CSAU (INTRODUCING AUTOMATION FOR CISCO SOLUTIONS) 1.0

Objetivo

After taking this course, you should be able to: â□¢ Articulate the role network automation and programmability play in the context of end-to-end network management and operations; â□¢ Define and differentiate between waterfall and agile software development methodologies; â□¢ Interpret and troubleshoot Python scripts with fundamental programming constructs built for network automation use cases; â□¢ Describe how DevOps principles, tools, and pipelines can be applied to network operations; â□¢ Understand the role of network automation development environments and associated technologies such as Python virtual environments, Vagrant, and Docker; â□¢ Understand and construct HTTP-based API calls to network devices; â□¢ Articulate the differences among and common use cases for XML, JSON, YAML, and protobuf; â□¢ Construct and interpret Python scripts using the Python requests module to automate devices that have HTTP-based APIs; â□¢ Understand the role YANG plays in network automation; â□¢ Understand that a number of tools exist to simplify working with YANG models; â□¢ Describe the functionality of RESTCONF and NETCONF and the differences between them; â□¢ Construct Ansible playbooks to configure network devices and retrieve operational state data from them; â□¢ Build Jinja2 templates and YAML data structures to generate desired state configurations.

PÃºblico Alvo

Professionals interested in knowing and implementing solutions using the automation tools. This course is a automation introduction and pre-requesite for these trainings: â□¢ Implementing Automation for Cisco Enterprise Solutions (ENAUI); â□¢ Implementing Automation for Cisco Data Center Solutions (DCAUI); â□¢ Implementing Automation for Cisco Security Solutions (SAUI); â□¢ Implementing Automation for Cisco Service Provider Solutions (SPAUI); â□¢ Implementing Automation for Cisco Collaboration Solutions (CLAUI).

PrÃ©-Requisitos

Before taking this course, you should have the following knowledge and skills: â□¢ Routing and switching including Open Shortest Path First (OSPF), Border Gateway Protocol (BGP), and basic configuration features such as interfaces, Simple Network Management Protocol (SNMP), and static routes; â□¢ Fundamentals of Python data structures and programming constructs such as loops, conditionals, and classes, or the equivalent of 3â□□6 months of experience writing Python scripts; â□¢ Basic Linux commands for navigating the file system and executing scripts; â□¢ Knowledge of working with text editors.

Carga HorÃ¡ria

16 horas (2 dias).

ConteÃºdo ProgramÃitico

Course Introduction
Course Outline
Course Goals & Objectives

Examining Network Management and Operations
Understand the underlying problem statement with regards to network management and operations
Network Automation and Programmability
Understand that Cisco has enabled nearly every core platform with APIs for modern network operations
Network Automation Use Cases
Articulate key use cases for network automation and understand that automation is greater than deploying configurations faster
Multidomain Network Automation
Comprehend the breadth and depth of APIs across Cisco technologies and the value that provides to enhanced network automations
Technology Domains: Understand what is possible and that there are different types of scripts that can be used for network automation

Exploring Software Development Methodologies
Define and differentiate between Waterfall and Agile software development methodologies
Impact of Software
Understand the impact of software
Waterfall Development Process
Understand the Waterfall software development methodology
Agile Methodology
Understand the Agile software development methodology and how Lean is applied to implement Agile

Using Python for Network Automation
Interpret and troubleshoot Python scripts with fundamental programming constructs built for network automation use cases
Python Fundamentals
Interpret basic scripts using constructs such as data structures, conditional blocks, and loops.
Enforce Python Fundamentals on the Interactive Interpreter
Comprehend common Python constructs such as data structures, loops, conditional statements.
Topology
Network Libraries
Compare and contrast Python modules and packages.
Python Package Management
Articulate the role of PyPI and pip along with how to install Python applications from source.
Understand how the Netmiko Python library can be used to automate network devices over SSH.
Inside Python Modules
Understand how Python modules enable modular development.
Automate Networks with Netmiko
Understand how to install a Python package and create a script to perform basic network automation using SSH with the Netmiko library.
Troubleshoot Python Scripts
Interpret and troubleshoot Python scripts that use nested data structures to perform network automation tasks

Describing NetDevOps: DevOps for Networking
Describe how DevOps principles, tools, and pipelines can be applied to network operations
Development and Operations
Describe the role and principles of DevOps within application development
Exploring DevOps Tools
Understand that DevOps tools go far beyond configuration management tools
Git Fundamentals
Articulate the value of using version control
Git Branches
Understand git-based workflow(s) that enable collaboration between internal and external teams.
Merge Conflicts
Understand how to resolve a merge conflict using git Continuous Integration
Describe the workflow of a CI/CD pipeline and applicability to network operations
Use the Git Version-Control System and Collaborate on an Internal Project
Understand how to use the most common git commands, collaborate with others on an internal project using branches, and pushing changes back to a repository
Practical Manage Merge Conflicts
Learn how to resolve merge conflicts when collaborating using a git version control system.

Managing Automation Development Environments
Understand the role of network automation development environments and associated technologies such as Python virtual environments, Vagrant, and Docker
Need for Reproducible Development Environments
Articulate the problem that causes the need for reproduceable development environments
Python Virtual Environments
Describe the benefits and how to use Python virtual environments
Vagrant
Articulate the role Vagrant can play to re-build development environments
Docker
Articulate the role Docker can play to re-build development environments
Build Reproduceable Automation Environments
Create reproducible automation environments Python virtual environments, Vagrant, and Docker.

Introducing HTTP Network APIs
Understand and construct HTTP-Based APIs calls to network devices
API Overview
Describe the key components and characteristics of an API
HTTP-Based APIs
Define the common HTTP request types and response codes when using HTTP-Based APIs
RESTful APIs vs. Non-RESTful APIs
Articulate common principles of RESTful APIs and identify if an API is RESTful
HTTP-Based Authentication
Describe common ways to authenticate to an HTTP-based API
Postman
Describe the value and purpose of Postman
Practical Use HTTP-Based APIs with Postman
Practical Use Postman to interact with HTTP-Based APIs
Streaming Telemetry

Compare and contrast streaming telemetry with traditional mechanisms such as SNMP

Reviewing Data Formats and Data Encoding
Articulate the difference and common use cases for XML, JSON, YAML, and protobuf
JavaScript Object Notation
Comprehend how JSON is used and how it relates to a Python dictionary.
Extensible Markup Language
Understand how XML is used.
gRPC and Protobuf
Describe the relationship between gRPC and protobuf
YAML Data Serialization Standard
Describe how YAML relates to JSON and common use cases for YAML.
Practical Use: Explore YAML and JSON Data
Understand the relationship between YAML, JSON, and Python dictionaries.

Using Python Requests to Automate HTTP-Based APIs
Construct and interpret Python scripts using the Python requests module to automate devices that have HTTPBased
APIs
Python Requests Overview
Make an API call to a network device using Python requests
HTTP Authentication
Describe common HTTP authentication mechanisms
Practical Use: Consume HTTP-Based APIs with Python Requests
Learn to use Python to automate devices with HTTP-Based APIs using the Python requests module.

Exploring YANG
Understand the role YANG plays within a network automation context
Introduction to YANG
Describe the function and role of a YANG model.
Types of YANG Models
Compare functionality, benefits, and uses of OpenConfig, IETF, and native YANG models

Using YANG Tools
Understand that a number of tools exist to simplify working with YANG models
YANG Tool Types
Understand that a number of tools exist to simplify working with YANG models
Pyang
Interpret a YANG module ASCII generated tree
YANG Explorer
Understand how to browse and visualize YANG models and YANG-based data
Practical Use: Explore YANG Tools
Navigate and Interpret YANG Models

Automating Model-Driven APIs with Python
Describe the functionality of RESTCONF and NETCONF and the differences between them
NETCONF Overview
Describe the benefits and functionality of NETCONF

Python ncclient
Interpret a Python script that uses ncclient.
RESTCONF Overview
Describe the functionality and benefits of RESTCONF
Practical Use: Explore RESTCONF with Python
Automate Network devices using RESTCONF with different YANG models using Python requests
Practical Use: Explore NETCONF with Python
Automate Network devices using NETCONF with different YANG models using the Python ncclient library.
Introducing Ansible for Network Automation

Introducing Ansible for Network Automation
Construct Ansible playbooks to configure network devices and retrieve operational state data from network devices
Configuration Management Tools
Describe the difference between agent and agentless configuration management tools
Cisco Ansible Integrations
Introduction to Ansible
Describe key Ansible terminology
Ansible Inventory File
Interpret basic Ansible inventory files, group, and host variables
Use the Cisco IOS Core Configuration Module
Construct an Ansible playbook that will configure Cisco network devices
Ansible Documentation
Ensure users know how to explore module documentation, browse, and navigate the official Ansible docs
Ansible Documentation Utility
Practical Use: Configure Network Devices with Ansible
Perform automated configuration management using Ansible.
Gather Cisco IOS Device Facts
Describe Ansible facts and how they’re used in a playbook
Use the Cisco IOS Core Command Module
Construct an Ansible playbook that will retrieve data from Cisco network devices
PracticaL Use: Collect Network Data with Ansible
Perform read-only automation with Ansible collecting facts and show commands and saving them tofiles.

Templating Configurations with Jinja2
Build Jinja2 templates and YAML data structures to generate desired state configurations
Jinja2 Overview
Describe the role of Jinja templating for network automation in Ansible
Basic YAML
Learn how to represent and model network data as YAML
Configuration Templating with Ansible
Construct and use Jinja2 templates within Ansible
Practical Use: Build and Deploy Configurations with Ansible
Construct Jinja2 templates and YAML data structures and auto-generate configurations with Ansible.

Lab outline
Lab 1: Use Network Automation Scripts
Lab 2: Enforce Python Fundamentals on the Interactive Interpreter

Lab 3: Automate Networks with Netmiko
Lab 4: Troubleshoot Python Scripts
Lab 5: Use the Git Version-Control System and Collaborate on an Internal Project
Lab 6: Manage Merge Conflicts
Lab 7: Build Reproduceable Automation Environments
Lab 8: Use HTTP-Based APIs with Postman
Lab 9: Explore YAML and JSON Data
Lab 10: Consume HTTP-Based APIs with Python Requests
Lab 11: Explore YANG Tools
Lab 12: Explore NETCONF with Python
Lab 13: Explore RESTCONF with Python
Lab 14: Configure Network Devices with Ansible
Lab 15: Collect Network Data with Ansible
Lab 16: Build and Deploy Configurations with Ansible