

Développement d'applications et automatisation des flux de travail avec les plateformes Cisco (DEVASC 200-901)

Date et durée
Code formation : DEVASC Durée : 5 jours Nombre d'heures : 35 heures
Formation avec préparation à la certification
Description
La formation DEVASC vous prépare à la certification Cisco® DevNet Associate et vous permet également d'acquérir des compétences en automatisation de réseau. Elle vous initie au développement d'applications réseau avec les plateformes Cisco et à la mise en œuvre de flux d'automatisation sur les infrastructures réseau, de sécurité, de collaboration et de systèmes d'information. Ce programme de 5 jours comprend des cours théoriques et pratiques sur la résolution des problèmes liés à l'utilisation des interfaces de programmation d'applications (API) de Cisco et sur le fonctionnement des outils de développement les plus récents. L'objectif de cette formation est de vous préparer à l'examen officiel Cisco 200-901 DEVASC. La réussite de celui-ci vous permettra d'obtenir le titre de Cisco Certified DevNet Associate.
Objectifs
En suivant la formation Cisco DevNet Associate, vous validerez les objectifs de compétence suivants : • comprendre l'importance des APIs et la mise en œuvre d'outils de contrôle de version pour le développement de logiciels modernes ; • expliquer les processus de développement logiciel et les bonnes pratiques ; • connaître les options de structuration et de conception d'un logiciel modulaire ; • comprendre les principes du protocole HTTP et son intégration dans les APIs en réseau ; • appliquer les concepts REST aux API basées sur le protocole HTTP ; • découvrir et décrire le fonctionnement des plateformes Cisco ; • expliquer les concepts de base des réseaux et comprendre la topologie d'un réseau simple ; • décrire les interactions entre les applications et le réseau ainsi que les outils utilisés pour résoudre les problèmes ; • appliquer les concepts de programmation basée sur des modèles pour automatiser des tâches courantes à l'aide de scripts Python ; • identifier les modèles de déploiement d'applications et les composants communs dans le pipeline de développement ; • comprendre les problèmes de sécurité et les types de tests les plus courants ; • utiliser la conteneurisation pour le développement en local ; • utiliser des outils pour automatiser une infrastructure en utilisant des scripts et un modèle de programmation orienté ;

- être bien préparé à l'examen Cisco 200-301 DEVASC afin d'obtenir la certification Cisco Certified DevNet Associate.

Points forts

Un formateur expert et certifié Cisco, des support de cours officiel avec des labs et une préparation à l'examen Cisco DEVASC 200-901.

Modalités d'évaluation

Travaux Pratiques

Pré-requis

Suivre la **formation Cisco DevNet Associate** nécessite les prérequis suivants :

- avoir des compétences de base en informatique, à savoir l'utilisation d'un système d'exploitation et la navigation sur Internet ;
- avoir une expérience pratique en programmation, notamment avec le langage Python.

Public

Cette formation s'adresse aux publics suivants :

- les ingénieurs en automatisation de réseaux, les développeurs logiciels, les programmeurs d'intégration de systèmes, les architectes d'infrastructure et les concepteurs de réseaux.

Cette formation s'adresse aux profils suivants

Ingénieur réseaux - télécoms

Développeur

Architecte informatique / SI

Programme

Domaine 1 : le développement et la conception de logiciels

- Comparaison entre les formats de données (XML, JSON et YAML).
- Analyse des formats de données en utilisant des structures de données Python.
- Présentation du concept de développement piloté par les tests.
- Comparaison des méthodes de développement de logiciels (Agile, Lean et Waterfall).
- Présentation des avantages de l'organisation du code par méthodes, fonctions, classes et modules.
- Présentation des avantages de l'utilisation de modèles de conception communs (MVC et Observer).
- Présentation des avantages de la mise en place d'un système de contrôle des versions.
- Utilisation des opérations de contrôle de version communes avec la plateforme Git.

Domaine 2 : la compréhension et l'utilisation des APIs

- Création d'une requête API REST afin de réaliser une tâche définie dans la documentation de l'API.
- Présentation des schémas d'utilisation communs relatifs aux Webhooks.
- Identification des exigences liées à l'utilisation des APIs.
- Analyse des codes de réponse HTTP communs relatifs aux API REST.
- Résolution de problème à partir d'un code de réponse HTTP, d'une requête et de la documentation de l'API.

- Identification des éléments d'une réponse HTTP (code de réponse, en-têtes et corps).
- Utilisation des méthodes d'authentification communes des APIs : (token de base, token personnalisé et clés d'APIs).
- Comparaison des différents types d'API (REST, RPC, synchrone et asynchrone).
- Création d'un script Python qui appelle une API REST via une bibliothèque de requêtes.

Domaine 3 : les plateformes Cisco et le développement

- Création d'un script Python exploitant un SDK Cisco à partir de la documentation fournie par ce dernier.
- Présentation des fonctions des plates-formes de gestion de réseau et des API de Cisco (Meraki, Cisco DNA Center, ACI, Cisco SD-WAN et NSO).
- Présentation des fonctions des plateformes de gestion de calcul et des APIs de Cisco (UCS Manager, UCS Director et Intersight).
- Présentation des fonctions des plateformes de collaboration et des APIs de Cisco (Webex Teams, Webex devices, Cisco Unified Communication Manager).
- Présentation des fonctions des plateformes et des API de sécurité de Cisco (Firepower, Umbrella, AMP, ISE et ThreatGrid).
- Explication du fonctionnement des APIs au niveau des périphériques et des interfaces dynamiques pour les systèmes IOS XE et NX-OS.
- Identification des ressources DevNet adaptées à un scénario donné (Sandbox, Code Exchange, support, forums, Learning Labs, et documentation API).
- Application des concepts de programmation orientée vers les modèles (YANG, RESTCONF et NETCONF) dans un environnement Cisco.
- Création de code pour réaliser une opération spécifique sur la base d'un ensemble de critères et d'une documentation de référence de l'API.

Domaine 4 : le déploiement et la sécurité des applications

- Présentation des avantages de l'edge computing (informatique en périphérie).
- Identification des caractéristiques relatives aux modèles de déploiement d'applications (cloud privé, cloud public, cloud hybride et edge computing).
- Identifier les caractéristiques de ces types de déploiement d'applications.
- Identification des composants d'un pipeline CI/CD dans les déploiements d'applications.
- Conception d'un test unitaire Python.
- Compréhension du contenu d'un fichier Docker.
- Utilisation des images Docker pour le développement en local.
- Identification des problèmes de sécurité des applications liés à la protection des secrets, au cryptage et au traitement des données.
- Explication du fonctionnement du pare-feu, du DNS, de l'équilibreur de charge et du proxy inverse dans le déploiement d'une application.
- Présentation des menaces majeures selon l'OWASP notamment les attaques de type XSS, les injections SQL et les attaques de type CSRF.
- Utilisation des commandes Bash (gestion des fichiers, navigation dans les répertoires et variables d'environnement).
- Identification des principes de DevOps.

Domaine 5 : l'infrastructure et l'automatisation

- Présentation des avantages de la programmation basée sur un modèle pour l'automatisation de l'infrastructure.
- Comparaison entre la gestion au niveau du contrôleur et la gestion au niveau de la machine.
- Présentation des outils de simulation et des outils de test du réseau (VIRL et pyATS).
- Identification des composants et des avantages du pipeline CI/CD dans l'automatisation de l'infrastructure.
- Compréhension des principes de l'infrastructure en tant que code.

- Présentation des outils d'automatisation Ansible, Puppet, Chef et Cisco NSO.
- Identification du flux de travail automatisé avec un script Python exploitant les APIs Cisco (ACI, Meraki, Cisco DNA Center ou RESTCONF).
- Identification du flux de travail automatisé avec un playbook Ansible (gestion des paquets, gestion des utilisateurs liés aux services, configuration des services de base et démarrage/arrêt).
- Identification du flux de travail automatisé avec un script bash (gestion des fichiers, installation d'applications, gestion des utilisateurs, navigation dans les répertoires, etc.).
- Interprétation des résultats d'une requête RESTCONF ou NETCONF.
- Interprétation des modèles YANG fondamentaux.
- Interprétation de la notion de unified diff.
- Présentation des principes et des avantages d'un processus de revue de code.
- Interprétation d'un diagramme de séquence comprenant des appels d'API.

Domaine 6 : les fondamentaux du réseau

- Compréhension et utilisation des adresses MAC et des réseaux locaux virtuels (VLAN).
- Compréhension et utilisation des adresses IP, des routages, des masques de sous-réseau et des préfixes, ainsi que des passerelles.
- Compréhension et utilisation des composants réseau communs comme les switches, les routeurs, les firewalls et les répartiteurs de charge.
- Interprétation d'un diagramme de topologie de réseau basique incluant des commutateurs, des routeurs, des pare-feu, des équilibreurs de charge et des valeurs de port.
- Compréhension des fonctions des plans de gestion, des plans de données et des plans de contrôle dans un dispositif de réseau.
- Compréhension du fonctionnement des services IP suivants : DHCP, DNS, TCP/IP, NAT, SNMP et NTP.
- Identification des valeurs de port des protocoles SSH, Telnet, HTTP, HTTPS et NETCONF.
- Identification des problèmes de connectivité des applications (problème de NAT, port de transport bloqué, proxy et VPN).
- Analyse des impacts sur les applications liés aux contraintes du réseau.

Cisco est une marque déposée de Cisco Systems, Inc. aux Etats-Unis et dans d'autres pays.