



Microsoft Services Workshop*Plus*

## PowerShell pour l'administrateur IT, partie 2

*Apprenez comment utiliser Powershell pour simplifier vos tâches administratives.*

L'atelier **PowerShell pour l'administrateur IT, partie 2** est une formation Workshop*PLUS* de 3 jours qui s'appuie sur les concepts présentés dans PowerShell pour l'administrateur IT, Partie 1. Les participants à la deuxième partie du cours doivent précédemment avoir terminé la partie 1.

Ce cours porte sur les concepts PowerShell avancées telles que l'adaptation de type, en collaboration avec les bibliothèques de codes externes, la gestion des erreurs et le débogage. Les participants reçoivent les connaissances et les compétences nécessaires pour utiliser PowerShell afin de simplifier l'administration de leurs environnements.

Cet atelier se concentre sur des démonstrations et des exercices pratiques. Chaque module s'appuie sur un domaine clé PowerShell et offre de nombreuses possibilités d'apprentissage pratique. Les sujets traités comprennent:

- Un examen de PowerShell pour l'administrateur IT, Partie 1
- Scripts avec Windows PowerShell
- Avancée des scripts avec Windows PowerShell
- Gestion des erreurs et débogage
- Adaptation de type
- Utilisation des bibliothèques de code externe

Cet atelier dure «3 jours pleins. Les participants sont invités à respecter les horaires et à assister à l'intégralité de l'atelier.

## Public concerné

Pour permettre un transfert d'expertise de grande qualité durant ces 4 jours de formation, l'atelier est limité à 16 personnes maximum.

Ce cours est destiné aux administrateurs IT qui souhaitent apprendre à utiliser PowerShell pour simplifier les tâches administratives, qu'il s'agisse de travailler avec le système d'exploitation, serveur Exchange, ou l'un des produits Microsoft utilisant PowerShell. Une expérience préalable avec l'atelier PowerShell et l'achèvement de PowerShell pour l'administrateur IT, Partie 1 est fortement recommandé.

## Programme

**Module 1: Révision de PowerShell pour l'administrateur informatique, Partie 1** Ce module fournit un rappel succinct des sujets les plus importants qui ont été abordés dans l'atelier PowerShell pour l'administrateur IT, Partie 1. Notez que ce module ne contient qu'un bref aperçu de la partie 1; les sujets importants tels que l'orientation objet, les fonctions Cmdlets, structures de boucle, le pipeline, WMI et les fournisseurs de PowerShell.

**Module 2: scripts avec Windows** Ce module couvre les quatre catégories de commandes qui sont disponibles dans PowerShell. Parmi les autres sujets: L'interprétation de commande, les opérateurs PowerShell, les tableaux et les tables de hachage, Pattern Matching et expressions régulières.

**Module 3: Avancée des scripts avec Windows PowerShell** Ce module s'appuie sur le module précédent et traite des sujets de scripts supplémentaires comme Script Flow Control, PowerShell Scopes, Console Input, Scriptblocks, Dot Sourcing et Script Libraries.

**Module 4: Gestion des erreurs et débogage** Ce module couvre la gestion des erreurs et de débogage dans PowerShell. Elle couvre également les stratégies pour faire face aux erreurs de gérées et non gérées. Les différentes méthodes de débogage sont discutées dans la console et l'environnement scripting intégré, de même que les options de débogage dans les scripts.

**Module 5: Adaptation Type** Ce module explique le système d'adaptation type utilisé par PowerShell. Est également inclus dans ce module une section sur la couche PSObject. Les

participants apprendront comment étendre les classes et les instances de classes avec des membres d'objets personnalisés.

**Module 6: Utilisation des bibliothèques de code externe** Ce module comprend un large éventail des sujets qui démontrent la flexibilité disponible lorsque vous travaillez avec PowerShell. Le .Net Framework Common Language Runtime et Global Assembly Cache sont discutées ainsi que des sections sur ADO.NET, SQL Management Objects (SMO), WMI, PowerShell CCE et XML. Le module se termine par une section sur les chaînes sécurisées et la signature du script dans PowerShell.