

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

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.
- **Type de système:** Architecture/Type du système (ex. x64).
- **Utilisateur connecté:** Compte utilisateur actif au moment de l'exécution.
- **Processeurs logiques:** Nombre de threads visibles par l'OS.

Traitements à appliquer

- La quantité de RAM doit être affichée en GB.

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.
- **Désignation courte:** Nom court (si disponible).
- **Identifiant CPU:** Identifiant système du CPU.
- **Socket:** Désignation du socket (si disponible).

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.
- **Date d'installation:** Date à laquelle Windows a été installé.
- **Éditeur:** Éditeur du système d'exploitation (ex. Microsoft Corporation).
- **Utilisateur enregistré:** Nom de l'utilisateur enregistré lors de l'installation.

Logiciels installés

Objectif : Réaliser un inventaire applicatif du poste.

A Faire : Lister les logiciels installés et des informations de base.

Commandes autorisées / recommandées : `Get-Package` , `Select-Object` , `Sort-Object`

Clé dans le json: `packages`

Informations à récupérer

Pour tous les objets retournés:

- **Nom du logiciel:** Nom du package/logiciel.
- **Version:** Version installée.
- **Type d'installation:** Origine/gestionnaire (ex. msi, Programs).
- **Référence interne:** Identifiant interne (si disponible).
- **Source:** Source du package (si disponible).
- **Statut:** Statut du package (si disponible).

Traitements à appliquer

- Trier la liste des logiciels par ordre alphabétique
- Ne conserver que les 30 premiers logiciels après tri.

Sécurité – Pare-feu

Objectif : Vérifier l'état du pare-feu Windows.

A Faire : Lister les profils de pare-feu et leur activation.

Commandes autorisées / recommandées : `Get-NetFirewallProfile`

Clé dans le json: `firewallProfiles`

Informations à récupérer

Pour tous les objets retournés:

- **Nom du profil:** Profil (Domain/Private/Public).

- **Activé:** Indique si le pare-feu est activé.
- **Action entrante par défaut:** Action par défaut (Allow/Block).
- **Notification:** Notification sur écoute (si disponible).
- **Fichier de log:** Chemin de log (si disponible).
- **Action sortante par défaut:** Action sortante (Allow/Block).

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.