Automating Cisco Data Center Networking Solutions v2.0 (300-635)

Exam Description: Automating Cisco Data Center Networking Solutions v2.0 (DCNAUTO 300-635) is a 90-minute exam associated with the CCNP Data Center Certification and CCNP Automation Certification. This exam certifies a candidate's knowledge of implementing Data Center automated solutions, including infrastructure as code, network element programmability, operations, and AI in automation. Technologies included are Cisco NX-OS, Cisco Nexus Dashboard, Cisco ACI (network-centric mode) and Cisco Nexus Hyberfabric. The course, Automating Cisco Data Center Networking 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.

15% 1.0 Network Automation Foundation

- 1.1 Describe OpenConfig, IETF, and native YANG models
- 1.2 Describe ACI-based network-centric mode including objects such as EPG, bridge domains, contracts, and VRFs
- 1.3 Describe DPUs in data center network switches
- 1.4 Describe NETCONF, gNMI, gRPC, and gNOI
- 1.5 Construct a gRPC payload based on a YANG module using tools such as YANG Suite and pyang

25% 2.0 Infrastructure as Code

- 2.1 Describe infrastructure as code (IaC) and GitOps
- 2.2 Construct network configuration templates with Jinja2 using features such as loops, conditionals, output modifiers, and filters
- 2.3 Construct an Ansible playbook with controller and device collections
- 2.4 Construct a Terraform plan with controller and device providers
- 2.5 Troubleshoot network automation solutions based on Ansible and Terraform

25% 3.0 Network Element Programmability

- 3.1 Construct a network automation solution with Python using ncclient to manage and monitor configurations
- 3.2 Construct a device-level network automation solution for Day-0 provisioning with POAP
- 3.3 Implement on-box programmability and automation with NX-OS using
  - 3.3.a Bash
  - 3.3.b Python Scripting
- 3.4 Describe the use of templates and policies in Nexus Dashboard
- 3.5 Construct network configuration templates with Nexus Dashboard

	3.6	Describe capabilities and features of NX-API
25%	**4.0**	**Operations**
	4.1	Describe use of network topology simulation related to data center operations
	4.2	Implement change validation for a network automation solution using pyATS CLI tools
	4.3	Describe architectural components of model-driven telemetry
	4.4	Configure a subscription for model-driven telemetry on NX-OS devices (gNMI and gRPC)
	4.5	Integrate a network automation solution with a network source of truth
	4.6	Construct a Python script that retrieves network health data from NX-OS devices via CLI and Nexus Dashboard
	4.7	Troubleshoot packet flows for containerized workloads on Linux hosts considering VLANs, veth, bond interfaces, subinterfaces, and bridges
10%	**5.0**	**AI in Automation**
	5.1	Describe AI-assisted code development for network automation
	5.2	Describe the security risks in a given AI-based network automation solution
	5.3	Describe the integration of network devices, controllers, and management platforms with AI agents