Automating and Programming Cisco Service Provider Solutions v1.0 (300-535)

Exam Description: Automating and Programming Cisco Service Provider Solutions v1.0 (SPAUTO 300-535) is a 90-minute exam associated with the CCNP Service Provider Certification and DevNet Professional Certification. This exam tests a candidate's knowledge of implementing service provider automated solutions, including programming concepts, orchestration, programming OS, and automation tools. The course, Implementing Cisco Service Provider Automation Solutions, helps candidates to prepare for this exam.

The following topics are general guidelines for the content likely to be included on the exam. However, other related topics may also appear on any specific delivery of the exam. To better reflect the contents of the exam and for clarity purposes, the guidelines below may change at any time without notice.

10% 1.0 Network Programmability Foundation

1.1 Utilize common version control operations with git (add, clone, push, commit, diff, branching, and merging conflict)

1.2 Describe characteristics of API styles (REST and RPC)

1.3 Describe the challenges encountered and patterns used when consuming APIs synchronously and asynchronously

1.4 Interpret Python scripts containing data types, functions, classes, conditions, and looping

1.5 Describe the benefits of Python virtual environments

1.6 Explain the benefits of using network configuration tools such as Ansible and Puppet for automating IOS XE or IOS XR platforms

30% 2.0 Automation APIs and Protocols

2.1 Describe the characteristics and use of YANG Data Models (OpenConfig, IETF, and Vendor)

2.2 Describe common HTTP authentication mechanisms (basic, token, and oauth)

2.3 Compare common data types (JSON, XML, YAML, plain text, gRPC, and protobuf)

2.4 Identify the JSON instance based on a YANG model

2.5 Identify the XML instance based on a YANG model

2.6 Interpret a YANG module tree generated by pyang

2.7 Implement configuration and operation management using RESTCONF protocol

2.8 Implement configuration and operation management using NETCONF protocol

2.9 Compare the NETCONF datastores

30% 3.0 Network Device Programmability

3.1 Deploy device configuration and validate operational state using ncclient

3.2 Construct a Python script using NETCONF with YDK

3.3 Deploy device configuration and validate operational state using NetMiko

3.4 Deploy device configuration and validate operational state using Ansible playbooks

3.5 Compare gNMI with NETCONF

3.6 Construct a Python script using RESTCONF with JSON
3.7 Construct Xpath notation for a given node or instance of a node
3.8 Diagnose model-driven dial-in/-out telemetry streams with gRPC for a Cisco IOS XR

30% 4.0 Automation and Orchestration Platforms

4.1 Describe ETSI NFV
4.2 Describe NSO architecture
4.3 Identify the benefits of NSO
4.4 Construct a Python script to configure a device using NSO RESTCONF API
4.5 Describe the management and automation of Cisco ESC components
4.6 Implement XR traffic controller (including topology information transfer to XTC)
4.7 Identify the uses of Cisco WAE
4.8 Construct a service template using NSO
4.9 Deploy a service package using NSO