

Windows Server

Gestion des Commandes CMD & PS

PARTIE 1 : COMMANDES « CMD »

A. GESTION DU SYSTEME & UTILISATEURS

Ouvrir une invite de commandes (cmd.exe). Pour obtenir la liste des commandes supportées par votre système, saisir : « help ». Pour obtenir de l'aide sur une commande saisir : « nom_cde /? »

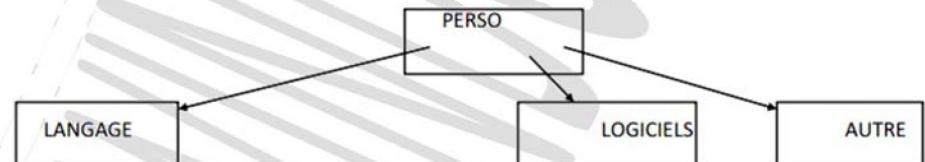
1. Utiliser la commande « SHUTDOWN » avec les options appropriées.
Arrêt du serveur :
Reboot du serveur :
Mise en veille prolongée du serveur :
Arrêt du serveur au bout d'une heure :
Annulation de la commande d'arrêt du serveur :
2. Trouver la commande et ses options appropriées pour changer les couleurs de l'invite de commande avec le fond en bleu-gris et le texte en jaune :
3. Trouver les commandes permettant de changer la date et l'heure du système :
4. Trouver la commande pour afficher les processus en cours :
Donner le n° d'identifiant correspondant au processus cmd.exe :
5. Utiliser la commande « START » pour démarrer le bloc note la fenêtre agrandie (la fermer) :
Puis pour démarrer le bloc note la fenêtre minimisée dans la barre des tâches (et la laisser ouverte) :
6. Repérer le PID du bloc note et trouver la commande pour terminer le processus à partir de ce PID :
7. Trouver la commande pour afficher le numéro de série du volume :
8. Trouver la commande pour afficher le numéro de version de l'OS :
9. Trouver la commande pour afficher toutes les informations du système :
Donner le nombre de type d'information existant à propos de la mémoire :
10. Utiliser la commande « DOSKEY » pour afficher l'historique des commandes :
Pour définir le nombre de lignes de l'historique des commandes :
A remarquer qu'il est possible d'utiliser la touche F7 pour afficher une boîte de dialogue de l'historique, puis la touche F9 pour saisir le numéro correspondant à la commande sélectionnée.
11. Changer le prompt de manière à ce qu'il affiche :
`le_n°_version_OS le_lecteur_et_chemin_courant =>`
12. Changer le prompt de manière à ce qu'il affiche :
`la_date>le_lecteur_et_chemin_courant $`
13. Que fait la commande « whoami » ?
14. Que fait la commande « reg query » ? Donner un exemple d'utilisation.
15. Que fait la commande « schtasks » ? Donner un exemple d'utilisation.

B. GESTION DES FICHIERS

1. Utiliser la commande « ASSOC » pour trouver avec quelle application est utilisée chacune des extensions suivante : .exe .ost .rtf .scp .vbs .vsd
2. Utiliser la commande « MKDIR » pour créer un répertoire de travail (dans lequel vous effectuerez le TP) à la racine du système tel que : « c:\TP_NOM\TP_CMD » :
3. Saisir un texte quelconque (en utilisant la commande « ECHO » et le signe >) afin d'y créer un fichier sous le nom « test.txt » puis utiliser la commande « TYPE » pour afficher le contenu de votre fichier :
4. Donner le nom de la commande pour vérifier que ce fichier est bien dans votre répertoire
5. Trouver la commande pour renseigner l'attribut donné par le système à ce fichier :
6. Ajouter l'attribut « lecture seule ». Essayer de modifier le contenu du fichier. Est-ce possible ?
Essayer de l'enregistrer sous le même nom. Est-ce possible ?
7. Décrire ce que fait la commande « type nul>nomfichier »
8. Saisir un texte quelconque afin d'y créer un fichier sous le nom « essai.dat » puis utiliser la commande « MORE » pour visualiser son contenu :
9. Trouver la commande pour changer le nom d'essai.dat en essai1.dat :
10. Trouver la commande pour copier essai1.dat vers essai2.dat :
11. Visualiser tous les fichiers « .dat » de votre répertoire et uniquement ceux-ci :
12. Copier les fichiers essai1.dat et essai2.dat dans un fichier global texte.dat :
13. Copier le fichier texte.dat vers texte1.dat :
14. Ajouter un texte à la suite du texte contenu dans le fichier texte1.dat :
15. Détruire (del) en une seule fois tous les fichiers dont le nom commence par « texte » :
16. Détruire tous les fichiers « essai?.dat » :

C. GESTION DES RÉPERTOIRES

1. Trouver la commande pour créer l'arborescence suivante dans votre répertoire :



Noter la(es) commande(s) passée(s) :

2. Décrire ce que fait la commande « tree »
3. Créer (echo>) un fichier essai.dat dans le répertoire AUTRE tout en restant dans le répertoire PERSO :
4. Trouver la commande pour se déplacer dans le répertoire AUTRE :
5. Le répertoire de travail étant AUTRE, copier ce fichier sous le répertoire LANGAGE :
6. Trouver la commande pour supprimer tous les répertoires créés dans le répertoire PERSO et le répertoire PERSO lui-même en une seule commande :

PARTIE 2 : COMMANDLETS POWERSHELL

A. OBTENIR DE L'AIDE, INFORMATIONS D'ALIAS & D'ATTRIBUTS

Afficher de l'aide sur une commande : `Get-Help Commande` (ex : `Get-Help Get-ChildItem`)

Afficher les exemples : `Get-Help Commande -Examples`

Afficher les alias : `Get-Alias`

Afficher la liste des méthodes et des propriétés des objets : `Commande | Get-member`

Réaliser les actions suivantes :

- 1) Afficher l'aide sur la commande `Get-Alias`
- 2) Afficher l'aide avec les exemples sur la commande `Get-Alias`
- 3) En appuyant sur cette aide : Afficher tous les alias dont le nom commence par la lettre g
- 4) Afficher la commande qui correspond à l'alias dont le nom est `sl` (SL)
- 5) Afficher tous les alias dont la définition est `Get-ChildItem` (Retrouver les alias des commandes DOS et Linux).
- 6) A partir de l'ex. 2 de l'aide de la commande `Get-PSDrive`, afficher les informations du volume C.
- 7) Afficher les méthodes et les propriétés des objets retournés par la commande `Get-Location`
- 8) Afficher les méthodes et les propriétés des objets retournés par la commande `Get-PSDrive`

B. GESTION DES FICHIERS ET DES DOSSIERS

Se déplacer dans les dossiers : `Set-Location chemin` (ex chemin : `c:\temp`)

Afficher le chemin du dossier courant : `Get-Location`

Afficher le contenu d'un dossier : `Get-ChildItem`

Créer un dossier : `New-Item nomDossier -ItemType directory`

Créer un fichier avec du texte :

`New-Item nomFichier.txt -ItemType file -Value "texte"`

Supprimer un fichier ou un dossier : `Remove-Item nomFichier.txt`

Déplacer un fichier :

`Move-Item nomFichier.txt -Destination chemin\nomFichier.txt`

Déplacer un dossier : `Move-Item nomDossier -Destination chemin\nomDossier`

Renommer un fichier ou dossier :

`Rename-Item nomFichier.txt -NewName nomFichier2.txt`

Copier un fichier : `Copy-Item nomFichier.txt -Destination nomFichier2.txt`

Copier un dossier avec ses fichiers :

`Copy-Item nomDossier -Destination nomDossier1 -Recurse`

Tester l'existence d'un fichier ou dossier : `Test-Path chemin/nomFichier.txt`

Réaliser les actions suivantes :

- 1) Afficher le chemin du dossier courant
- 2) Se déplacer à la racine de la partition C: (chemin `c:\`)

3) Afficher la liste des dossiers et fichiers.

4) A cet emplacement, créer un dossier nommé `REP-VotreNOM\testPowerShell`

5) Se déplacer dans le dossier `c:\REP-VotreNOM\testPowerShell`

6) Créer un dossier nommé `testdossier`

7) Créer un fichier nommé `test1.txt`, contenant la phrase "Tp PowerShell 1"

8) Afficher la liste des dossiers et fichiers

9) Copier le fichier `test1.txt` sous le nom `test2.txt`

10) Renommer le fichier `test1.txt` avec le nom `essai1.txt`

11) Copier le fichier `essai1.txt` dans le dossier `testdossier\essai1.txt`

12) Afficher la liste des fichiers du dossier et des sous-dossiers de `testPowerShell`

13) Copier le dossier `testdossier` (avec ses fichiers) dans un nouveau dossier `test2dossier`

14) Déplacer le fichier `test2.txt` dans le dossier `testdossier`

15) Supprimer le dossier `test2dossier` (avec ses fichiers)

16) Tester l'existence du dossier `c:\windows`

17) Afficher le contenu du dossier `c:\windows`

18) Afficher la liste des fichiers .exe du dossier `c:\windows`

C. INITIATION AUX VARIABLES, PROPRIÉTÉS & MÉTHODES DES OBJETS

Le nom d'une variable commence toujours par \$, il peut inclure tout caractère alphanumérique ou le trait de soulignement. Windows PowerShell permet de créer des variables qui sont pour l'essentiel des objets nommés. La sortie de toute commande Windows PowerShell valide peut être stockée dans une variable.

Exemple : `$loc = Get-Location`

Il est possible d'utiliser `Get-Member` pour afficher des informations sur le contenu de variables.

Exemple : `$loc | Get-Member` (idem `Get-Location | Get-Member`)

Le nom de la variable suivi du point permet d'accéder aux propriétés de l'objet référencé par la variable, exemple pour la propriété `Path` de la variable `$loc`. Exemple : `$loc.Path`

L'usage de la touche tabulation [tab] permet de compléter le nom de la propriété. De même, l'exécution d'une méthode (action) d'un objet : Exemple : `$fichier.Delete()`
Pour les méthodes, ne pas oublier les parenthèses avec ou sans paramètre.

Réaliser les actions suivantes :

1) Affecter à la variable `$loc`, le résultat de la commande `Get-Location`.

2) Afficher les propriétés et les méthodes de la variable `$loc`

3) Afficher le chemin du dossier courant contenu dans cette variable.

4) Afficher les informations sur le disque contenu par cette variable.

5) Afficher les informations sur le 'Provider' contenu par cette variable.

6) Affecter à la variable `$lect`, le résultat de la commande `Get-PSDrive -Name C`

7) Afficher les propriétés et les méthodes de la variable `$lect`

- 8) A partir de la variable \$lect, afficher la description du lecteur C, afficher la taille en octet du volume utilisé, afficher la taille en octet du volume libre (pour avoir ces tailles en Go, diviser par 1GB en utilisant l'opérateur /)
- 9) Affecter à la variable \$fichier, le résultat de la commande

```
Get-ChildItem c:\REP-VotreNOM\testPowerShell\essai1.txt
```
- 10) Afficher les propriétés et les méthodes de la variable \$fichier
- 11) A partir de la variable \$fichier, afficher le nom du fichier, afficher la taille en octet du fichier, afficher le nom complet du fichier (avec le chemin), afficher l'extension seule du fichier, afficher la date du dernier accès.
- 12) A l'aide d'une méthode de la variable \$fichier, copier ce fichier dans un nouveau fichier nommé :

```
C:\REP-VotreNOM\TestPowerShell\essai2.txt
```
- 13) A partir de la variable \$fichier, supprimer le fichier essai1.txt
- 14) Vérifier avec la commande Get-ChildItem
- 15) Lancer notepad.exe et réduire la fenêtre du Bloc-notes
- 16) Lancer la commande Get-Process et vérifier que le Bloc-notes soit bien dans les processus actifs
- 17) Affecter à la variable \$proc, le résultat de la commande Get-Process notepad
- 18) Afficher les propriétés et les méthodes de la variable \$proc
- 19) A partir de la variable \$proc, afficher la description du processus, afficher le chemin d'accès de l'exécutable.
- 20) A partir de la variable \$proc, supprimer (tuer) le processus du Bloc-notes

D. ACCES AUX INFORMATIONS DU SYSTEME

Les classes WMI (Windows Management Instrumentation) décrivent les ressources qui peuvent être gérées. Cette couche permet de récupérer beaucoup d'informations sur les machines exécutant Windows (version de l'OS, taille de la RAM, processus en cours, version du BIOS, cadence actuelle du processeur...) mais également d'exécuter des tâches d'administration (mise à jour de paramètres, reboot, démarrer/arrêter un processus ...). Il existe des centaines de classes WMI, dont certaines contiennent des dizaines de propriétés.

La commande principale est Get-WmiObject, elle permet de lire ces ressources.

Exemple pour consulter les informations suivantes :

Graphiques : Get-WmiObject win32_videocontroller

Système : Get-WmiObject win32_operatingsystem

Disques : Get-WmiObject win32_logicaldisk

Il est toujours possible d'affecter le résultat de la commande Get-WmiObject à une variable, et de consulter les propriétés et les méthodes de l'objet à l'aide de la commande Get-Member.

Si le résultat de la commande est un ensemble d'objets, la variable affectée est un tableau d'objet, l'accès au premier élément se fait alors de la manière suivante \$var[0], au second élément : \$var[1], etc.

Réaliser les actions suivantes :

- 1) Afficher toutes les informations concernant le contrôleur vidéo de votre système
- 2) Affecter à la variable \$video, le résultat de la commande précédente
- 3) Afficher les propriétés et les méthodes de la variable \$video

- 4) A partir de la variable \$video, afficher le nom du contrôleur, la version du driver, le mode vidéo (résolution) et le nom du processeur vidéo
- 5) Afficher les informations concernant le système d'exploitation
- 6) Affecter à la variable \$os, le résultat de la commande précédente
- 7) A partir de la variable \$os, afficher le nom du système, le type d'architecture (32-64 bits), la date d'installation.
- 8) Afficher les informations concernant les disques logiques de votre système
- 9) Affecter à la variable \$vol, le résultat de la commande précédente
 Si votre système comporte plusieurs disques logiques, la variable \$vol est un tableau d'objets.
- 10) A partir de la variable \$vol, et pour le premier disque logique seulement, afficher le nom du volume, afficher la taille, afficher l'espace libre, afficher le système de fichiers (pour avoir ces tailles en Go, diviser par 1GB en utilisant l'opérateur /).

PARTIE 3 : COMMANDES « NET »

Expérimenter les quelques commandes suivantes :

1. Vérifier l'état du compte administrateur : net user administrateur
 Activer le compte administrateur : net user administrateur /active:yes
 Définir un mot de passe : net user administrateur *Désactiver le compte administrateur : net user administrateur /active:no
2. Obtenir le nom du serveur : hostname
 Renommer le serveur (ex. : SRV03) : netdom renamecomputer OldName /Newname:SRV03
 Reboot immédiat du serveur : shutdown /r /t 0
3. Obtenir la liste des cartes réseaux installées et leurs paramètres :
 netsh interface ipv4 show interfaces
*Une métrique est assignée à chaque interface, ainsi qu'un Idx...
 Plus la valeur métrique est petite, plus l'utilisation de cette interface est prioritaire.*
4. Obtenir la liste des configurations d'adresse IP :
 netsh interface ipv4 show addresses
5. Visualiser les paramètres de tous les profils du pare-feu :
 netsh advfirewall show allprofiles
6. Désactiver tous les profils du pare-feu :
 netsh advfirewall set allprofiles state off
7. Réactiver tous les profils du pare-feu :
 netsh advfirewall set allprofiles state on
8. Activer Windows sur le serveur : slmgr.vbs -ato
 (aucun message généré si l'activation réussie)