

# Puppet, gérer ses configurations logicielles sous Linux

**Cours Pratique de 2 jours - 14h**

**Réf : LPP - Prix 2025 : 1 930 HT**

Le prix pour les dates de sessions 2026 pourra être révisé

Ce cours vous apprendra à gérer votre infrastructure de manière centralisée avec l'application Puppet. Vous verrez comment déployer des logiciels avec la configuration choisie, maintenir à jour les systèmes et les applications ainsi qu'utiliser des classes et des modules pour répondre à vos besoins actuels et futurs.

## OBJECTIFS PÉDAGOGIQUES

À l'issue de la formation l'apprenant sera en mesure de :

Comprendre le fonctionnement de Puppet

Identifier les concepts et principes de l'automatisation

Déployer des logiciels et leur configuration

Savoir installer et configurer Puppet

## LE PROGRAMME

dernière mise à jour : 02/2024

### 1) Présentation

- Pourquoi la gestion d'infrastructure centralisée ?
- Pourquoi Puppet ?
- Les différentes versions et distributions de Puppet.
- Le fonctionnement général de Puppet.

### 2) Mise en œuvre

- Installer un Puppet master.
- Configurer un Puppet master.
- Installer et configurer l'agent Puppet.

*Travaux pratiques : Installation et configuration initiale des outils. Prise en main de l'environnement.*

### 3) Prise en main du langage de Puppet

- Les principaux types supportés par Puppet.
- La syntaxe du langage.
- Générer des fichiers à partir de templates.
- Les relations entre ressources.
- Les ressources définies.
- La remontée d'informations avec Facter.

*Travaux pratiques : Gérer les utilisateurs et les groupes. Déployer des logiciels et leur configuration. Gérer et superviser les services. Décrire la configuration actuelle avec Puppet.*

### 4) Réutilisation de modules Puppet

- Pourquoi réutiliser des modules existants ?
- Les modules, les classes.
- Utiliser un module existant.

## FINANCEMENT

Ce cours fait partie des actions collectives Atlas.

## PARTICIPANTS

Administrateurs systèmes, Architectes, Ingénieurs systèmes et réseaux, Chefs de projets intégrant une équipe qui utilise déjà Puppet ou souhaitant le mettre en œuvre dans un contexte professionnel.

## PRÉREQUIS

Avoir les connaissances de base et une pratique de l'administration Unix/Linux.

## COMPÉTENCES DU FORMATEUR

Les experts qui animent la formation sont des spécialistes des matières abordées. Ils ont été validés par nos équipes pédagogiques tant sur le plan des connaissances métiers que sur celui de la pédagogie, et ce pour chaque cours qu'ils enseignent. Ils ont au minimum cinq à dix années d'expérience dans leur domaine et occupent ou ont occupé des postes à responsabilité en entreprise.

## MODALITÉS D'ÉVALUATION

Le formateur évalue la progression pédagogique du participant tout au long de la formation au moyen de QCM, mises en situation, travaux pratiques...

Le participant complète également un test de positionnement en amont et en aval pour valider les compétences acquises.

## MOYENS PÉDAGOGIQUES ET TECHNIQUES

- Les moyens pédagogiques et les méthodes d'enseignement utilisés sont principalement : aides audiovisuelles, documentation et support de cours, exercices pratiques d'application et corrigés des exercices pour les stages pratiques, études de cas ou présentation de cas réels pour les séminaires de formation.
- À l'issue de chaque stage ou séminaire, ORSYS fournit aux participants un questionnaire d'évaluation du cours qui est ensuite analysé par nos équipes pédagogiques.
- Une feuille d'émargement par demi-journée de présence est fournie en fin de formation ainsi qu'une attestation de fin de formation si le stagiaire a bien assisté à la totalité de la session.

## MODALITÉS ET DÉLAI S D'ACCÈS

L'inscription doit être finalisée 24 heures avant le début de la formation.

## ACCESIBILITÉ AUX PERSONNES HANDICAPÉES

Pour toute question ou besoin relatif à l'accessibilité, vous pouvez joindre notre équipe PSH par e-mail à l'adresse psh-accueil@orsys.fr.

- Pourquoi structurer son code Puppet sous forme de modules réutilisables ?

- Ecrire un nouveau module.

*Travaux pratiques : Sélectionner et installer un module existant. Utiliser un module existant.*

*Restructurer du code Puppet sous forme de classes réutilisables. Ecriture d'un module réutilisable.*

## 5) Connecter des agents au Puppet master

- L'architecture maître-agent.

- L'authentification des clients.

- La gestion des certificats SSL.

- Planifier le lancement de l'agent.

*Travaux pratiques : Intégrer un nouveau client à une infrastructure Puppet. Lancer l'agent comme service résident. Lancer l'agent avec cron.*

## 6) Bonnes pratiques pour déployer et utiliser Puppet

- Le formatage du code.

- Utiliser la gestion de versions et l'intégrer avec Puppet.

- Séparer le code générique des paramètres spécifiques à un système.

- Gérer les données privées, comme les mots de passe et clés privées SSL.

- Superviser les rapports.

*Travaux pratiques : Utiliser Puppet-lint pour homogénéiser notre code. Extraire les paramètres spécifiques à un système vers Hiera.*

# LES DATES

---

## CLASSE À DISTANCE

2025 : 27 nov.

2026 : 30 mars, 15 juin, 29 oct.

## PARIS

2025 : 20 nov.

2026 : 30 mars, 15 juin, 29 oct.

## LYON

2025 : 27 nov.