**METGY LEO SIO1**

**TP - Maîtrise du CMD Windows : Administration système et réseau**

**Objectifs pédagogiques**

- Maîtriser les commandes essentielles de l'invite de commandes Windows
- Comprendre la gestion des fichiers, processus et réseaux
- Développer des compétences en troubleshooting système

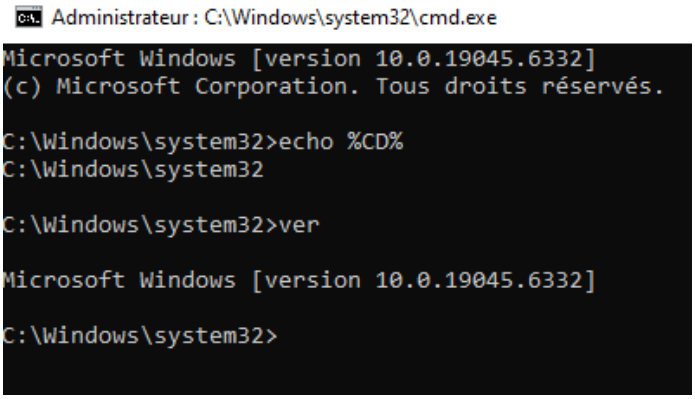
## Exercice 1 : Prise en main de l'environnement CMD

### 1.1 Lancement et configuration

- Ouvrez CMD en tant qu'administrateur :
  - Win + X → Invite de commandes (admin) **OU**
  - Win + R → cmd → Ctrl + Shift + Entrée

**Tâches :**

1. Vérifiez votre dossier courant avec ***echo %CD%***
2. Affichez la version de Windows avec ver
3. Capture d'écran montrant ces informations




### 1.2 Personnalisation de l'environnement

- Tapez ***color 0A***
- Tapez ***prompt $T [$P]$G***
- Tapez ***title Session_Admin_CMD***

**Questions :**

- Que fait la commande color ?

  Elle change la couleur du texte dans le CMD.
- Comment le prompt a-t-il changé ?

  $T affiche l'heure, [$P] affiche le chemin d'accès complet du répertoire courant entre crochets et $G affiche le caractère >.
- À quoi sert la commande title ?

  La commande title sert à donner un titre à sa fenêtre d'invite de commande.

## Exercice 2 : Exploration avancée du système de fichiers

### 2.1 Navigation et arborescence

**Tâches :**

1. Créez la structure suivante :

*TP_CMD*

*Documents/*

*Textes/*

*Images/*

*Archives/*

*Temp/*

### 2.2 Attributs de fichiers avancés

*echo CONTENU_SECRET > Documents\fichier_cache.txt*

*dir*

*attrib +H Documents\fichier_cache.txt*

*dir Documents*

*dir Documents /A*

**Questions :**

- Que signifient les attributs +H ?
  Les attributs +H servent à cacher les fichiers ou dossiers (H pour hidden en anglais).
- Comment afficher les fichiers cachés avec dir ?
  En utilisant la commande dir … /a.

## Exercice 3 : Gestion des processus et services

### 3.1 Analyse des processus système

*tasklist /FO TABLE /V*

**Tâches :**

1. Lancez le Bloc-notes et Paint
2. Identifiez leurs PID avec :

*tasklist | findstr "notepad mspaint"*

```
15:16:13,69[E:\BLOC 1 M.SGHAYRON\TP_CMD]>tasklist | findstr "notepad mspaint"
notepad.exe                  14040 Console                    2     14 956 Ko
mspaint.exe                  13020 Console                    2     32 000 Ko

15:17:05,32[E:\BLOC 1 M.SGHAYRON\TP_CMD]>
```

### 3.2 Gestion fine des processus

*:: Tuer un processus par PID*

*taskkill /PID [PID_NUMBER] /F*

```
15:18:32,64[E:\BLOC 1 M.SGHAYRON\TP_CMD]>taskkill /PID 13020 /F
Opération réussie : le processus avec PID 13020 a été terminé.

15:19:48,57[E:\BLOC 1 M.SGHAYRON\TP_CMD]>taskkill /PID 14040 /F
Opération réussie : le processus avec PID 14040 a été terminé.
```

*:: Tuer par nom d'image*

***taskkill /IM notepad.exe /T***

```
15:19:58,36[E:\BLOC 1 M.SGHAYRON\TP_CMD]>
15:19:58,47[E:\BLOC 1 M.SGHAYRON\TP_CMD]>taskkill /IM notepad.exe /T
Opération réussie : un signal de fin a été envoyé au processus de PID 10864,
processus enfant de PID 11820.
```

**Questions :**

- Quelle est la différence entre /F et /T ?

  Le /F force la fermeture d'un processus ciblé même s'il ne répond pas alors que le /T permet non seulement de tuer le processus principal mais aussi tous ses processus enfants qu'il a lancé.

- Pourquoi faut-il être prudent avec taskkill ?

  Car on peut forcer la fermeture d'un ou des processus essentiels au bon fonctionnement de Windows ou autre OS.

## Exercice 4 : Diagnostic réseau avancé

### 4.1 Analyse complète de la configuration

***ipconfig /all***

***ou***

***netsh interface ip show config***

```
Carte Ethernet Eth 1 - Realtek :

   Suffixe DNS propre à la connexion. . . : sio.edu
   Description. . . . . . . . . . . . . . : Realtek PCIe GbE Family Controller
   Adresse physique . . . . . . . . . . . : 18-60-24-F4-CF-5C
   DHCP activé. . . . . . . . . . . . . . : Oui
   Configuration automatique activée. . . : Oui
   Adresse IPv6 de liaison locale. . . . .: fe80::dc7:4b5e:77de:d412%4(préféré)
   Adresse IPv4. . . . . . . . . . . . . .: 192.168.28.52(préféré)
   Masque de sous-réseau. . . . . . . . . : 255.255.255.0
   Bail obtenu. . . . . . . . . . . . . . : lundi 6 octobre 2025 12:09:44
   Bail expirant. . . . . . . . . . . . . : lundi 6 octobre 2025 17:08:22
   Passerelle par défaut. . . . . . . . . : 192.168.28.253
   Serveur DHCP . . . . . . . . . . . . . : 192.168.28.253
   IAID DHCPv6 . . . . . . . . . . . : 102260772
   DUID de client DHCPv6. . . . . . . . : 00-01-00-01-2E-63-32-A0-18-60-24-F4-CF-5C
   Serveurs DNS. . .  . . . . . . . . . . : 193.49.251.6
                                       54.38.53.123
                                       8.8.8.8
   NetBIOS sur Tcpip. . . . . . . . . . . : Activé
```

**Questions :**

- Quelle est votre adresse MAC ?

  Mon adresse MAC est : 18-60-24-F4-CF-5C

- Le DHCP est-il activé ?

  Oui le DHCP est activé.

- C'est quoi un DHCP ?

  C'est un protocole réseau qui permet à un appareil connecté de recevoir automatique son adresse IP.

- Quel serveur DNS utilisez-vous ?

  J'utilise ce serveur DNS : 193.49.251.6

### 4.2 Tests de connectivité

***ping 8.8.8.8 -n 4***

***ping google.com***

```
15:24:48,94[E:\BLOC 1 M.SGHAYRON\TP_CMD]>ping google.com

Envoi d'une requête 'ping' sur google.com [142.250.179.110] avec 32 octets de données :
Réponse de 142.250.179.110 : octets=32 temps=15 ms TTL=114
Réponse de 142.250.179.110 : octets=32 temps=14 ms TTL=114
Réponse de 142.250.179.110 : octets=32 temps=14 ms TTL=114
Réponse de 142.250.179.110 : octets=32 temps=15 ms TTL=114

Statistiques Ping pour 142.250.179.110:
    Paquets : envoyés = 4, reçus = 4, perdus = 0 (perte 0%),
Durée approximative des boucles en millisecondes :
    Minimum = 14ms, Maximum = 15ms, Moyenne = 14ms
```

***tracert google.com***

***pathping google.com***

```
15:28:40,11[E:\BLOC 1 M.SGHAYRON\TP_CMD]>tracert google.com

Détermination de l'itinéraire vers google.com [142.250.179.110]
avec un maximum de 30 sauts :

  1    <1 ms    <1 ms    <1 ms  192.168.28.253
  2     1 ms     1 ms     1 ms  10.96.9.245
  3    10 ms    10 ms    10 ms  172.18.128.93
  4    11 ms    10 ms    11 ms  10.251.0.2
15:33:54,66[E:\BLOC 1 M.SGHAYRON\TP_CMD]>pathping google.com

Détermination de l'itinéraire vers google.com [142.250.179.110]
avec un maximum de 30 sauts :
  0  host.docker.internal [192.168.28.52]
  1  192.168.28.253
  2  10.96.9.245
  3  172.18.128.93
  4  10.251.0.2
  5  routeur-upjv.ac-amiens.fr [195.221.156.15]
  6  194.167.1.141
  7  vl221-be2-ren-nr-compiegne-rtr-091.noc.renater.fr [193.51.181.190]
  8  xe-0-0-8-ren-nr-paris2-rtr-131.noc.renater.fr [193.51.177.206]
  9  192.178.70.144
 10  72.14.233.195
 11  142.251.49.135
 12  par21s20-in-f14.1e100.net [142.250.179.110]

Traitement des statistiques pendant 300 secondes...
```

**Tâches :**

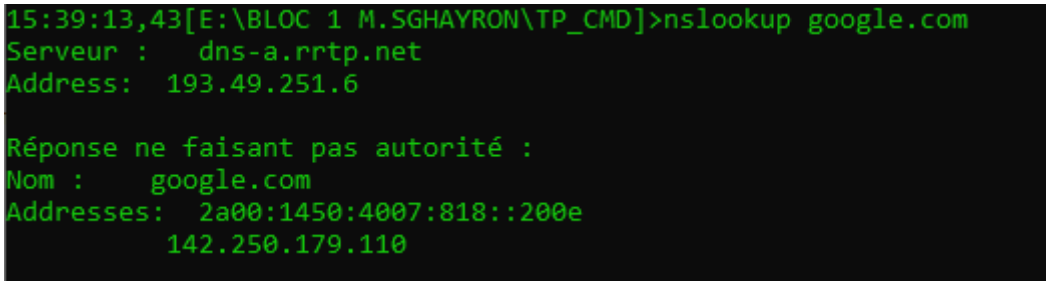
1. Explique le résultat du premier ping
    - Réponse de 8.8.8.8 : octets=32 temps=12 ms TTL=115

8.8.8.8 est le serveur DNS public de Google, octect=32 est la taille du paquet envoyé et reçu, temps=12ms c'est le temps aller-retour entre mon ordinateur et le serveur donc 12 millisecondes et le TTL = 115 c'est une valeur qui indique combien de « sauts » (routeurs) le paquet peut encore faire avant d'être détruit.

2. Comparez les résultats de tracert et pathping

   Les résultats de tracert et pathping sont identiques.

3. Testez la résolution DNS avec nslookup google.com




### 4.3 Statistiques réseau

***netstat -ano***

***netstat -an | find ":80"***

***netstat -e 5***

**Questions :**

- Que montre netstat -ano ?

  Cela montre les connexions réseau actives sur mon ordinateur avec trois options :

  -a qui permet d'afficher toutes les connexions et ports d'écoute (TCP et UDP).

  -n qui permet d'afficher les adresses et numéros de ports sous forme numérique (pas de résolution en noms).

  -o qui permet d'afficher le PID (identifiant du processus) de chaque connexion.

- Comment identifier les connexions établies sur le port 80 ?

  On peut identifier les connexions établies sur le port 80 grâce à cette commande : netstat -ano | findstr ":80"

## Exercice 5 : Scripting basique et automatisation

### 5.1 Création d'un script de sauvegarde

Créez un fichier sauvegarde.bat :

```
@echo off
echo === SAUVEGARDE DOSSIERS IMPORTANTS ===
set BACKUP_DIR=%USERPROFILE%\Backup_%DATE%
mkdir "%BACKUP_DIR%"

xcopy "%USERPROFILE%\Documents\*.txt" "%BACKUP_DIR%\Textes\" /S /I /Y
xcopy "%USERPROFILE%\Desktop\*.pdf" "%BACKUP_DIR%\PDF\" /S /I /Y

echo Sauvegarde terminee: %BACKUP_DIR%
dir "%BACKUP_DIR%" /S
pause
```

1. Que fait la commande @echo off et pourquoi est-elle utile ici ?

   La commande @echo off sert à masquer les commandes exécutées lorsqu'on lance le script.

   Elle est utile car dans notre cas il y a beaucoup de commandes au sein du script.

2. Que fait la commande set BACKUP_DIR=%USERPROFILE%\Backup_%DATE% ?

   Cette commande créer une variable qui contient le chemin d'accès de l'utilisateur suivi d'un sous dossir nommé Backup_ ainsi que de la date.

   - Que contient la variable %USERPROFILE% ?

     Cette variable contient le chemin complet vers le dossier personnel de l'utilisateur actuellement connecté sur Windows.

   - Que contient la variable %BACKUP_DIR% après son exécution ?

     Après son exécution elle contient le chemin complet du dossier personnel de l'utilisateur, suivi du dossier Backup_ ainsi que la date au format JJ/MM/AAAA.

3. Que fait mkdir "%BACKUP_DIR%" ?

   Cette commande créer un dossier dont le chemin est contenu dans la variable %BACKUP_DIR% .

   - Que se passerait si le dossier existait déjà ?

     La commande ne fait rien, le dossier reste tel quel et le script continue normalement, comme si le dossier avait été créé.

4. Que fait xcopy "%USERPROFILE%\Documents*.txt" "%BACKUP_DIR%\Textes\" /S /I /Y ?

   Cette commande copie les fichiers de type .txt présents dans le chemin d'accès de l'utilisateur et dans le dossier Documents vers le dossier Backup_ et son sous dossier Textes.

   - Que signifient les options /S /I /Y ?

     /S permet de copier tous les fichiers dans le dossier source et dans ses sous-dossiers, sauf les dossiers vides.

     /I permet de considérer que la destination s'agit d'un dossier si elle n'existe pas (sinon xcopy demanderait une confirmation).

     /Y permet de supprimer la demande de confirmation avant d'écraser des fichiers existants à la destination (copie silencieuse).

5. Pourquoi crée-t-on deux dossiers différents (Textes et PDF) ?

   On créer deux dossiers différents afin de séparer les types de fichier et de s'organiser pour simplifier les sauvegardes, restaurations ou nettoyages.

6. Que fait la commande dir "%BACKUP_DIR%" /S ?

   Cette commande affiche à l'écran la liste complète de tous les fichiers et dossiers présents dans %BACKUP_DIR% et dans tous ses sous-dossiers, avec leurs détails (taille, date, heure, etc.).

7. Quel est le rôle de pause ? Que se passerait si on le supprimait ?

Son rôle est d'arrêter de script et d'attendre que l'utilisateur appuie sur une touche pour continuer ou pour fermer la fenêtre.

Si on le supprimerait le script se fermerait avec l'invite de commande à la fin de son exécution.

**Améliorations**

1. Comment pourriez-vous **sauvegarder d'autres types de fichiers** (images, vidéos) ?

   En ajoutant d'autres commandes xcopy avec d'autres types de fichiers comme .jpg ou .mp4 .

2. Comment rendre le script **plus robuste** si un fichier est en cours d'utilisation ou s'il y a un problème de droits ?

   Tout d'abord il faudrait exécuter le script en tant qu'administrateur et utiliser la commande robocopy qui est plus robuste et permet de gérer les fichiers verrouillés, avec des options de retry .

3. Comment automatiser ce script pour qu'il **s'exécute tous les jours** sans intervention manuelle ?

   On pourrait automatiser ce script à l'aide du planificateur de tâches Windows.

**5.2 Surveillance système**

Créez monitor.bat :

```
@echo off
:loop
cls
echo === MONITORING SYSTEME ===
echo Date: %DATE% - Heure: %TIME%
echo.
echo === PROCESSUS ===
tasklist /FO TABLE | findstr /I "chrome firefox"
echo.
echo === CONNEXIONS RESEAU ===
netstat -an | find ":80"
timeout /t 10 /nobreak
goto loop
```

1. Que fait la commande @echo off ? Pourquoi est-elle utile dans ce script ?

   @echo off permet de cacher la liste de commande qui s'exécute dans le script, elle est utile pour rendre le tout lisible et surtout dans notre cas le script s'actualise tout les 10 secondes et tourne en boucle.

   Tout d'abord elle permet d'afficher la DATE et L'HEURE.

   Ensuite elle affiche une liste de processus ciblant chrome et firefox.

   En dernier elle affiche une liste de connexions réseau sur le port 80

   Le script s'actualise toute les 10 secondes

2. Quelle est la fonction de l'étiquette :loop et du goto loop ?

   La fonction :loop et goto loop permet de créer une boucle dans le script.

3. Que se passe-t-il quand on exécute cls ? Pourquoi est-ce important dans ce script ?

   Quand on exécute cls cela nettoie l'invite de commande à chaque itération de la boucle, c'est important dans notre script pour rendre le tout plus lisible et d'enlever les anciennes informations qui peuvent être erronées car le script s'actualise tout les 10 secondes.

4. Que vont afficher %DATE% et %TIME% ? Que se passerait si on supprimait cette ligne ?

   %DATE% affiche la date du jour et %TIME% l'heure actuelle.

   Si on supprime cette ligne, on ne sait pas à quel moment précis les informations sur les processus et connexions réseau ont été affichées.

5. À quoi sert tasklist /FO TABLE | findstr /I "chrome firefox" ?

   Cette commande permet d'afficher une liste de processus sous forme de tableau avec un filtre permettant de conserver uniquement les processus de chrome et firefox.

   - Que se passerait si on remplaçait "chrome firefox" par "notepad" ?

     Cela afficherait uniquement les processus concernant le bloc-notes de Windows.

6. Que fait la commande netstat -an | find ":80" ? Que signifie le :80 ?

   Cette commande affiche toutes les connexions réseau actives et les ports d'écoute sur la machine, avec les adresses IP et les numéros de ports, en mode numérique (-n) et toutes (-a).

   Le :80 fait référence au port 80, qui est le port standard utilisé par le protocole HTTP (le web non sécurisé).

7. Quel est le rôle de timeout /t 10 /nobreak ? Que se passerait si on mettait /t 0 ?

   La commande timeout /t /10 /nobreak ralentit la boucle pour qu'elle affiche les infos toutes les 10 secondes, sans que l'utilisateur puisse interrompre cette pause.

   Si on mettait /t 0 cela indiquerait pas de pause au sein de la boucle ce qui veut dire que cela peut devenir illisible et surcharger le CPU aussi.

8. Pourquoi le script utilise-t-il une boucle infinie ? Quels sont les avantages et inconvénients d'un tel fonctionnement ?

   Le script utilise une boucle infinie afin d'actualiser les informations en tant réel car d'un instant à l'autre de nouvelles connexions réseau peuvent apparaître par exemple.

   Son principal avantage dans notre cas c'est son actualisation qui permet d'avoir une surveillance continue.

Les inconvénients sont l'utilisation excessive du processeur (CPU) et que le script ne s'arrête jamais il tourne en continue sauf quand on l'arrête manuellement avec CTRL + C par exemple.

**Améliorations**

1. Comment pourriez-vous modifier le script pour surveiller **plusieurs ports réseau** en même temps, par exemple 80 et 443 ?

   Pour surveiller plusieurs ports réseau en même temps, on peut utiliser cette commande : *netstat -an | findstr /C :":80" /C :":443"*

2. Comment pourriez-vous faire en sorte que le script **enregistre l'historique** des processus et connexions dans un fichier texte pour consultation ultérieure ?

   On peut créer une variable : *"LOGFILE=%~dp0monitor_log.txt" et l'initialiser à la fin juste avant le timeout avec %LOGFILE% .*

3. Comment pourriez-vous modifier le script pour qu'il **s'arrête automatiquement** après un certain nombre de cycles ou après un certain temps ?

   On peut également créer une varible : set /a counter=0 et d'ajouter ces deux lignes :

   set /a counter+=1

   if %counter%==5 goto end

   juste avant le timeout , en l'occurence le script s'arrêtera au bout de 5 cycles (5 goto end).

4. Que pourriez-vous faire pour rendre l'affichage **plus lisible ou esthétique**, par exemple avec des couleurs différentes pour les sections ?

   On peut rajouter la commande color avant chaque section exemple :

   **color 0a**

   **echo === MONITORING SYSTEME ===**

   **echo Date: %DATE% - Heure: %TIME%**

   **echo.**

   La section « Monitoring Systeme » sera désormais de couleur verte.