

Mis à jour le 11/10/2024

S'inscrire

# Formation Puppet

3 jours (21 heures)

## Présentation

**Puppet** est un logiciel libre permettant de gérer, installer, configurer ou mettre à jour efficacement un nombre important de machines tout en s'adaptant aux spécificités de chacune d'entre elles. Une fois les configurations effectuées leur maintenance sera simple et rapide avec la possibilité d'être facilement restaurable. Puppet est livré avec de nombreux outils de gestion de code et de tests simplifiant au maximum l'utilisation de ces nombreuses fonctionnalités, tels que Hiera & r10k. Ces outils permettent également d'améliorer la clarté & la lisibilité de votre infrastructure, de garder une architecture cohérente et de faciliter le déploiement des configurations nouvellement créées sans perdre en organisation.

Puppet possède deux couches, l'une étant un langage de configuration décrivant les hôtes et les services, l'autre permettant à l'administrateur d'implémenter la configuration sur des plateformes variées telles que Windows / Linux / OSX. L'administrateur peut définir la configuration d'un service que Puppet se chargera de monitorer et d'exécuter.

Dans cette formation vous découvrirez comment rendre votre infrastructure scalable. Nous verrons comment déployer vos configurations sur toutes vos machines en maîtrisant toutes les subtilités et les différentes fonctionnalités de Puppet et Puppet Enterprise ainsi que leurs outils respectifs.

Comme toutes nos formations, celle-ci vous présentera la dernière version stable en date ([Puppet 8.7](#) à la date de l'article).

## Objectifs

- Être capable de réduire la complexité de la gestion de votre infrastructure
- Utiliser Hiera et r10k pour séparer le code de la data
- Maîtriser une architecture scalable via Puppet
- Savoir automatiser des tâches d'infrastructure avec Puppet
- Utilisez les outils & l'écosystème Puppet

## Public visé

- Développeurs, Architectes, Administrateurs système / Sysadmin

## Pré-requis

- Connaissance de base d'un système Unix / Linux, des notions de base en scripts / shell
- Avoir une bonne connaissance en administration système Windows
- Connaître les bases de la programmation orienté objet
- Avoir déjà manipulé des outils de déploiement
- [Tester Mes Connaissances](#)

## Programme sur Notre Formation Puppet Enterprise version 8 : DevOps Scalable Infrastructure

### Jour 1 - Les bases de Puppet

- Pré-requis Puppet - connaissances, habilitations, outils, ...
- Packaging - construire, intégrer, déployer des packages et des référentiels (repositories)
- Concepts IAC (Infrastructure As Code) - Code (modules puppet), Configurations (hieradata), Gestion des requêtes et workflows
- Les Produits et services de Puppet - Vue d'ensemble
- Périmètre de Puppet - KISS (Keep It Stupid and Simple) par rapport aux concepts d'ingénierie
- Interfaces utilisateur - Console Puppet, APIs, CLI

### Jour 2 - Tenir ses serveurs à jour

- Hieradata - organiser la hiérarchie des données de votre infrastructure (centraliser, hériter, sécuriser les configurations autant que vous le pouvez, puis remplacer seulement si nécessaire)
- Puppet Compilation - Comprendre le cycle de vie d'une exécution d'agent Puppet
- Puppet Développement - Créez vos propres modules
- Puppet idempotency concept - Appliquez le catalogue autant de fois que vous le souhaitez

### Jour 3 - Déployer, mettre à jour, migrer les applications distribuées sur l'infrastructure

- Développement de Puppet - faits, fonctions et types personnalisés pour améliorer vos modules développés
- Puppet Forge - Ne réinventez pas la roue !
- Puppet Development - Définir une pile d'applications avec Puppet
- Puppet Discovery - Connaissances de l'état de l'infrastructure en temps réel (bientôt obsolète)
- Comprendre les concepts : déclaratifs et impératifs
- Infrastructure - Configurez votre environnement de laboratoire (entreprise puppet, gitlab, rpmbuild, dépôt rpm)

### Jours 4 & 5 - Module Complémentaire sur demande (+2 jours)

Depending on what is possible to play with (is there a laboratory? a git server? a repository? etc), this part is dedicated to practice and will be adjusted onsite regarding participant skills and needs.

## Full Stack Deployment

- OS (Linux / Unix / Windows), Network equipments - Quick wins are better than long fails
- System administration - DNS, NTP, Users, Roles, SSH keys, Kernel parameters, Filesystems, etc
- Middleware administration - Databases (servers/clients), Web servers
- Software administration - Challenge your R&D team to get the best quality only (or let them do the hard work in production)
- Patch management - Puppet remediate

## Reporting

- Vulnerability scan (based on task and remediate)
- Change notification - mailing, HMI

## Server provisioning

- Baremetal install - Razor overview
- VM install - VMWARE ESX, VRA, VRO

////////////////////////////////////

## Langage Puppet

- Coding style
- Créer des modules
- Créer un Manifest
- Classes
- Paramètres de Classes
- Types
- Providers
- Templates

## Rôles et Profils

- Profils : les bonnes pratiques / best practices
- Utiliser le mot-clé include
- Utilisation de sous-répertoires pour des groupes
- Masquer la complexité : les paramètres, les valeurs par défaut et l'abstraction
- Décider comment définir vos paramètres pour les classes de composants
- Recherche automatique de paramètre de classe & fonction de recherche

- Rôles : les bonnes pratiques
- Include & nommage des rôles
- Décider de la granularité des rôles pour vos nœuds

## Ressources

- Types
  - k5login
  - nagios\*
  - schedule
  - interface
  - package
  - router
  - host
  - exec
  - interface ...
- ressources virtuelles et exportées
- Tags
- Load balancing
- Connexion à la base de données

## Hiera 5

- Organiser les fichiers
- Gestion des clés avec Yalm
- Utilisation de la fonction Lookup
- Debug
- Présentation de Jerakia

## Environnements

- Présentation de r10k
- Création d'environnement de développement
- Déploiement d'environnement avec r10k

## Code Manager

- Organiser son code
- Utilisation de RBAC
- Gestion du repository

## Workflow

- Création d'un Workflow
- Repositories

- Tasks
- Utilisation de PDK

## Continuous integration

- Puppet Pipelines
- Gestion des plug-ins
- Découverte de Jenkins
- Intégration de PDK
- Création de Test Unitaire avec Puppet RSpec

## Tasks et Discovery

- Installation de Bolt
- Installation de Discovery
- Gestion des sources
- Installation des agents
- Gestion des services

## Orchestration

- Création d'une application
- Gestion de la base de données
- Déployer une application

## Puppet Enterprise

- Ajout de nodes sur PE
- Classification dynamique de nœuds avec des groupes
- Orchestrer les stacks d'applications

## Détection d'erreurs et résolution : Troubleshooting & Profiling

- Attente de signature de certificat
- Réutilisation de certificat
- Paramètres de connexion
- Erreurs de syntaxe
- Ressources manquantes
- Dépendances
- Authentification

## Modules avancés : 2 jours supplémentaires

## Étendre Puppet : Extending Puppet (module optionnel)

- Custom facts & Debugging facts
- Fonctions personnalisées : Custom functions
- Types & providers
- Creating and distributing the type
- Namevar : special attribute
- Properties, optional, parameters, defaults
- Input Values : Vérification, Validate block, munge datatype
- Méthodes Exists, create et destroy
- GET & SET methods : Gérer les propriétés de type
- Implémentation de la méthode self.instances

## Tuning avancé : Scaling (module optionnel)

- Load balancing
- Estimation du nombre d'agents à supporter
- Autorités de certification
- Tuning de PuppetServer
  - Implémentation de Puppet Entreprise / Open Source
- Tuning de Puppet DB
  - Gestion des threads CPU
  - Gestion de la taille du tas / heap
- PuppetDB avec PGTune
- Paramétrage automatique

## Sociétés concernées

Cette formation s'adresse à la fois aux particuliers ainsi qu'aux entreprises, petites ou grandes, souhaitant former ses équipes à une nouvelle technologie informatique avancée ou bien à acquérir des connaissances métiers spécifiques ou des méthodes modernes.

## Positionnement à l'entrée en formation

Le positionnement à l'entrée en formation respecte les critères qualité Qualiopi. Dès son inscription définitive, l'apprenant reçoit un questionnaire d'auto-évaluation nous permettant d'apprécier son niveau estimé sur différents types de technologies, ses attentes et objectifs personnels quant à la formation à venir, dans les limites imposées par le format sélectionné. Ce questionnaire nous permet également d'anticiper certaines difficultés de connexion ou de sécurité interne en entreprise (intraentreprise ou classe virtuelle) qui pourraient être problématiques pour le suivi et le bon déroulement de la session de formation.

## Méthodes pédagogiques

Stage Pratique : 60% Pratique, 40% Théorie. Support de la formation distribué au format numérique à tous les participants.

## Organisation

Le cours alterne les apports théoriques du formateur soutenus par des exemples et des séances de réflexions, et de travail en groupe.

## Validation

À la fin de la session, un questionnaire à choix multiples permet de vérifier l'acquisition correcte des compétences.

## Sanction

Une attestation sera remise à chaque stagiaire qui aura suivi la totalité de la formation.