the cisco-ia.yang data model, which provides “config/replace” functionality, as an alternative to NETCONF Candidate Configuration Capability.

What port does NETCONF use?

A

NETCONF runs RPCs over SSH on port 830.

What port does NETCONF use?

A

NETCONF runs RPCs over SSH on port 830.

Which Cisco IOS XE platforms support NETCONF?

A

NETCONF is supported on the following platforms:

- Cisco Catalyst 3650 Series
- Cisco Catalyst 3850 Series
- Cisco Catalyst 4500 Series
- Cisco Catalyst 9300 Series
- Cisco Catalyst 9400 Series
- Cisco Catalyst 9500 Series
- Cisco ASR 1000 Series
- Cisco 4000 Series ISRs
- Cisco Integrated Services Virtual Router (ISRv)
- Cisco Cloud Services Router (CSR) 1000v

Are there any plans to support NETCONF on any other current Cisco Catalyst platforms?

A

No, there are no plans to add NETCONF support to any other current Cisco Catalyst platforms.

RESTCONF

What is RESTCONF?

RESTCONF (RFC 8040) is a network management protocol that uses HTTP methods to provide create, read, update, and delete operations for YANG data models.

How is RESTCONF related to NETCONF?

NETCONF and RESTCONF are alternative protocols for performing operations on YANG data models. NETCONF is used by network management systems and SDN controllers to integrate with network devices. RESTCONF is used for more ad hoc network device integration.

How do RESTCONF and NETCONF protocol operations compare?

RESTCONF uses HTTP methods to identify the create, read, update, and delete operations requested for a particular resource. The following table shows how RESTCONF operations relate to NETCONF protocol operations.

RESTCONF	NETCONF
OPTIONS	none
HEAD	<get-config>, <get>
GET	<get-config>, <get>
POST	<edit-config> (nc:operation="create")
POST	invoke an RPC operation
PUT	<copy-config> (PUT on datastore)
PUT	<edit-config> (nc:operation="create/replace")
PATCH	<edit-config> (nc:operation depends on PATCH content)
DELETE	<edit-config> (nc:operation="delete")

What platforms support RESTCONF?

RESTCONF is supported on the following platforms (as of Cisco IOS XE 16.6):

- ASR 1000 Series
- 4000 Series ISRs
- ISRv
- CSR 1000v

RESTCONF is planned to be added to the Cisco Catalyst 3650, 3850 and 9000 platforms in a future Cisco IOS XE release. There are no plans to add RESTCONF support to other Cisco Catalyst platforms.

YANG data models

What is a data model?

A data model describes how data is represented and accessed. It explicitly and precisely determines the structure, syntax, and semantics of the data.

What is YANG?

YANG (RFC 7950) is a data modeling language used to model configuration and operational state data manipulated by NETCONF, RESTCONF, and other network configuration protocols. YANG is a modular language representing data structures in a tree format. YANG can be used to define the format of event notifications emitted by network elements, and it allows data modelers to define the signature of remote procedure calls that can be invoked on network elements via the NETCONF protocol. The data modeling language comes with a number of built-in data types. Additional application-specific data types can be derived from the built-in data types. Because YANG is protocol independent, it can be converted into any encoding format, such as XML, JSON, GBP, etc., that the network configuration protocol supports.

What is an “open” YANG data model?

An open YANG data model is one defined by an ecosystem (standards body or industry organization) such as IETF, IEEE, ITU, Metro-Ethernet Forum, CableLabs, and OpenConfig. Network device vendors deliver “open” models on their platforms that are compliant with the definition.

Does Cisco IOS XE support open YANG data models?

Yes, Cisco IOS XE supports open YANG models across multiple ecosystems, with nearer-term focus on the OpenConfig, IETF, and CableLabs® (DOCSIS®) defined models.

What is OpenConfig?

OpenConfig is an informal working group of network operators sharing the goal of moving our networks toward a more dynamic, programmable infrastructure by adopting software-defined networking principles such as declarative configuration and model-driven management and operations.

What is a “native” YANG data model?

A native YANG data model is one that exposes the features and functionality unique to a specific vendor’s network device or operating system.

Does Cisco IOS XE support native YANG data models?

Yes, Cisco IOS XE supports native models for the most frequently used features.

Does Cisco IOS XE have 100 percent of its features covered by native YANG data models?

No, native models cover only the most frequently used Cisco IOS XE features. All new Cisco IOS XE features are intended to be 100 percent modeled. Given the nearly 30-year history of Cisco IOS Software, there are lots of features we have chosen not to model. If customers are using some of these more obscure features, we can model them and update them on a device outside of the standard Cisco IOS XE software update process in a new “model package” (using the in-service model update feature).

How does an application determine which data models are supported on a particular device?

NETCONF and RESTCONF enabled devices advertise the capabilities (that is, the YANG data models supported on that device) during an initial “hello” message and capabilities exchange.

How do humans determine which models are supported on a particular device?

Cisco is publishing the YANG data models supported on each network OS, by release and platform, on an open YANG Model GitHub site. Note: XML “compatibility” files detail specific models supported on each platform. Alternatively, the YANG catalog provides a way to search YANG data models based on capabilities.

Is there an easy way to understand the capabilities of a YANG data model?

Yes, a number of open source tools are available, such as pYANG and YANG-explorer, that allow you to view the tree structure of a specific YANG data model.

Do you support role-based access for different YANG data models?

No. Role-based access for models is not supported.

Are APIs based on YANG models available?

Yes, Cisco publishes Python and C++ model-based APIs in a YANG Development Kit (YDK) repository.

Data path manipulation APIs

Q

What is a data path manipulation API?

A

This is an API used to manipulate packets in an automated manner. It will modify the packet header such that the forwarding of the packet can be altered. This was the top use case of the onePK Data Path service set.

Q

Is a YANG data model available?

A

Yes, Cisco-IOS-XE-Service-Insertion.yang model has been added in 16.5.1, but it supports only routing.

C67-739503-00 08/17