



Puppet : automatiser la gestion de ses configurations

UX161

Durée: 2 jours

1 520 €

7 au 8 février
2 au 3 mai

2 au 3 septembre
9 au 10 décembre

Public :

Administrateurs systèmes, Architectes, Ingénieurs systèmes et réseaux, Chefs de projets

Objectifs :

A l'issue de la formation, le stagiaire sera capable de gérer une infrastructure de manière centralisée avec l'application Open Source Puppet.

Connaissances préalables nécessaires :

avoir les connaissances de base et une pratique de l'administration Linux

Objectifs pédagogiques :

Identifier les concepts et principes de l'automatisation
Comprendre le fonctionnement de Puppet
Savoir installer et configurer Puppet
Déployer des logiciels et leur configuration

Programme :

Concepts et principes de l'automatisation

Présentation du besoin et des fonctionnalités de Puppet.
Gestion des configurations logicielles.
Collecte et centralisation des informations.
Définition des configurations cibles, mises à jour automatiques ou manuelles.
L'orchestration dans le cas d'un cloud d'entreprise.
La gestion de configuration de machines virtuelles.

Fonctionnement de Puppet

Historique de puppet et les différentes versions de puppet.
Présentation de Puppet Enterprise.
Architecture
Principe client/serveur, modules de configuration, les agents Puppet, la console, l'outil de gestion de cloud, les manifests puppet.
Présentation des plateformes supportées comme serveur et comme client (agent puppet).
Mode opératoire : définition des configurations, vérification de l'état des clients, simulation des changements proposés par Puppet, application sur les systèmes cibles.



Phirio

Installation et configuration

Prérequis systèmes.

Atelier : Installation d'une infrastructure Puppet :serveur, base de données, agents.

Etude du fichier de configuration puppet.conf. Configuration des rôles : maître, agent, console... Gestion des certificats sur les clients.

Déploiement des logiciels et leur configuration

Le langage puppet

Introduction : présentation des manifests, modules, templates, des modules disponibles sur la forge puppet.

Les manifests : description du langage déclaratif de puppet.

Syntaxe, dépendance entre instructions. Etudes des variables, présentation des facts. Déclarations conditionnelles : if, case, selectors

Atelier : rédaction d'un exemple de manifest de création d'un fichier et modification des droits, application de ce manifest sur un poste client cible.

Etude détaillée des 'ressources' puppet :file, package, service, user, exec, notify, ...

Exemple de la ressource 'file' :différents attributs disponibles pour déterminer les caractéristiques d'un fichier, s'assurer de sa présence, ...

Exemple de la gestion de dépendance :production de manifest pour l'installation de sshd

Les classes et modules. Installation de modules.

Atelier : création d'un module pour propager la configuration de bash sur les postes clients.

Définition des templates.

Atelier : création de templates en utilisant le langage ERB.

Etude des paramètres de classes. Définition de nouveaux types de ressources.

Utilisation et génération de documentations au format puppet. Organisation du site .pp.

Bonnes pratiques

Retours d'expériences et méthodes d'organisation et de développement des scripts puppet. Présentation des patterns pour puppet.

Atelier : mise en évidence de sources d'erreurs et proposition de solutions appliquant les bonnes pratiques