TP2 Automatisation avec PowerShell

1/

* Version de mon PowerShell :

Une image contenant texte, capture d’écran, Police, logiciel

Description générée automatiquement

Version de la console :

Une image contenant texte, capture d’écran, Police

Description générée automatiquement

* On a environ 1629 commandes disponibles

Une image contenant texte, capture d’écran, Police

Description générée automatiquement

* On a aussi 640 Cmdlets qui sont des commandes PowerShell intégrées.

Une image contenant texte, capture d’écran, Police

Description générée automatiquement

2/ On va lister les 50 derniers évènements relatifs a des incidents de sécurité avec la commande : Get-WinEvent -LogName Security -MaxEvents 50

Une image contenant capture d’écran, texte

Description générée automatiquement

3/ On peut lister toutes les règles de notre parefeu Windows a l’aide de la commande : Get-NetFirewallRule

Une image contenant texte, capture d’écran

Description générée automatiquement

Voici une règle de mon firewall pour chrome par exemple.

4/ On va lister tous les fichiers dont la taille dépasse les 100 MB :

Une image contenant texte, capture d’écran, logiciel, Logiciel multimédia

Description générée automatiquement

On utilise la commande Get-ChildItem avec le -Path ou l’on précise le dossier de notre choix.

Avec le -Recurse on parcours tous les sous-dossiers des dossiers de notre répertoire et ensuite notre condition Where-Object pour filtrer les fichiers dont la taille est supérieure à 100MB.

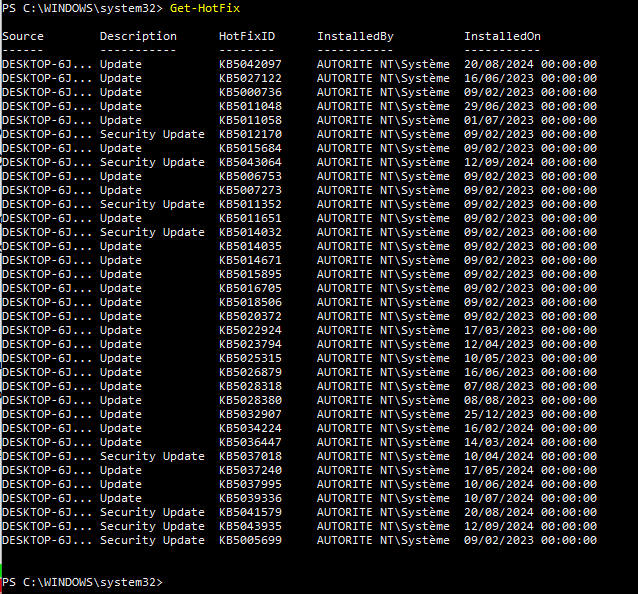
5/ Pour lister le catalogue complet des menaces que Windows Defender peut détecter on utilise la commande : Get-MpThreatCatalog

Une image contenant texte, capture d’écran

Description générée automatiquement

6/

On affiche les mises a jour installé sur notre système :



Le script qui permet d’afficher les machines qui n’ont pas fait la mise a jour n° XXX

Une image contenant texte, capture d’écran, Police

Description générée automatiquement

7/ Pour sauvegarder des informations tels que des mots de passes et login de façon sécurisé, on peut utiliser le format SecureString

Une image contenant texte, capture d’écran, Police

Description générée automatiquement

Si l’on veut afficher ces informations :

Une image contenant texte, capture d’écran, Police

Description générée automatiquement

Le mot de passe n’est pas visible

8/ Afficher les variables d’environnement a l’aide de la commande : Get-ChildItem Env :

Une image contenant texte, capture d’écran

Description générée automatiquement

9/ On va afficher es certificats sur le system puis créer un certificat auto-signé et comparer les cles publiques :

Une image contenant texte, capture d’écran

Description générée automatiquement

Les clés publiques sont différentes.

10 / On va créer un script qui récupère les statuts des services lancés sur notre machine et créer un document html avec le nom le display name et le statut du service et mettre la ligne en rouge si le statut est Stopped ou en vert si c’est Running.

Une image contenant texte, capture d’écran, Police, Parallèle

Description générée automatiquement

Script : <https://github.com/Romso94/CIL-Master1/blob/main/S%C3%A9curit%C3%A9%20Syst%C3%A8mes%20Exploitation/TP2/ServicesStatus.ps1>

11/ On va créer un script pour afficher des informations sur une machine :

* ComputerName
* Status
* OSVersion
* OSCaption
* OSArchitecture
* IPAddres
* MACAddress
* VM
* Model
* Manufacturer
* DateBuilt
* LastBootTime

Une image contenant texte, capture d’écran, logiciel

Description générée automatiquement

Script : <https://github.com/Romso94/CIL-Master1/blob/main/S%C3%A9curit%C3%A9%20Syst%C3%A8mes%20Exploitation/TP2/report.ps1>