Préface

Depuis les origines de l'informatique, la sécurité n'a jamais été une option, mais une nécessité vitale. Face à la sophistication croissante des attaques, nombreux sont ceux qui pensent qu'il faut des outils complexes, coûteux ou réservés aux initiés pour défendre un système. Pourtant, un trésor sommeille déjà dans chaque machine Windows : l'invite de commandes, plus connue sous le nom de **CMD**.

Ce livre est né de cette conviction simple mais puissante : on peut sécuriser, analyser, comprendre et renforcer son système grâce à des outils natifs, gratuits et souvent oubliés. **CMD** n'est pas seulement une relique du passé. Elle est, entre des mains formées et vigilantes, une sentinelle puissante, rapide et silencieuse.

En tant qu'auteur passionné par les questions de souveraineté numérique, d'autodéfense informatique et de vulgarisation des savoirs techniques, j'ai voulu créer un

pont entre ceux qui débutent dans le monde de la cybersécurité et ceux qui cherchent à approfondir leur maîtrise, sans être noyés dans des langages ou frameworks trop lourds.

Ce livre est aussi une manière de démystifier **CMD**, de l'humaniser, de montrer qu'il n'est pas nécessaire d'être un hacker dans un film hollywoodien pour comprendre comment fonctionne la sécurité sur un PC. Il suffit de curiosité, de rigueur, et d'un bon guide.

J'espère que ces pages vous inspireront autant qu'elles vous formeront. Que vous soyez étudiant, technicien, enseignant, passionné, ou simple utilisateur prudent, ce livre vous appartient.

Mbida Mbida Camille

Introduction générale:

Dans un monde numérique où les menaces évoluent constamment, la sécurité informatique est devenue une préoccupation majeure pour les particuliers comme pour les entreprises. Bien que des outils graphiques sophistiqués soient disponibles, le bon vieux **CMD (Invite de commandes)** de Windows reste un allié puissant et souvent sous-estimé pour diagnostiquer, surveiller et sécuriser votre système.

Ce livre vous guidera à travers les capacités de CMD, en transformant cet outil en apparence basique en un instrument essentiel de votre arsenal de cybersécurité. Que vous soyez un professionnel de la sécurité, un administrateur système ou simplement un utilisateur soucieux de protéger ses données, vous découvrirez comment maîtriser les commandes et les scripts pour renforcer la posture de sécurité de votre environnement Windows.

Nous explorerons des concepts fondamentaux jusqu'aux techniques avancées, en vous fournissant des exemples pratiques et des scripts prêts à l'emploi. Préparez-vous à débloquer le potentiel caché de CMD et à prendre le contrôle de votre sécurité informatique.

Chapitre 1 : Qu'est-ce que CMD :

L'Invite de commandes (CMD), souvent appelée simplement "CMD" ou "ligne de commande", est une interface utilisateur textuelle qui permet d'exécuter des programmes et de saisir des commandes. Présente dans toutes les versions de Windows depuis ses débuts, elle est l'héritière directe des systèmes d'exploitation basés sur DOS. Contrairement aux interfaces graphiques (GUI) qui utilisent des icônes et des fenêtres, CMD fonctionne uniquement avec du texte.

Historique et importance

CMD tire ses racines de COMMAND.COM des systèmes MS-DOS. Malgré l'avènement des interfaces graphiques, Microsoft a conservé et amélioré CMD en raison de sa flexibilité, de sa puissance et de sa capacité à automatiser des tâches. Pour les administrateurs système et les professionnels de la sécurité, CMD est un outil indispensable pour les opérations qui nécessitent précision, rapidité et scripting.

Comment accéder à CMD:

Vous pouvez ouvrir l'Invite de commandes de plusieurs façons :

- Recherche Windows : Tapez "cmd" dans la barre de recherche Windows et sélectionnez "Invite de commandes".
- Menu Démarrer : Allez dans "Accessoires Windows" et cliquez sur "Invite de commandes".
- Exécuter (Win+R) : Tapez "cmd" et appuyez sur Entrée.

Pour des opérations de sécurité, il est souvent nécessaire d'exécuter CMD en tant qu'administrateur. Faites un clic droit sur "Invite de commandes" et choisissez "Exécuter en tant qu'administrateur". Cela confère les privilèges nécessaires pour modifier des paramètres système sensibles.

Premières commandes essentielles:

Voici quelques commandes de base pour vous familiariser avec l'environnement :

- **cls** : Efface l'écran de la console.

- **dir** : Affiche le contenu du répertoire courant.
- **cd** : Change de répertoire (**ex**: cd C:\Windows).
- **help** : Affiche une liste des commandes disponibles.
- **exit** : Ferme l'Invite de commandes.

Ces commandes fondamentales sont les briques de base sur lesquelles nous construirons des opérations de sécurité plus complexes.

Chapitre 2 : Navigation dans l'environnement Windows

Une maîtrise de la navigation dans les dossiers et les fichiers est cruciale pour utiliser CMD efficacement.

Commandes de base de navigation:

- **cd <chemin>** : Permet de changer de répertoire.
  - **cd ..** : Remonte au répertoire parent.
  - **cd ** : Ramène à la racine du lecteur.
  - **cd C:\Users\VotreNom\Documents** : Se déplace vers un chemin spécifique.

- **dir <options>** : Liste le contenu d'un répertoire.
  - **dir /w** : Affiche les fichiers et dossiers en largeur.
  - **dir /s** : Inclut les sous-répertoires.

- **dir /a:h** : Affiche les fichiers cachés.
- **dir *.log** : Recherche tous les fichiers avec l'extension .log.

Gestion des fichiers et répertoires:

- **mkdir <nom_repertoire>** : Crée un nouveau répertoire.
- **rmdir <nom_repertoire> ou rd <nom_repertoire>** : Supprime un répertoire vide.
- **rmdir /s /q <nom_repertoire>** : Supprime un répertoire non vide sans confirmation.
- **copy <source> <destination>** : Copie des fichiers.
- **copy C:\temp\fichier.txt D:\backup**
- **move <source> <destination>** : Déplace des fichiers ou des répertoires.
- **del <fichier>** : Supprime des fichiers.
- **del /q *.tmp** : Supprime tous les fichiers temporaires sans confirmation.
- **ren <ancien_nom> <nouveau_nom>** : Renomme un fichier ou un répertoire.

- **type <fichier>** : Affiche le contenu d'un fichier texte.

Les chemins absolus et relatifs:

- **Chemin absolu** : Spécifie le chemin complet depuis la racine du lecteur (**ex: C:\Windows\System32**).

- **Chemin relatif** : Spécifie le chemin par rapport au répertoire courant (**ex: ..\Documents**).

Comprendre ces concepts est fondamental pour interagir avec le système de fichiers de manière précise.

Chapitre 3 : Automatiser les tâches simples:

L'une des forces de CMD est sa capacité à automatiser des tâches répétitives grâce aux scripts batch (**.bat** ou **.cmd**).

Notions de base des scripts batch:

Un script batch est un fichier texte contenant une série de commandes CMD qui sont exécutées séquentiellement.

- **Création** : Ouvrez un éditeur de texte (Bloc-notes), tapez vos commandes et enregistrez le fichier avec l'extension .bat ou .cmd.

- **Exécution** : Double-cliquez sur le fichier ou exécutez-le depuis CMD.

Commandes utiles pour les scripts:

- **echo <message>** : Affiche un message à l'écran.

  - **echo off** : Désactive l'affichage des commandes pendant l'exécution.

- **pause** : Met le script en pause et attend une touche de l'utilisateur.

- **rem <commentaire>** : Ajoute un commentaire dans le script (non exécuté).

- **call <script_autre>** : Exécute un autre script batch et revient au script appelant.

- **start <programme>** : Démarre un programme ou un fichier.

Variables d'environnement:

Les variables d'environnement sont des valeurs dynamiques qui affectent la manière don't les processus s'exécutent.

- **echo %PATH%** : Affiche le chemin de recherche des exécutables.

- **set <variable>=<valeur>** : Définit une nouvelle variable.

- **set PATH=%PATH%;C:\mon_outil** : Ajoute un chemin au PATH existant.

Exemple de script d'automatisation (Nettoyage de fichiers temporaires):

```
@echo off
```

```
Echo Nettoyage des fichiers temporaires…

Del /s /q %TEMP%\*.*

Del /s /q C:\Windows\Temp\*.*

Echo Nettoyage termine.

Pause
```

Ce script simple supprime les fichiers temporaires du profil utilisateur et du répertoire Windows. De tels scripts peuvent être utilisés pour des tâches de maintenance régulières, améliorant ainsi la performance et potentiellement la sécurité en éliminant des fichiers superflus qui pourraient être exploités.

Chapitre 4 : Diagnostic système avec CMD

CMD offre des outils puissants pour diagnostiquer l'état de votre système, ce qui est une première étape cruciale pour identifier d'éventuels problèmes de sécurité.

Informations système détaillées

- **systeminfo** : Affiche des informations détaillées sur le système d'exploitation, la configuration matérielle, les mises à jour installées, etc. C'est une commande très utile pour un audit rapide.

Systeminfo | findstr /B /C:"Nom de l" /C:"Version du syst" /C:"Nom d"

Cette commande filtre les informations pour afficher le nom de l'OS, sa version et le nom de l'hôte.

Gestion des disques et du système de fichiers

- **chkdsk** : Vérifie l'intégrité du système de fichiers et corrige les erreurs.

- **chkdsk C: /f** : Vérifie le lecteur C: et corrige les erreurs trouvées. Utile pour détecter des corruptions de disque qui pourraient masquer des activités malveillantes.

- **fsutil volume diskfree C:** : Affiche l'espace libre sur le lecteur C:.

- **sfc /scannow** : Vérifie et répare les fichiers système Windows protégés corrompus ou modifiés. Indispensable pour s'assurer qu'aucun fichier système n'a été altéré par un logiciel malveillant.

Diagnostic réseau basique (local)

Bien que la surveillance réseau soit abordée plus en détail plus tard, voici quelques commandes de diagnostic de base :

- **ipconfig /all** : Affiche la configuration IP détaillée de toutes les cartes réseau, y compris les adresses MAC, les serveurs DNS, etc. Utile pour vérifier si les paramètres réseau ont été modifiés de manière inattendue.

- **ping <adresse_ip_ou_domaine>** : Vérifie la connectivité à un hôte distant.

- **tracert <adresse_ip_ou_domaine>** : Trace le chemin vers un hôte distant.

Ces commandes fournissent une image claire de l'état de santé de votre système, permettant de détecter des anomalies qui pourraient indiquer une compromission ou des vulnérabilités.

Chapitre 5 : Surveillance des processus et tâches

Surveiller les processus en cours d'exécution et les tâches planifiées est essentiel pour détecter les activités suspectes.

Afficher et gérer les processus

- **tasklist** : Liste tous les processus en cours d'exécution sur le système, avec leur PID (identifiant de processus), le nom de l'image, l'utilisation de la mémoire, etc.

Tasklist /svc

Cette commande affiche les services associés à chaque processus. Utile pour identifier les processus sans service légitime.

- **taskkill /pid <pid> ou taskkill /im <nom_image>** : Termine un processus.

  - **taskkill /f /im notepad.exe** : Force la fermeture de Notepad. Utile pour stopper des processus malveillants identifiés.

Surveillance des tâches planifiées

Les tâches planifiées peuvent être utilisées par des attaquants pour maintenir la persistance sur un système.

- **schtasks /query /fo LIST /v** : Affiche toutes les tâches planifiées avec des informations détaillées.

  - Examinez attentivement les tâches inconnues ou celles qui s'exécutent avec des privilèges élevés.

- **schtasks /create /tn "MaTache" /tr "C:\Scripts\monscript.bat" /sc DAILY /st 09:00** : Crée une nouvelle tâche planifiée.

- **schtasks /delete /tn "MaTache" /f** : Supprime une tâche planifiée.

Identification des processus suspects

Lors de la revue de **tasklist**, recherchez :

- Noms de processus inconnus : Des noms étranges ou mal orthographiés.

- Processus consommant beaucoup de ressources : Une utilisation CPU ou mémoire anormalement élevée sans raison apparente.

- Processus s'exécutant à partir de répertoires inhabituels : Les logiciels malveillants se cachent souvent

dans des dossiers temporaires ou des répertoires utilisateur non standards.

- Processus sans signature numérique : Bien que CMD ne le montre pas directement, des outils externes ou PowerShell peuvent vérifier cela.

Une surveillance régulière des processus est une ligne de défense essentielle contre les logiciels malveillants et les intrusions.

Chapitre 6 : Surveillance réseau locale

Comprendre et surveiller votre activité réseau locale est crucial pour détecter les communications suspectes.

Afficher les connexions réseau actives

- **netstat -ano** : Affiche toutes les connexions réseau actives (TCP et UDP), les ports d'écoute, l'adresse distante, l'état de la connexion et le PID du processus qui a établi la connexion.

Netstat -ano | findstr ESTABLISHED

Cette commande filtre pour n'afficher que les connexions établies, souvent indicatives d'une communication active. Utilisez le **PID** obtenu avec **tasklist** pour identifier le programme.

Vérifier les adresses IP et les tables de routage

- **ipconfig /all** : Comme mentionné précédemment, pour vérifier votre configuration IP.

- **route print** : Affiche la table de routage IP de votre machine. Des routes inattendues peuvent indiquer une manipulation par un attaquant pour rediriger le trafic.

Afficher les informations de résolution DNS

- **ipconfig /displaydns** : Affiche le cache du résolveur DNS. Utile pour voir les noms de domaine récemment résolus et détecter des résolutions vers des serveurs DNS malveillants ou des domaines de C&C (Command and Control).

- **ipconfig /flushdns** : Vide le cache DNS.

Identification des communications suspectes

Lors de l'analyse de **netstat -ano**, recherchez :

- Connexions à des adresses IP inconnues ou suspectes : Des bases de données comme VirusTotal ou AbuseIPDB peuvent aider à vérifier la réputation d'une IP.

- Connexions sur des ports inhabituels : Des applications légitimes utilisent généralement des ports standard (80, 443, 21, 22, 3389, etc.). Une connexion sur un port aléatoire ou rarement utilisé est suspecte.

- Processus inconnus effectuant des connexions réseau : Si un processus que vous ne reconnaissez pas établit des connexions, c'est un signe d'alerte.

La surveillance réseau locale est votre première ligne de défense pour détecter les communications sortantes d'un système compromis.

Chapitre 7 : Audits de sécurité internes

CMD peut être un outil précieux pour réaliser des audits de sécurité de base sur votre système.

Vérification des privilèges utilisateur

- **whoami /priv** : Affiche les privilèges de sécurité du compte utilisateur courant. Permet de vérifier quels droits un utilisateur possède réellement.

- **whoami /groups** : Affiche les groupes de sécurité auxquels l'utilisateur courant appartient.

Vérification des utilisateurs et groupes locaux

- **net user** : Liste tous les comptes utilisateurs locaux sur le système.

- **net user <nom_utilisateur>** : Affiche les informations détaillées sur un utilisateur spécifique.

  - Recherchez des comptes utilisateurs inconnus ou inactifs qui pourraient être des portes dérobées.

- **net localgroup** : Liste tous les groupes locaux.

- **net localgroup Administrators** : Affiche les membres du groupe Administrateurs. C'est crucial pour s'assurer que seuls les utilisateurs légitimes ont des privilèges administratifs.

Audit des partages réseau

- **net share** : Liste tous les partages réseau actifs sur le système.

  - Vérifiez s'il y a des partages inattendus ou mal configurés qui pourraient exposer des données sensibles.

- **net view \\<nom_ordinateur>** : Affiche les partages d'un ordinateur distant.

Vérification des mises à jour Windows

Bien que CMD ne permette pas directement d'installer des mises à jour, vous pouvez vérifier certaines informations.

- **systeminfo | findstr /B /C:"Correctif(s) logiciel(s)"** : Liste les mises à jour installées. Comparez-les aux

dernières mises à jour de sécurité disponibles pour Windows. Un système non à jour est une cible facile.

Ces commandes fournissent une base solide pour un audit de sécurité initial, vous aidant à identifier les configurations faibles ou les comptes compromis.

Chapitre 8 : Sécuriser les accès et les comptes

La gestion et la sécurisation des comptes utilisateurs sont primordiales pour empêcher les accès non autorisés.

Changer les mots de passe

- **net user <nom_utilisateur> <nouveau_mot_de_passe>** : Permet de changer le mot de passe d'un utilisateur local.

Net user JohnDoe P@ssw0rd123!

Il est impératif d'utiliser des mots de passe forts et uniques.

Désactiver ou supprimer des comptes

- **net user <nom_utilisateur> /active:no** : Désactive un compte utilisateur sans le supprimer. Le compte ne peut plus être utilisé pour se connecter.

- **net user <nom_utilisateur> /delete** : Supprime définitivement un compte utilisateur. Utilisez cette commande avec prudence.

Gérer les membres des groupes

- **net localgroup <nom_groupe> <nom_utilisateur> /add** : Ajoute un utilisateur à un groupe local.

- **net localgroup <nom_groupe> <nom_utilisateur> /delete** : Supprime un utilisateur d'un groupe local.

Net localgroup Administrators JohnDoe /delete

Cette commande retire l'utilisateur JohnDoe du groupe Administrateurs, réduisant ainsi ses privilèges.

Appliquer des politiques de mots de passe

Bien que la configuration des politiques de groupe (**gpedit.msc** ou **secpol.msc**) soit généralement effectuée via l'interface graphique, il est bon de savoir que ces politiques influencent CMD. Pour un système autonome, vous pouvez vérifier la politique de mot de passe actuelle avec :

- **net accounts** : Affiche les paramètres des politiques de compte, y compris la longueur minimale du mot de passe et l'âge du mot de passe.

  - Exigez des mots de passe complexes, une longueur minimale et des changements réguliers.

Une gestion rigoureuse des comptes et des privilèges est une pierre angulaire de la sécurité informatique.

Chapitre 9 : Configuration du pare-feu via CMD

Le Pare-feu Windows est une composante essentielle de la sécurité. CMD vous permet de le configurer avec précision.

Vérifier l'état du pare-feu

- **netsh advfirewall show allprofiles** : Affiche l'état du pare-feu pour tous les profils (domaine, privé, public).

- **netsh advfirewall show currentprofile** : Affiche l'état du profil actuel.

Activer/Désactiver le pare-feu

- **netsh advfirewall set currentprofile state on** : Active le pare-feu pour le profil courant.

- **netsh advfirewall set currentprofile state off** : Désactive le pare-feu pour le profil courant (à n'utiliser qu'en cas de dépannage et jamais sur un système en production).

Ajouter des règles de pare-feu

- Règle d'entrée (Inbound Rule) : Autoriser une application

Netsh advfirewall firewall add rule name="Autoriser Apache" dir=in action=allow program="C:\Apache24\bin\httpd.exe" enable=yes

- Règle de sortie (Outbound Rule) : Bloquer une adresse IP et un port

Netsh advfirewall firewall add rule name="Bloquer IP Malveillante" dir=out action=block remoteip=192.168.1.100 protocol=any enable=yes

- Autoriser un port spécifique :

Netsh advfirewall firewall add rule name="Autoriser HTTP" dir=in action=allow protocol=TCP localport=80

Supprimer des règles de pare-feu

- **netsh advfirewall firewall delete rule name="Autoriser Apache"** : Supprime une règle par son nom.

- **netsh advfirewall firewall delete rule name=all** : Supprime toutes les règles (à utiliser avec une extrême prudence !).

Exporter/Importer la configuration du pare-feu

- **netsh advfirewall export "C:\backup\firewall_config.wfw"** : Exporte la configuration actuelle du pare-feu.

- **netsh advfirewall import "C:\backup\firewall_config.wfw"** : Importe une configuration du pare-feu.

La gestion du pare-feu via CMD offre un contrôle granulaire et la possibilité d'automatiser le déploiement de règles de sécurité.

Chapitre 10 : Protéger les fichiers sensibles

Protéger vos fichiers et dossiers sensibles est une priorité absolue. CMD propose plusieurs façons d'y parvenir.

Gestion des permissions de fichiers (ACL)

La commande **icacls** permet de visualiser et de modifier les listes de contrôle d'accès (ACL) des fichiers et des dossiers.

- Afficher les permissions :

Icacls C:\DossierSensible

Cela affichera qui a quels droits (lecture, écriture, exécution, modification, contrôle total).

- Supprimer toutes les permissions héritées et accorder un accès spécifique :

Icacls C:\DossierSensible /inheritance:d /grant UtilisateurA:(OI)(CI)F

/inheritance:d désactive l'héritage. **(OI)(CI)F** accorde un contrôle total (**F**) au dossier, aux sous-dossiers (**CI**) et aux fichiers (**OI**) pour **UtilisateurA**.

- Refuser l'accès à un utilisateur :

Icacls C:\DossierSensible /deny ToutLeMonde:(OI)(CI)F

Cette commande est très puissante, utilisez-la avec prudence. Les règles de refus ont priorité.

- Réinitialiser les permissions :

Icacls C:\DossierSensible /reset

Restaure les ACLs à leurs valeurs héritées par défaut.

Chiffrement de fichiers (EFS)

Le Système de fichiers de chiffrement (EFS) permet de chiffrer des fichiers individuels ou des dossiers.

- **cipher /e <fichier_ou_dossier>** : Chiffre un fichier ou un dossier.

- **cipher /d <fichier_ou_dossier>** : Déchiffre un fichier ou un dossier.

- **cipher /u /n** : Met à jour les clés de chiffrement de l'utilisateur.

Le chiffrement EFS est lié au compte utilisateur et nécessite une gestion attentive des certificats de récupération.

Masquer les fichiers

- **attrib +h <fichier>** : Rend un fichier caché.
- **attrib -h <fichier>** : Rend un fichier visible.
  - Attention : Le masquage n'est pas une mesure de sécurité robuste, car les fichiers cachés sont facilement visibles en affichant les fichiers cachés dans l'Explorateur Windows ou avec dir /a:h.

Utilisez une combinaison de permissions robustes et, si nécessaire, de chiffrement pour protéger efficacement vos données sensibles.

Chapitre 11 : Détection de programmes suspects

Détecter les programmes malveillants ou suspects est une étape cruciale de la défense. CMD, bien que limité par rapport à des outils spécialisés, offre des pistes.

Analyser les processus en cours

- Revisitez la commande **tasklist /svc**. Recherchez :
  - Des processus avec des noms de fichiers étranges (ex: **aBcDeF.exe**).
  - Des processus sans description ni informations sur l'éditeur.
  - Des processus consommant des ressources de manière anormale.
  - Des processus qui ne devraient pas s'exécuter avec des privilèges élevés.
- Utilisez **wmic process get Caption,CommandLine,ProcessId,ParentProcessId** pour

voir la ligne de commande complète des processus, ce qui peut révéler des paramètres suspects.

Examiner les points d'exécution automatique (Autoruns)

Les logiciels malveillants s'installent souvent dans des emplacements qui leur permettent de démarrer avec Windows.

- Clés de registre de démarrage :
  - **reg query HKLM\SOFTWARE\Microsoft\Windows \CurrentVersion\Run**
  - **reg query HKLM\SOFTWARE\Microsoft\Windows \CurrentVersion\RunOnce**
  - **reg query HKCU\SOFTWARE\Microsoft\Windows \CurrentVersion\Run**
  - **reg query HKCU\SOFTWARE\Microsoft\Windows \CurrentVersion\RunOnce**
  - Recherchez des entrées qui pointent vers des exécutables inconnus ou des emplacements suspects.

- Dossiers de démarrage :

- **dir "%APPDATA%\Microsoft\Windows\Start Menu\Programs\Startup"**
- **dir "C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Startup"**
- Vérifiez si des raccourcis vers des applications inconnues s'y trouvent.

- Services Windows :
  - **sc query state= all** : Liste tous les services. Recherchez les services inconnus ou ceux avec un type de démarrage "Automatique" qui ne sont pas reconnus.
  - **sc qc <nom_service>** : Affiche la configuration d'un service, y compris le chemin de l'exécutable.

Vérifier les extensions de fichiers

Les attaquants peuvent masquer des fichiers exécutables en leur donnant des extensions inattendues (ex: **.txt.exe**).

- Soyez vigilant lors de l'ouverture de fichiers.

- Assurez-vous que l’affichage des extensions de fichiers est activé dans l’Explorateur Windows.

La détection de programmes suspects est un processus continu qui combine l’analyse des processus, des points de persistance et une vigilance générale.

Chapitre 12 : Création d'un environnement de test

Avant d'appliquer des techniques de sécurité sur un système de production, il est impératif de les tester dans un environnement isolé.

Pourquoi un environnement de test ?

- **Sécurité** : Empêche d'endommager accidentellement un système réel.
- **Apprentissage** : Permet d'expérimenter et de comprendre le comportement des commandes.
- **Reproductibilité** : Permet de reproduire des scénarios d'attaque ou de défense.
- **Développement de scripts** : Permet de créer et de déboguer des scripts sans risque.

Options pour un environnement de test

Machines virtuelles (VM) :

- Utilisez des logiciels comme **Oracle VirtualBox** (gratuit), **VMware Workstation Player** (gratuit pour usage personnel) ou **Hyper-V** (intégré à Windows Pro/Enterprise).

- Installez une version propre de Windows (par exemple, Windows 10 ou 11) à l'intérieur de la VM.

- **Snapshots** : Les VMs permettent de prendre des "instantanés" (snapshots) de l'état du système. C'est crucial : vous pouvez expérimenter, et si quelque chose tourne mal, revenir à un état précédent en un clic.

Machines physiques dédiées (moins commun pour les particuliers) :

- Utiliser un vieil ordinateur que vous pouvez réinitialiser sans souci. Moins flexible que les VMs.

Configuration du laboratoire de test

- **Installation propre de Windows** : Commencez toujours par une installation propre pour éviter des interférences inattendues.

- **Isolation réseau** : Si vous simulez des attaques réseau, configurez les VMs pour qu'elles soient sur un réseau interne ou isolé, sans accès à votre réseau principal ou à Internet, sauf si cela fait partie du scénario de test.

- **Outils essentiels** : Installez uniquement les outils nécessaires pour le test.

- **Documentation** : Gardez une trace des modifications que vous apportez à l'environnement.

Un environnement de test bien configuré est la pierre angulaire d'un apprentissage et d'une expérimentation sécurisés en cybersécurité.

Chapitre 13 : Scripts CMD de détection automatique

Automatiser la détection des menaces avec des scripts CMD peut grandement améliorer votre capacité de réponse.

Script de surveillance des processus suspects

Ce script peut être planifié pour s'exécuter régulièrement.

```
@echo off

Echo Surveillance des processus suspects…

Echo.

Set
LOGFILE=C:\Logs\process_audit_%date:~6,4%_%date:~3,
2%_%date:~0,2%_%time:~0,2%%time:~3,2%%time:~6,2
%.log

Echo Date/Heure: %date% %time% >> %LOGFILE%
```

Echo ---------- >> %LOGFILE%

Tasklist /svc /fo list > %TEMP%\current_processes.txt

Findstr /V /I /L /C:"svchost.exe" /C:"explorer.exe" /C:"csrss.exe" /C:"winlogon.exe" /C:"lsass.exe" /C:"smss.exe" /C:"dwm.exe" /C:"RuntimeBroker.exe" /C:"cmd.exe" %TEMP%\current_processes.txt >> %LOGFILE%

Echo. >> %LOGFILE%

Echo Connexions reseau suspectes (ports non standards ou IP externes inconnues)… >> %LOGFILE%

Netstat -ano | findstr /V /I "127.0.0.1" | findstr /V /I "0.0.0.0" | findstr /V /I "::1" >> %LOGFILE%

Echo. >> %LOGFILE%

Echo Verifier les taches planifiees (nouvelles ou modifiees)… >> %LOGFILE%

Schtasks /query /fo LIST /v | findstr /I /C:"TaskName" /C:"TaskPath" /C:"Last Run Time" /C:"Last Result" >> %LOGFILE%

Echo. >> %LOGFILE%

Echo Verifier les points de demarrage (Run keys)… >> %LOGFILE%

Reg query HKLM\SOFTWARE\Microsoft\Windows \CurrentVersion\Run >> %LOGFILE%

Reg query HKCU\SOFTWARE\Microsoft\Windows \CurrentVersion\Run >> %LOGFILE%

Del %TEMP%\current_processes.txt

Echo Audit complete. Le rapport est disponible dans %LOGFILE%

Explication : Ce script enregistre les informations sur les processus, les connexions réseau, les tâches planifiées et les clés de démarrage dans un fichier journal. La ligne **findstr /V /I /L /C**: exclut les processus Windows courants. Vous devrez l'adapter à votre environnement pour exclure d'autres processus légitimes.

Script de vérification des sommes de contrôle de fichiers critiques

Utilisez cet outil pour calculer les sommes de contrôle (hashes) de fichiers importants. Stockez ces hashes de

référence et comparez-les régulièrement pour détecter des modifications non autorisées.

```
@echo off

Set “CRITICAL_FILE=C:\Windows\System32\cmd.exe”

Set “REFERENCE_HASH=VOTRE_HASH_REFERENCE_ICI”
REM ← Remplacez par le hash de référence original

Echo Verification de l’integrite de %CRITICAL_FILE%...

For /f “tokens=2,3” %%a in (‘certutil -hashfile “%CRITICAL_FILE%” MD5’) do (

   If “%%a”==”MD5” set CURRENT_HASH=%%b

)
```

If “%CURRENT_HASH%”==”%REFERENCE_HASH%” (

Echo Le fichier est integre. Hash actuel : %CURRENT_HASH%

) else (

Echo ALERTE: Le fichier a ete modifie ! Hash actuel : %CURRENT_HASH%, Hash de reference : %REFERENCE_HASH%

)

Pause

Pré-requis : Vous devez calculer le hash MD5 du fichier cmd.exe sur un système propre et le placer dans la

variable REFERENCE_HASH. Répétez pour d'autres fichiers critiques comme lsass.exe, explorer.exe, etc.

Ces scripts sont des points de départ. Adaptez-les à vos besoins spécifiques et intégrez-les dans vos routines de sécurité.

Chapitre 14 : Avancer vers PowerShell, WMI, Reg

CMD est puissant, mais pour une sécurité avancée, il est essentiel de connaître ses successeurs et compléments.

PowerShell : La nouvelle génération

PowerShell est la suite logique de CMD. C'est un environnement de script et d'automatisation beaucoup plus puissant et orienté objet.

- Pourquoi migrer ?
  - Accès à **.NET Framework** : Des fonctionnalités beaucoup plus riches.
  - **Cmdlets** : Des commandes spécialisées et intuitives (ex: Get-Process, Get-NetTCPConnection).
  - Pipeline : Permet de chaîner les commandes pour traiter les objets.
  - Accès à WMI et COM : Contrôle total du système.
  - Gestion à distance : Capacité à administrer des machines à distance.

- Exemple (équivalent tasklist avec plus de détails) :

Get-Process | Select-Object Name,Id,Path,StartTime,CPU,VM | Format-Table -AutoSize

Cette commande PowerShell est beaucoup plus riche en informations que tasklist.

WMI (Windows Management Instrumentation)

WMI est une technologie Microsoft qui fournit une interface unifiée pour gérer et surveiller les composants du système d'exploitation Windows et des applications. CMD peut interroger WMI via la commande wmic.

- Exemples d'utilisation de wmic pour la sécurité :
  - Processus avec chemin complet : **wmic process get description,executablepath**
  - Comptes locaux : **wmic useraccount get name,sid,disabled,installDate**

- Logiciels installés : **wmic product get name,version,vendor** (peut être lent)
- Services : **wmic service get name,state,startmode,pathname** (utile pour trouver les exécutables des services)
- Informations sur le système : **wmic computersystem get model,manufacturer,name**
- Contrôler des services : **wmic service where “name=’Spooler’” call stop**

REG (Commande de Registre)

La commande **reg** permet de manipuler la base de registre Windows directement depuis CMD. Le registre est un endroit clé où les logiciels malveillants peuvent persister et stocker des configurations.

- Afficher une clé/valeur : **reg query HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run /v “notepad.exe”**
- Ajouter une valeur : **reg add HKCU\Software\MyApp /v “Path” /t REG_SZ /d “C:\Program Files\MyApp\app.exe” /f**

- Supprimer une valeur : **reg delete HKCU\Software\MyApp /v “Path” /f**

- Supprimer une clé : **reg delete HKCU\Software\MyApp /f**

Bien que ce chapitre se concentre sur l’avancement, la compréhension de wmic et reg étend considérablement vos capacités de diagnostic et de gestion de la sécurité via CMD.

Chapitre 15 : Gestion avancée du réseau

Au-delà des diagnostics de base, CMD permet une gestion plus approfondie des interfaces réseau.

Gestion des interfaces réseau

- **netsh interface show interface** : Affiche la liste des interfaces réseau et leur statut.
- **netsh interface set interface "Ethernet" admin=disable** : Désactive une interface réseau par son nom.
- **netsh interface set interface "Ethernet" admin=enable** : Active une interface réseau.

Configuration IP avancée

- **netsh interface ip set address "Ethernet" static 192.168.1.100 255.255.255.0 192.168.1.1** : Configure une adresse IP statique.

- **netsh interface ip set dns “Ethernet” static 8.8.8.8 primary** : Définit un serveur DNS primaire.

- **netsh interface ip add dns “Ethernet” 8.8.4.4 index=2** : Ajoute un serveur DNS secondaire.

Ces commandes sont utiles pour verrouiller une configuration réseau après une compromission ou pour automatiser le déploiement de machines sécurisées.

Nettoyage du cache ARP

Le cache ARP (Address Resolution Protocol) peut être empoisonné par des attaques MITM (Man-in-the-Middle).

- **arp -a** : Affiche le cache ARP.

- **arp -d** : Supprime toutes les entrées du cache ARP. Utile pour vider les entrées malveillantes après une attaque ARP.

Vérification des ports ouverts (basique)

Bien que netstat liste les connexions, CMD n’a pas de scanner de ports intégré comme Nmap. Cependant, vous pouvez utiliser des techniques rudimentaires :

- Tentative de connexion (pour des ports spécifiques et manuellement) :

  - **telnet <adresse_ip> <port>** : Si Telnet client est installé, vous pouvez tester la connectivité à un port. Si la connexion réussit, le port est ouvert. Si ce n'est pas installé par défaut, activez-le via "Fonctionnalités de Windows".

  - **timeout /t 1 & echo > \\<adresse_ip>\ipc$** (rudimentaire pour vérifier l'accès au partage IPC$, souvent le port 445)

Pour des scans de ports plus robustes, des outils tiers sont nécessaires. Cependant, la gestion via CMD permet d'appliquer des règles de pare-feu précises pour contrôler les accès aux ports.

Chapitre 16 : Exploiter les journaux d'événements Windows

Les journaux d'événements Windows sont une mine d'informations cruciale pour la détection d'incidents et la forensique. CMD permet de les interroger et de les exporter.

La commande wevtutil

Wevtutil est l'outil en ligne de commande pour manipuler les journaux d'événements.

- Lister les journaux disponibles : **wevtutil el**
- Interroger un journal (ex: Sécurité) :

Wevtutil qe Security /f:text /c:10

Cette commande affiche les 10 dernières entrées du journal de sécurité.

- Filtrer les événements par ID :

Wevtutil qe Security /f:text /q:"*[System[(EventID=4624)]]"

Recherche tous les événements 4624 (connexion réussie).

- Filtrer par date/heure :

Wevtutil qe System /f:text /q:"*[System[TimeCreated[@SystemTime >='2025-06-20T00:00:00Z']]]"

Recherche les événements après une certaine date.

- Exporter un journal :

Wevtutil export-log Security "C:\Logs\security_log.evtx"

Exporte le journal de sécurité dans un fichier EVTX. Ce fichier peut être analysé sur une autre machine ou avec des outils spécialisés.

- Effacer un journal (avec prudence ! pour la forensique, toujours copier avant d'effacer) :

Wevtutil cl Security

Efface le journal de sécurité.

Événements de sécurité critiques à surveiller

- **4624** : Connexion réussie à un compte.
- **4625** : Échec de connexion à un compte.
- **4648** : Une tentative de connexion a été effectuée à l'aide des informations d'identification explicites (utile pour détecter des tentatives d'exécution en tant qu'autre utilisateur).
- **4672** : Les privilèges spéciaux ont été attribués à la nouvelle connexion (indique qu'un compte administrateur s'est connecté).
- **4720** : Un compte d'utilisateur a été créé.
- **4722** : Un compte d'utilisateur a été activé.
- **4723** : Un compte d'utilisateur a changé de mot de passe.
- **4732** : Un membre a été ajouté à un groupe de sécurité local (surveillez le groupe Administrateurs).

La maîtrise de wevtutil est indispensable pour toute analyse de sécurité sérieuse.

Chapitre 17 : Contenir une attaque via CMD

En cas d'incident de sécurité, la capacité à réagir rapidement est essentielle. CMD peut être utilisé pour des mesures de confinement d'urgence.

Déconnexion réseau

- Désactiver une interface réseau :

Netsh interface set interface "Ethernet" admin=disable

Cela coupera immédiatement la connectivité réseau de la machine, empêchant le malware de communiquer avec son serveur de commande et de contrôle (C2) ou de se propager.

- Bloquer les communications via le pare-feu : Si vous ne voulez pas couper totalement le réseau, bloquez les communications spécifiques :

Netsh advfirewall firewall add rule name="Blocage Urgence" dir=out action=block remoteip=any enable=yes

Cette règle bloque tout trafic sortant, mais permet au trafic entrant d'être géré par d'autres règles. À utiliser avec prudence.

Arrêter les processus malveillants

- **tasklist** pour identifier le PID du processus suspect.
- **taskkill /f /pid <PID>** : Force l'arrêt du processus.
- **taskkill /f /im <nom_image.exe>** : Force l'arrêt du processus par son nom (utile si plusieurs instances).

Supprimer les points de persistance

- Supprimer les tâches planifiées suspectes :
  - **schtasks /delete /tn "NomDeLaTacheSuspecte" /f**
- Supprimer les entrées de registre de démarrage malveillantes :
  - **reg delete HKLM\SOFTWARE\Microsoft\Windows \CurrentVersion\Run /v "NomDeL'entrée" /f**
- Supprimer les services malveillants :
  - Identifiez le nom du service avec **sc query** puis **sc delete "NomDuService"**

Isolation des fichiers suspects

- Déplacer les fichiers : Déplacez les exécutables malveillants vers un dossier isolé (par exemple, un dossier non exécutable et restreint en permissions, comme **C:\Quarantine**).

  - **move "C:\Users\Public\malware.exe" "C:\Quarantine\malware.exe"**

- Changer les permissions :

  - **icacls "C:\Quarantine\malware.exe" /deny Everyone☹RX) : Empêche l'exécution et la lecture par quiconque.**

Redémarrage ou arrêt d'urgence

- **shutdown /r /t 0** : Redémarre immédiatement le système.

- **shutdown /s /t 0** : Arrête immédiatement le système.

Ces commandes sont des outils de réponse rapide pour limiter les dommages d'une attaque en cours. Elles ne

remplacent pas une analyse forensique complète et une éradication.

Chapitre 18 : Études de cas pratiques

Appliquons les connaissances acquises à des scénarios réels.

Cas 1 : Détection d'un processus inconnu

Scénario : Votre ordinateur est lent et vous suspectez un logiciel malveillant.

1. **Vérifiez les processus** :

Tasklist /svc /fo list > processes.txt

Notepad processes.txt

Analysez le fichier **processes.txt**. Recherchez des noms étranges, des PID sans service, des consommations mémoire/CPU anormales.

2. **Analysez les connexions réseau du processus :**

Netstat -ano | findstr "<PID_suspect>"

Si le PID suspect est connecté à des adresses IP inconnues, c'est un signe fort.

3. **Vérifiez la localisation du fichier exécutable :**

Wmic process where "ProcessId='<PID_suspect>'" get ExecutablePath,CommandLine

Si le chemin est un répertoire temporaire ou utilisateur, c'est suspect.

4. **Tuez le processus :**

Taskkill /f /pid <PID_suspect>

Cas 2 : Audit d'un nouveau compte utilisateur

Scénario : Un nouvel utilisateur a été ajouté au système. Vous voulez vous assurer de ses privilèges.

1. **Lister les comptes :**

Net user

2. **Détails du compte :**

Net user <NomDuNouvelUtilisateur>

Vérifiez l'heure de dernière connexion, les groupes auxquels il appartient.

3. **Vérifier l'appartenance aux groupes critiques :**

Net localgroup Administrators

Net localgroup Remote Desktop Users

Assurez-vous qu'il n'est pas dans des groupes à privilèges élevés sans raison.

4. **Si nécessaire, ajustez les permissions :**

Net localgroup Administrators <NomDuNouvelUtilisateur>/delete

Cas 3 : Configuration du pare-feu pour un serveur web

Scénario : Vous installez un serveur web (Apache) et voulez ouvrir le port 80 et 443 tout en bloquant le reste.

1. **Activer le pare-feu (si ce n'est pas déjà fait) :**

Netsh advfirewall set currentprofile state on

2. **Ajouter des règles pour HTTP et HTTPS :**

Netsh advfirewall firewall add rule name="Autoriser HTTP" dir=in action=allow protocol=TCP localport=80

Netsh advfirewall firewall add rule name="Autoriser HTTPS" dir=in action=allow protocol=TCP localport=443

3. **Ajouter une règle pour le trafic sortant (si vous voulez restreindre)** : (N'ajoutez pas cette règle si vous n'avez pas d'autres règles de sortie spécifiques, cela bloquera tout !)

Netsh advfirewall firewall add rule name=”Bloquer Tout Sortant Sauf Specifique” dir=out action=block remoteip=any enable=yes

Puis, vous ajouteriez des règles allow pour les services spécifiques don’t votre serveur a besoin (DNS, mises à jour, etc.).

Ces études de cas illustrent comment les commandes CMD peuvent être enchaînées pour résoudre des problèmes de sécurité concrets.

Chapitre 19 : Intégration avec des outils externes (bonus)

Bien que ce livre se concentre sur CMD, il est important de savoir comment il peut interagir avec des outils tiers pour renforcer la sécurité.

Exécuter des outils d'analyse

Vous pouvez utiliser CMD pour lancer des scanners de virus, des outils de diagnostic réseau, ou des utilitaires de forensique.

- Lancer un scan antivirus :

"C:\Program Files\MonAntivirus\antivirus.exe" /scan C:

(Le chemin et les options varient selon l'antivirus).

- Lancer **Nmap** (scanner de ports) :

"C:\Program Files\Nmap\nmap.exe" -Pn 192.168.1.1/24

- Lancer **Sysinternals Suite** : Les outils Sysinternals (Process Explorer, Autoruns, PsExec, etc.) sont très puissants et peuvent être lancés depuis CMD.

“C:\SysinternalsSuite\procexp.exe”

Automatisation avec des outils externes

Les scripts CMD peuvent servir de “colle” pour orchestrer l’exécution d’outils externes et traiter leurs sorties.

- Exemple : Scannez avec Nmap, puis analysez le résultat avec un script Python.

```
@echo off
Set NMAP_PATH=”C:\Program Files\Nmap\nmap.exe”
Set TARGET_IP=192.168.1.1
Set OUTPUT_FILE=”C:\temp\nmap_scan.xml”
%NMAP_PATH% -Pn %TARGET_IP% -oX %OUTPUT_FILE%
Echo Nmap scan completed. Processing results…
```

Python C:\Scripts\parse_nmap.py %OUTPUT_FILE%

Echo Result processing finished.

Pause

Récupération de données avec des utilitaires spécialisés

En cas de perte de données ou de besoin de récupération forensique, des outils comme Recuva ou des utilitaires de disques peuvent être lancés via CMD.

L'intégration avec des outils externes transforme CMD en un puissant orchestrateur pour des opérations de sécurité plus complexes, combinant la puissance de la ligne de commande avec des fonctionnalités spécialisées.

Chapitre 20 : Scripts CMD pour la cybersécurité (modèle pro)

Ce chapitre propose des scripts plus élaborés et des modèles que vous pouvez adapter à vos besoins professionnels.

Modèle 1 : Audit de sécurité complet du système (Générateur de rapport)

Ce script collecte diverses informations et les consolide dans un rapport.

@echo off

Setlocal

Set “LOG_DIR=C:\Security_Audits”

If not exist “%LOG_DIR%” mkdir “%LOG_DIR%”

Set
“REPORT_FILE=%LOG_DIR%\Security_Audit_Report_%date:~6,4%_%date:~3,2%_%date:~0,2%_%time:~0,2%%time:~3,2%%time:~6,2%.txt”

Echo # Rapport d’Audit de Securite - %computername%

Echo Date et Heure de l’Audit: %date% %time%

Echo ----------

Echo. >> “%REPORT_FILE%”

Echo # Rapport d’Audit de Securite - %computername% >> “%REPORT_FILE%”

Echo Date et Heure de l’Audit: %date% %time% >> “%REPORT_FILE%”

Echo ---------- >> “%REPORT_FILE%”

Echo. >> “%REPORT_FILE%”

Echo ## Informations Systeme

Echo.

Systeminfo >> “%REPORT_FILE%”

Echo. >> “%REPORT_FILE%”

Echo ## Etat du Pare-feu Windows

Echo.

Netsh advfirewall show allprofiles >> “%REPORT_FILE%”

Echo. >> “%REPORT_FILE%”

Echo ## Connexions Reseau Actives (Netstat)

Echo.

Netstat -ano >> “%REPORT_FILE%”

Echo. >> “%REPORT_FILE%”

Echo ## Processes en Cours d’Execution

Echo.

Tasklist /svc /v >> “%REPORT_FILE%”

Echo. >> “%REPORT_FILE%”

Echo ## Taches Planifiees

Echo.

Schtasks /query /fo LIST /v >> “%REPORT_FILE%”

Echo. >> “%REPORT_FILE%”

Echo ## Utilisateurs et Groupes Locaux (Administrateurs)

Echo.

Net user >> “%REPORT_FILE%”

Net localgroup Administrators >> “%REPORT_FILE%”

Echo. >> “%REPORT_FILE%”

Echo ## Partages Reseau

Echo.

Net share >> “%REPORT_FILE%”

Echo. >> “%REPORT_FILE%”

Echo ## Points de Demarrage (Run Keys)

Echo.

Reg query HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run >> "%REPORT_FILE%"

Reg query HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run >> "%REPORT_FILE%"

Echo. >> "%REPORT_FILE%"

Echo ## Fichiers Systemes Proteges (SFC) – Resultat du dernier scan

Echo.

Findstr /C:"[SR]" %windir%\Logs\CBS\CBS.log | findstr /C:"Cannot repair member file" /C:"Repaired file" >> "%REPORT_FILE%"

Echo. >> "%REPORT_FILE%"

Echo Audit Termine. Le rapport a ete sauvegarde dans: %REPORT_FILE%

Echo.

Endlocal

Pause

Modèle 2 : Vérification des modifications de fichiers critiques

Ce script utilise un fichier de base de référence pour vérifier si des fichiers critiques ont été modifiés.

Étape 1 : Créer le fichier de base de référence (une seule fois sur un système propre)

@echo off

Set
"REF_FILE=C:\Security_Audits\file_hashes_reference.txt"

Echo Creation du fichier de reference pour les hashes…

Certutil -hashfile C:\Windows\System32\ntoskrnl.exe MD5 >> %REF_FILE%

Certutil -hashfile C:\Windows\System32\lsass.exe MD5 >> %REF_FILE%

Certutil -hashfile C:\Windows\System32\winlogon.exe MD5 >> %REF_FILE%

Certutil -hashfile C:\Windows\System32\cmd.exe MD5 >> %REF_FILE%

Certutil -hashfile C:\Windows\System32\net.exe MD5 >> %REF_FILE%

Certutil -hashfile C:\Windows\System32\tasklist.exe MD5 >> %REF_FILE%

Certutil -hashfile C:\Windows\System32\netstat.exe MD5 >> %REF_FILE%

Echo Fichier de reference cree: %REF_FILE%

Pause

Étape 2 : Script de vérification (à exécuter régulièrement)

@echo off

Setlocal

Set
“REF_FILE=C:\Security_Audits\file_hashes_reference.txt”

Set
“LOG_FILE=C:\Security_Audits\file_integrity_check_%date:~6,4%_%date:~3,2%_%date:~0,2%_%time:~0,2%%time:~3,2%%time:~6,2%.log”

If not exist “%REF_FILE%” (

Echo ERREUR: Le fichier de reference des hashes “%REF_FILE%” n’existe pas.

Echo Veuillez executer le script de creation de reference d'abord.

Pause

Exit /b 1

)

Echo Verification de l'integrite des fichiers critiques… >> "%LOG_FILE%"

Echo Date et Heure: %date% %time% >> "%LOG_FILE%"

Echo ---------- >> "%LOG_FILE%"

```
For /f “tokens=*” %%f in (‘type “%REF_FILE%”’) do (

   For /f “tokens=2,3” %%a in (‘echo %%f’) do (

      Set “FILENAME=%%a”

      Set “REFERENCE_HASH=%%b”



      Echo Verifiant “%FILENAME%”…

      For /f “tokens=2,3” %%c in (‘certutil -hashfile “!FILENAME!” MD5’) do (
```

```
        If "%%d"=="!REFERENCE_HASH!" (

            Echo [OK] "%FILENAME%" Hash: %%d >> "%LOG_FILE%"

        ) else (

            Echo [ALERTE] "%FILENAME%" MODIFIE! Actuel: %%d, Reference: !REFERENCE_HASH! >> "%LOG_FILE%"

        )

    )

  )

)
```

Echo Verification terminee. Rapport sauvegarde dans: %LOG_FILE%

Echo.

Endlocal

Pause

Note : Le script de vérification utilise des variables retardées (!VARIABLE!) qui nécessitent **setlocal enabledelayedexpansion** si utilisé directement. J'ai simplifié pour la clarté du livre, mais gardez cela à l'esprit pour le débogage.

Ces scripts montrent comment CMD, combiné à une bonne logique de programmation, peut être un outil puissant pour la cybersécurité proactive et la réponse aux incidents.

Annexes

Commandes CMD de sécurité par catégorie

- Informations Système : **systeminfo, wmic computersystem, wmic os**

- Processus et Services : **tasklist, taskkill, sc query, sc qc, wmic process, wmic service**

- Réseau : **ipconfig, netstat, ping, tracert, route print, arp, netsh advfirewall, netsh interface ip**

- Fichiers et Dossiers : **dir, copy, move, del, ren, attrib, icacls, cipher**

- Utilisateurs et Groupes : **net user, net localgroup, whoami, wmic useraccount**

- Registre : **reg query, reg add, reg delete**

- Tâches Planifiées : **schtasks**

- Journaux d'Événements : **wevtutil**

Ressources utiles

- Documentation Microsoft : La référence officielle pour toutes les commandes CMD. Recherchez “Microsoft Docs <command_name>”.

- SS64.com : Une excellente référence rapide pour les commandes CMD.

- Livres et cours sur PowerShell : Pour aller au-delà des limites de CMD.

- Forums de cybersécurité : Partagez vos scripts, posez des questions, apprenez des autres.

Glossaire des termes clés

- **ACL (Access Control List) :** Liste des permissions sur un fichier ou dossier.

- **Batch Script** : Fichier texte contenant une série de commandes CMD.

- **CMD (Command Prompt) :** Interface en ligne de commande de Windows.

- **EFS (Encrypting File System)** : Fonctionnalité de chiffrement de fichiers intégrée à Windows.

- **Hash (Somme de contrôle)** : Empreinte numérique unique d'un fichier, utilisée pour vérifier son intégrité.

- **Incident Response** : Processus de gestion et de confinement d'un incident de sécurité.

- **Malware** : Logiciel malveillant.

- **NetBIOS** : Ancien protocole réseau.

- **PID (Process ID)** : Identifiant unique d'un processus en cours d'exécution.

- **PowerShell** : Langage de script et shell de ligne de commande plus avancé que CMD.

- **Registre Windows** : Base de données hiérarchique des paramètres du système d'exploitation et des applications.

- **Rootkit** : Ensemble d'outils logiciels conçus pour cacher l'existence de certains processus ou programmes d'un système d'exploitation à d'autres méthodes d'inspection.

- **Threat Hunting** : Recherche proactive de menaces non détectées.

- **WMI (Windows Management Instrumentation)** : Technologie de gestion des composants Windows.

Conclusion

Vous avez maintenant parcouru un chemin qui vous a transformé, passant d'un utilisateur occasionnel de l'Invite de commandes à un expert capable d'exploiter ses capacités pour des objectifs de cybersécurité. Nous avons exploré les fondations, la navigation, l'automatisation, le diagnostic, la surveillance, et la réponse aux incidents, démontrant que CMD, malgré son âge, reste un outil pertinent et puissant dans l'arsenal de tout professionnel ou passionné de sécurité.

Les compétences que vous avez développées ne se limitent pas à la simple exécution de commandes ; elles résident dans la compréhension des mécanismes internes de Windows, la capacité à analyser les informations brutes, et l'art de l'automatisation pour une détection et une réponse rapides.

N'oubliez jamais que la cybersécurité est un domaine en constante évolution. Continuez à expérimenter dans des environnements de test, à lire les documentations

officielles, à suivre l'actualité des menaces, et à approfondir vos connaissances. Les scripts présentés dans ce livre sont des points de départ ; l'adaptation et l'amélioration continue seront vos meilleurs atouts.

En fin de compte, la maîtrise de la sécurité informatique ne consiste pas seulement à connaître les outils, mais à développer un état d'esprit vigilant et proactif. CMD est un excellent professeur pour cela, vous forçant à penser de manière structurée et logique.

Que ce livre soit le tremplin vers votre prochaine étape dans le monde fascinant de la cybersécurité. Le voyage ne fait que commencer !