

Épreuve intégrée – Audit d'un poste Windows via PowerShell (Adams)

Vous êtes technicien informatique et vous devez collecter des informations essentielles sur un poste Windows afin de les transmettre à un serveur de surveillance du parc.

 niveau

Ce document présente l'énoncé personnalisé de l'épreuve intégrée et précise les informations à collecter pour ce poste.

Contraintes

- Travail en PowerShell, utilisation du pipeline attendue
- Script dans un fichier `.ps1` lisible, structuré et commenté
- `Get-CimInstance` est autorisée et recommandée
- Chaque bloc doit être réalisé dans sa propre fonction
- Aucun accès direct au registre
- Export final d'un objet au format JSON à l'aide de `ConvertTo-Json`
- Respectez **EXACTEMENT** les valeurs qui vous sont demandées

Blocs à réaliser

Informations générales du poste

Objectif : Identifier de manière fiable la machine auditée dans un parc informatique.

A Faire : Récupérer les informations générales permettant d'identifier un poste Windows.

Commandes autorisées / recommandées : `$env:COMPUTERNAME`, `Get-CimInstance Win32_ComputerSystem`

Clé dans le json: `device`

Informations à récupérer

- **Nom de la machine**: Nom local du poste.
- **Constructeur du poste**: Fabricant (ex. Dell, HP, Lenovo).

- **Modèle du poste:** Référence du modèle matériel.
- **RAM totale:** Quantité totale de mémoire vive.
- **Utilisateur connecté:** Compte utilisateur actif au moment de l'exécution.
- **Présence d'un hyperviseur:** Indique si le système tourne dans un environnement virtualisé.
- **Type de système:** Architecture/Type du système (ex. x64).

Traitements à appliquer

- La quantité de RAM doit être affichée en MB.

Processeur (CPU)

Objectif : Décrire les capacités CPU du poste.

A Faire : Récupérer les informations essentielles du processeur.

Commandes autorisées / recommandées : `Get-CimInstance Win32_Processor`

Clé dans le json: `cpu`

Informations à récupérer

Pour tous les objets retournés:

- **Modèle du processeur:** Nom complet du processeur.
- **Nombre de cœurs:** Nombre de cœurs physiques.
- **Nombre de threads:** Nombre de processeurs logiques du CPU.
- **Fréquence maximale:** Fréquence max en MHz.
- **Désignation courte:** Nom court (si disponible).
- **Identifiant CPU:** Identifiant système du CPU.

Système d'exploitation

Objectif : Identifier précisément le Windows installé.

A Faire : Récupérer les informations essentielles du système d'exploitation.

Commandes autorisées / recommandées : `Get-CimInstance Win32_OperatingSystem`

Clé dans le json: `os`

Informations à récupérer

- **Nom du système:** Nom complet du système d'exploitation Windows installé.
- **Version:** Version principale du système d'exploitation.

- **Numéro de build:** Numéro de build précis de Windows.
- **Architecture OS:** Architecture du système (32-bit ou 64-bit).
- **Langues MUI:** Liste des langues d'interface utilisateur installées sur le système.
- **Date d'installation:** Date à laquelle Windows a été installé.

Cartes graphiques (GPU)

Objectif : Inventorier les GPU et leurs pilotes.

A Faire : Lister les cartes graphiques détectées sur la machine.

Commandes autorisées / recommandées : `Get-CimInstance Win32_VideoController`

Clé dans le json: `gpus`

Informations à récupérer

Pour tous les objets retournés:

- **Nom de la carte graphique:** Nom complet de la carte graphique ou du contrôleur vidéo.
- **Fabricant:** Fabricant ou fournisseur du contrôleur vidéo.
- **Version du pilote:** Version du pilote graphique installé.
- **Mémoire vidéo:** Quantité de mémoire vidéo dédiée (en octets).
- **Processeur graphique:** Nom du GPU ou du processeur graphique utilisé.
- **Date du pilote:** Date d'installation ou de publication du pilote graphique.
- **Mode vidéo:** Description complète du mode vidéo actif (résolution + couleurs).

Traitements à appliquer

- La quantité de RAM doit être affichée en GB.

Adaptateurs réseau

Objectif : Inventorier l'état des interfaces réseau.

A Faire : Lister les adaptateurs réseau et leur état.

Commandes autorisées / recommandées : `Get-NetAdapter`

Clé dans le json: `networkAdapters`

Informations à récupérer

Pour tous les objets retournés:

- **Nom de l'interface:** Nom logique de l'interface réseau.

- **Description:** Description matérielle complète de la carte réseau.
- **Statut:** État actuel de l'interface réseau (Up, Down, Disabled).
- **Adresse MAC:** Adresse MAC physique de l'interface réseau.
- **Vitesse du lien:** Vitesse de connexion négociée de l'interface réseau.
- **Type de média:** Type de média réseau utilisé (Ethernet, Wi-Fi, etc.).

Traitements à appliquer

- Ne conserver que les interfaces dont l'état est « Up ».
- Ne conserver que les interfaces matérielles.
- Trier les interfaces par vitesse de connexion décroissante.

Résultat attendu

Votre script doit produire un **objet** structuré contenant les informations demandées dans les blocs ci-dessus et l'exporter au format JSON.

Votre script **DOIT fonctionner correctement** et fournir les informations demandées.

Présentation orale

Vous devrez être capable d'expliquer :

- comment vous avez identifié les bonnes informations
- pourquoi vous avez choisi certaines commandes
- comment vous avez appliqué les tris / filtres / seuils
- comment vous avez structuré les données
- Vous devrez être en mesure de modifier le script pour ajouter des informations supplémentaires ou corriger des erreurs.