



PowerShell - Notions avancées & DevOps

Vous utilisez régulièrement PowerShell mais vous souhaitez aller plus loin en apprenant les meilleures pratiques. Vous souhaitez découvrir les dernières fonctionnalités de Windows PowerShell qui vous permettront d'être encore plus efficace dans votre travail ? N'hésitez pas, ce cours est fait pour vous.

En plus de vous perfectionner, vous apprendrez le jargon et tous les concepts DevOps associés. Vous serez capable à la fin du cours de mettre en place des processus automatisés de livraison continue/Infrastructure-as-code, gage de qualité et de meilleure productivité.

Durée conseillée : 5 jours

Objectif du cours :

- Approfondir les concepts de PowerShell pour réaliser des scripts de qualité professionnelle,
- Apprendre les bonnes pratiques,
- Comprendre le concept et les outils DevOps sur la plateforme Microsoft,
- Découvrir les dernières fonctionnalités et orientation future de PowerShell,
- Etre autonome et savoir comment et où trouver de l'aide en cas de nécessité.

Public :

- Administrateurs systèmes
- Architectes systèmes
- Utilisateurs avancés
- Développeurs

Prérequis :

- Bonnes connaissances de PowerShell (idéalement avoir suivi le cours « Les fondamentaux »)
- Connaissances de l'administration système Windows

Occupation des journées de formation (à titre indicatif):

- Jour 1 : parties 1 à 2
- Jour 2 : parties 2 à 3
- Jour 3 : parties 4 à 5
- Jour 4 : parties 6 à 7
- Jour 5 : parties 8 à 9

Remarque : des travaux pratiques ponctuent chaque partie de cours théorique.



Itinéraire pédagogique

1. Fonctions avancées
 - a. Généralités
 - b. L'attribut [CmdletBinding]
 - c. Activer les modes verbose et debug
 - d. Accepter des données via le pipeline (par valeur et par nom de propriété)
 - e. Créer des jeux de paramètres
 - f. Ajout d'une aide intégrée
2. Objets personnalisés
 - a. Retourner des objets et non pas du texte
 - b. Création d'objets (et collections d'objets) à partir de zéro
 - c. Ajout de membres à un objet existant
 - d. Formatage d'un objet personnalisé
3. Création de modules
 - a. Pourquoi faire ?
 - b. Modules scripts, binaires et modules dynamiques
 - c. Ecriture d'un manifeste de module
 - d. Conversion d'un script en module
 - e. Publier son module dans la PowerShell Gallery
 - f. Mise en place d'une galerie privée de modules
4. DevOps ? Vous avez dit DevOps ?
 - a. Présentation du concept
 - b. Livraison continue et Infrastructure-as-Code
 - c. CloudOS et stratégie de Microsoft
 - d. Introduction au « Release Pipeline Model »
 - e. Présentation de la boîte à outils DevOps
 - f. Mise en place d'un pipeline automatisé
5. PESTER : améliorez la qualité de vos scripts
 - a. Découverte du principe du « testing »
 - b. Tests unitaires
 - c. Tests d'intégration
 - d. Mise en œuvre d'un jeu de tests automatisés
6. Gestion des configurations avec Desired State Configuration (DSC)
 - a. Plateformes supportées
 - b. Vue d'ensemble de l'architecture et terminologie
 - c. Mode Push vs mode Pull
 - d. Application d'une configuration sur un parc de machines
 - e. Présentation des ressources DSC (intégrées et additionnelles)
 - f. Ecriture d'une configuration
7. Just Enough Admin (JEA)
 - a. Présentation du module



- b. Mise en œuvre d'une politique JEA pour déléguer certaines commandes PowerShell uniquement
- c. Exécution de commandes à distance « en tant que... »
- 8. Dépôts de code (repository)
 - a. Présentation du concept
 - b. Gestion des versions
 - c. Introduction à GIT et à GitHub
 - d. Mise en place d'un dépôt local, d'un dépôt distant et synchronisation des deux
- 9. PowerShell Best Practices
 - a. Présentation du PowerShell Script Analyzer
 - b. Recueil des meilleurs pratiques