Powershell sur Windows 10 : qu’est-ce, comment l’ouvrir et quelques commandes

Powershell apparu sur Windows 7 et très mis en avant sur Windows 10.
Dans cet article, nous vous expliquerons ce qu’est Powershell et comment l’ouvrir sur Windows 10.
Enfin, quelques commandes basiques seront données pour avoir un aperçu dans Windows 10.

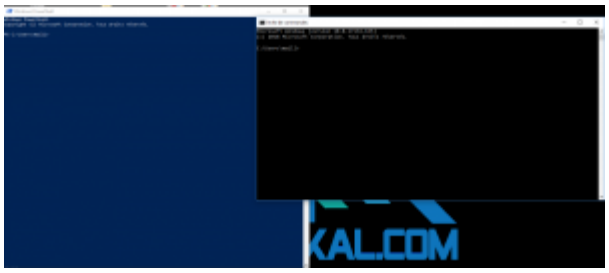
Powershell sur Windows 10 : qu’est-ce, comment l’ouvrir et quelques commandes.

Qu’est-ce que Powershell sur Windows 10 ?

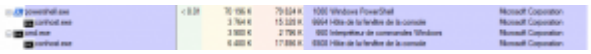
Powershell est donc le digne successeur de l’invite de commandes de Windows très vieillissante.
Ce dernier est disponible sur Windows 7, Windows 8.1 et Windows 10.

Le mot shell signifie interface logiciel, il s'agit donc d'une couche logiciel entre l'utilisateur et Windows qui est beaucoup plus poussé que l'invite de commandes.
Notamment Powershell permet l'exécution de script très poussé orienté objet, donc rien à voir avec cmd.exe.
Certains fonctions reprennent aussi les Unix shell comme la possibilité de passer le résultat d'une commande à l'autre avec le pipe.
Le but de Microsoft est de faire un langage de script aussi développé que ceux sous Unix afin de concurrencer ce dernier dans le monde des serveurs.

D'un point de vue présentation, il n'y a cependant guerre de différences entres les deux.
Ci-dessous une capture d'écran de Windows 10 avec à gauche la fenêtre Powershell et à droite l'invite de commandes.



De même au niveau des processus, seul le nom change, les deux interfaces s'appuie sur le processus conhost.exe.


A ce jour sur Windows 10 1803, Powershell n'est pas disponible depuis les options de récupération de Windows où l'invite de commandes est encore présente.

Comment ouvrir Powershell sur

Windows 10

Méthode 1 : clic droit menu démarrer

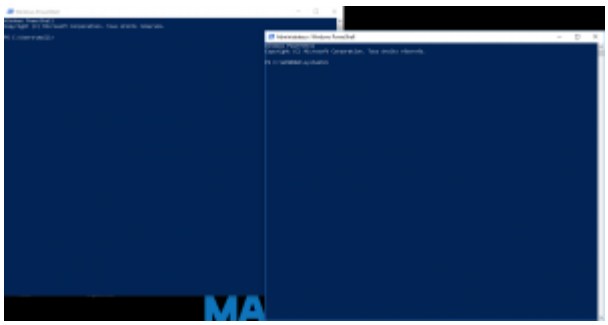
Il existe de multiples façon de lancer Powershell sur Windows 10.
La première méthode consiste à faire un clic droit sur le menu Démarrer de Windows 10, vous trouverez alors les options :

- *Windows PowerShell* qui lance l'application avec l'utilisateur courant mais sans les droits administrateur.
- *Windows Powershell (admin)* même chose mais l'application est exécutée avec un jeton administrateur.

Ce fonctionnement est dicté par le contrôle des comptes utilisateur (UAC) qui est identique pour toutes les applications de Windows.
Plus d'informations sur notre article complet : Le contrôle des comptes utilisateurs (UAC) de Windows

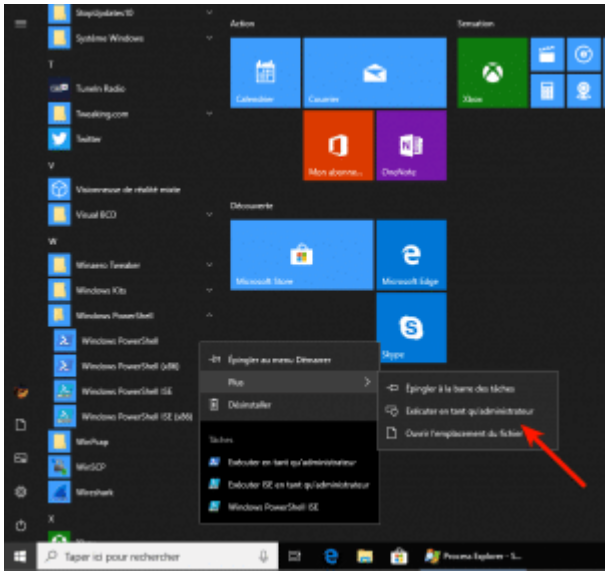
On retrouve alors le même principe que pour l'invite de commandes avec à gauche, Powershell lancé sans le jeton

administrateur et à droite avec.
De ce fait, le chemin donné n'est pas le même selon le cas.
Sans les droits administrateur, on obtient le chemin du profil utilisateur alors qu'avec les droits administrateur, le chemin est C:\Windows\system32



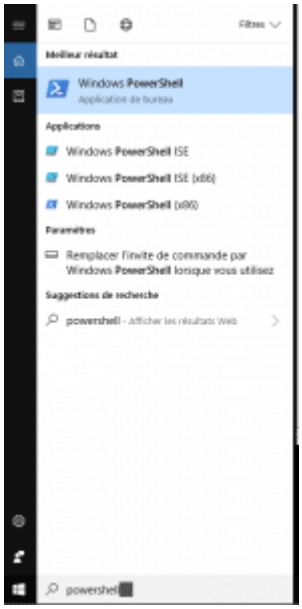
Méthode 2 : menu Démarrer de Windows 10

Comme toute application installée ou présente en natif sur Windows, celle-ci est listé dans le menu Démarrer de Windows 10.
Dans le cas de Powershell, il faut se rendre dans le menu Windows Powershell.
A partir de là, toutes les déclinaisons sont proposées avec Powershell en 32-bits ou 64-bits et administrateur ou non administrateur.



Méthode 3 : Par Cortana

Enfin la dernière méthode consiste à utiliser Cortana et faire une recherche sur le mot PowerShell.
Cela permet d'obtenir là aussi toutes les déclinaisons des raccourcis de lancement de PowerShell.



Quelques commandes Powershell

Dans Powershell, on ne parle pas de commandes mais de **cmdlets.** oici la description de Microsoft : « *Une cmdlet (prononcez « command-let ») est une commande à fonction unique qui manipule des objets dans Windows PowerShell. Vous pouvez reconnaitre les cmdlets par leur format de nom – un verbe et un nom séparés par un tiret (-), comme Get-Help, Get-Process, et Start-Service*. »

Pour débuter en programmation Powershell, vous pouvez suivre ces deux liens :

- PowerShell Documentation
- A Task-Based Guide to Windows PowerShell Cmdlets
- Une page pour débuter en Powershell en français

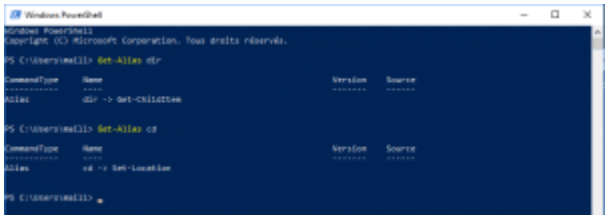
Voici quelques commandes Powershell qui permettent de

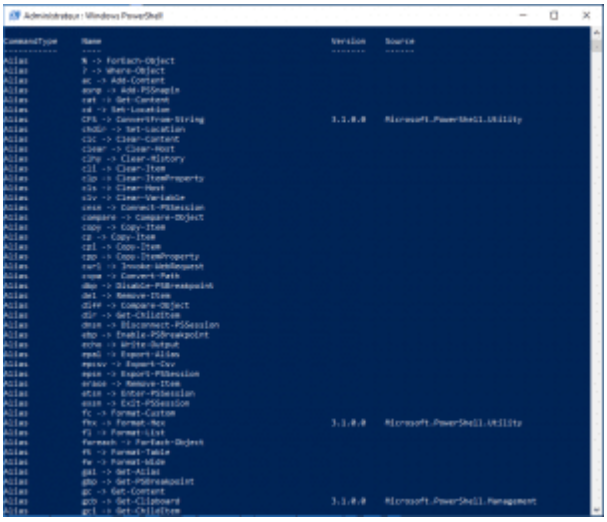
manipuler les fichiers ou dossiers.
Ce sont ici que des exemples puisque Powershell ne reste pas qu'à ces aspects de fichiers ou dossiers et permet bien d'autres choses.
Par exemple, Windows Defender a ses propres cmdlets pour configurer ce dernier, vous trouverez des exemples sur la page : Les réglages avancés de Windows Defender : Tutoriel

PowerShell (Cmdlet)	PowerShell (Alias)	CMD.EXE / COMMAND.COM	Shell Unix	Description
Get-ChildItem	gci, dir, ls	dir	ls	Liste les fichiers / répertoires du répertoire (courant)
Get-Content	gc, type, cat	type	cat	Obtenir le contenu d'un fichier
Get-Command	gcm	help	help, which	Liste des commandes
Get-Help	help, man	help	man	Aide
Clear-Host	cls, clear	cls	clear	Efface l'écran
Copy-Item	cpi, copy, cp	copy	cp	Copier un ou plusieurs fichiers / l'arborescence complète
Move-Item	mi, move, mv	move	mv	Déplacer un fichier / répertoire
Remove-Item	ri, del, erase, rmdir, rd, rm	del, deltree, erase, rmdir, rd	rm, rmdir	Supprimer un fichier / répertoire
Rename-Item	rni, ren, mv	ren, rename	mv	Renommer un fichier / répertoire
Get-Location	gl, pwd	cd	pwd	Afficher le répertoire de travail courant
Pop-Location	popd	popd	popd	Changer le répertoire courant vers le répertoire le plus récemment poussé sur la pile
Push-Location	pushd	pushd	pushd	Pousser le répertoire courant sur la pile
Set-Location	sl, cd, chdir	cd, chdir	cd	Changer le répertoire courant
Tee-Object	tee	NC	tee	Diriger l'entrée vers un fichier ou une variable, puis la passer dans un pipeline
Write-Output	echo, write	echo	echo	Afficher des chaînes, variables etc sur la sortie standard
Get-Process	gps, ps	tlist	ps	Liste de tous les processus en cours d'exécution
Stop-Process	spps, kill	kill	kill	Arrêter un processus en cours d'exécution
Select-String	sls, findstr	find, findstr	grep	Recherche d'une chaine de caractère
Set-Variable	sv, set	set	env, export, set, setenv	Définir la valeur d'une variable / créer une variable
Invoke-WebRequest	iwr, wget, curl	NC	wget, cURL	Obtient le contenu d'une page web

source https://fr.wikipedia.org/wiki/Windows_PowerShell

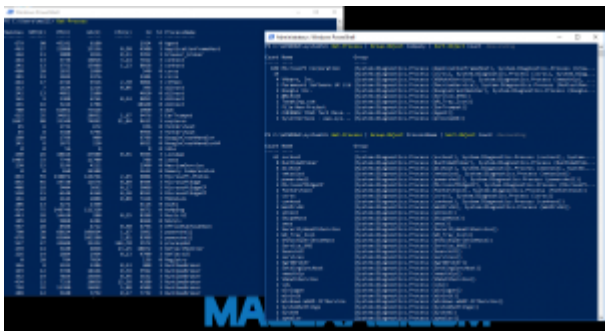
Le cmdlets Get-Alias permet d'obtenir l'équivalent d'une commande de l'invite de commandes en cmdlets.





Enfin il est possible de passer le résultat d'un cmdlets à un autre avec le pipe comme c'est le cas sur Unix.
A gauche, on utilise le cmdlets Get-Process pour obtenir la liste des processus en cours de fonctionnement.

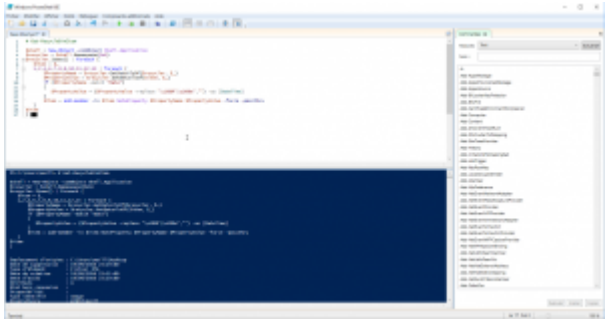
A droite même chose mais on groupe par le nom de compagny et on compte le nombre de processus sur ce groupe ou encore simplement le nombre de processus.
On constate qu'il y a 62 processus svchost.exe et 8 processus RuntimeBroker.



Il est aussi possible de gérer ses partitions de disques avec PowerShell : Créer, supprimer, formater des partitions de disque en Powershell

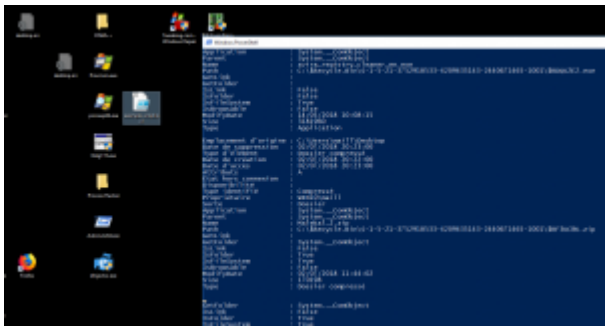
Script Powershell

Windows propose un éditeur qui permet d'écrire des scripts Powershell : **Windows PowerShell ISE**
Il s'agit d'un éditeur classique avec la complénation des commandes disponibles, correction de syntaxte etc.
Et bien sûr vous pouvez enregistrer votre script, l'exécuter et le déboguer.



Le script enregistré aura alors l'extension .ps1 et un clic droit puis exécuter permet de lancer ce dernier.
La fenêtre Powershell s'ouvre et exécuter ce dernier et se ferme exactement comme c'est le cas d'une fenêtre de script de

l’invite de commandes.



Powershell est les malwares

Les scripts batchs utilisant l’invite de commandes n’étaient pas très utilisés par les malwares car assez limités.
Toutefois des commandes cmd.exe pouvait être lancé souvent pour installer l’infection dans le système « drop ».

Powershell étant beaucoup plus sophistiqué et pouvant faire beaucoup plus de choses, des scripts Powershell ont été utilisés par des malwares.
Un article complet existe sur le sites sur les logiciels malveillants en PowerShell : Les virus ou trojan Powershell