



## PowerShell - Scripting

---

### Déscription :

Plus récent que WSH, PowerShell offre des perspectives attrayantes en termes d'automatisation de tâches d'administration des systèmes Microsoft. Aussi les administrateurs rompus aux technologies Microsoft peuvent avoir besoin d'étendre leur domaine de compétences aux scripts de nouvelle génération. Les participants à cette formation sauront mettre en oeuvre et maintenir des solutions de scripts facilitant les tâches d'administration courantes à l'aide de PowerShell

### Objectifs

- Les bases du langage
- Les types et opérateurs
- Les structures de contrôle et les fonctions
- Scriptblocks et objets
- Gestion de l'affichage et navigation
- Traitement de fichiers texte et XML
- Gestion d'erreurs
- Utilisation de .NET et COM via PowerShell
- Utilisation des objets WMI
- Utilisation de Windows PowerShell pour des tâches d'administration
- Gestion de la sécurité des scripts PowerShell

### Publics

Administrateurs systèmes.

### Durée

4 jours

### Pré-requis

Connaissance d'un langage de script

### Programme de cette formation

#### Les bases du langage

- Pourquoi un nouveau Shell : le PowerShell ?
- Le lien entre PowerShell et .NET.

- 
- Installation de PowerShell.
- Exemples d'utilisations classiques.
- Les commandes et les cmdlets. Principes.
- Les Alias. Leur utilisation.
- Mécanismes d'interprétation.
- Les pipelines : comportement des flux et leur paramétrage.
- La notion de profil.
- Stratégie d'exécution.
- La documentation PowerShell et les aides

### **Les types et opérateurs**

- Le système de typage, règles de conversion.
- Les variables. Définition. Portée.
- Les types de base.
- Les expressions régulières. Qualificateur. Exemples.
- Opérateurs (arithmétique, assignation, logique, chaîne, redirection, etc.).
- Les comparateurs et les collections.

### **Les structures de contrôle et les fonctions**

- Structures de contrôle utilisant des cmdlets.
- Structures conditionnelles.
- Boucle for. Boucle while. Boucle foreach.
- Structure d'aiguillage (switch).
- Fonctions, modificateurs d'étendue.
- Phases d'exécution d'une fonction.
- Utilisation des fonctions dans un pipeline.
- Passage d'arguments à un script.

### **Scriptblocks et objets**

- Manipulation d'objets : extension d'objets, sélection, ajout de propriétés.
- Ajout de mots-clés au langage.
- Construction dynamique de code à l'exécution.

### **Gestion de l'affichage et navigation**

- Contrôle de sortie et redirection : pagination, suppression, impression.
- Navigation : gestion des lecteurs. Déplacement dans les lecteurs PowerShell.
- Accès aux données d'environnement système.

### **Traitement de fichiers texte et XML**

- Manipulation de fichiers XML de configuration.
- Traitement de fichiers structurés ou non structurés.

### **Gestion d'erreurs**

- Variables de préférence.
- Catégories d'erreur. Critiques, non critiques.

- 
- Erreurs de terminaison de scripts.
- Capture d'une erreur. Notion d'exception.
- Débogage de scripts : trace d'exécution, mode strict.
- Examen du journal d'événements Windows.

#### **Utilisation de .NET et COM via PowerShell**

- Utilisation des bibliothèques d'objets pour manipuler les composants du système.
- Accès à l'internet via PowerShell : exemples.
- PowerShell et l'interface graphique : formulaires.
- Accès aux journaux d'événements.
- Accès au réseau.

#### **Utilisation des objets WMI**

- Positionnement de WMI. Administrer Windows avec WMI.
- Listage des classes WMI.
- Notion d'adaptateur d'objet WMI.
- Utilisation des méthodes WMI.
- Utilisation des événements WMI.

#### **Utilisation de Windows PowerShell pour des tâches d'administration**

- Gestion des processus locaux et des services. Démarrage et arrêt.
- Collecte d'informations. Inventaire des installations logicielles.
- Changement d'état d'un ordinateur.
- Utilisation d'imprimantes.