Surveillance et inventaire d'un parc informatique

La gestion d'un parc informatique moderne ne se limite pas à installer des ordinateurs. Elle implique un **suivi continu**, une **vision globale de l'état des machines**, et la capacité de **détecter rapidement les anomalies**. Cette surveillance repose sur des outils techniques, des scripts d'inventaire et une architecture adaptée aux besoins de l'organisation.

Découverte

Pourquoi surveiller et inventorier un parc informatique ?

Dans un contexte professionnel (école, entreprise, administration), un parc informatique peut contenir **des dizaines ou des centaines de postes**. Sans outil adapté, il devient difficile de :

- savoir **quel matériel est installé**
- vérifier que les **postes sont conformes** (OS, versions, sécurité)
- anticiper les **pannes ou saturations**
- détecter des **comportements anormaux** (processus gourmands, espace disque insuffisant)
- répondre rapidement à un **incident utilisateur**

👉 La surveillance et l'inventaire sont donc des **missions centrales du technicien informatique**.

Comment surveiller et inventorier un parc informatique ?

Dans le monde professionnel, la surveillance et l'inventaire d'un parc informatique reposent généralement sur des outils spécialisés (solutions de monitoring, plateformes d'inventaire, agents dédiés). Ces outils sont souvent complexes, distribués, et intégrés à des infrastructures complètes.

👉 Les outils utilisés dans ce cours ne sont pas ceux que l'on retrouve tels quels en entreprise, mais la philosophie et l'architecture sont strictement les mêmes.

L'objectif **pédagogique** est de comprendre comment fonctionne un système de surveillance, pas d'apprendre un outil propriétaire particulier.

Une philosophie commune à tous les outils professionnels

Quel que soit l'outil (open source ou commercial), on retrouve toujours les mêmes principes fondamentaux :

- un **agent côté poste client**, chargé de collecter les informations
- un **serveur central**, qui reçoit, stocke et analyse les données
- une **interface de supervision (dashboard)**, qui permet de visualiser l'état du parc
- un système d'**alertes**, déclenchées en cas de problème

Ce modèle est universel, que l'on utilise un simple script PowerShell ou une plateforme de monitoring industrielle.

Monitoring passif et monitoring actif

Monitoring passif

Le monitoring passif consiste à :

- collecter régulièrement des informations
- les stocker pour analyse
- consulter l'état du parc via des tableaux de bord

Il permet :

- une vision globale
- un suivi dans le temps
- l'identification de tendances (usure disque, montée en charge, etc.)

Monitoring actif

Le monitoring actif va plus loin :

- il définit des **seuils**
- déclenche des **alertes automatiques**
- permet une réaction rapide avant l'incident utilisateur

Exemples :

- alerte si l'espace disque devient critique
- alerte si un service important est arrêté
- alerte si un poste ne répond plus au réseau

Transposition pédagogique

Dans ce cours :

- le **script PowerShell joue le rôle de l'agent**
- l'**API centrale joue le rôle du serveur de collecte**
- le **dashboard web représente la console de supervision**
- les **seuils et filtres** simulent les règles d'alerte

Ainsi, même si les outils sont simplifiés, tu manipules les **mêmes concepts, les mêmes architectures et le même raisonnement** qu'un technicien informatique en environnement professionnel.

Inventaire vs surveillance : deux notions complémentaires

Inventaire informatique

L'inventaire consiste à **collecter des informations statiques ou semi-statiques** sur les postes :

- matériel (CPU, RAM, disques, cartes réseau…)
- système d'exploitation
- logiciels installés
- configuration réseau
- firmware (BIOS, carte mère)

Ces données servent à :

- documenter le parc
- assurer la conformité
- préparer des renouvellements de matériel
- auditer un environnement

Surveillance (monitoring)

La surveillance concerne des **données dynamiques**, observées régulièrement :

- espace disque disponible
- état des services
- charge CPU / mémoire

- état réseau
- disponibilité des postes

Objectif : **détecter un problème avant qu'il n'impacte l'utilisateur**.

Architecture logique d'un système de surveillance

Un système de surveillance de parc repose généralement sur **3 composants principaux**.

Les postes clients (agents)

Chaque poste surveillé doit être capable de :

- collecter ses propres informations
- les transmettre à un serveur central

Deux approches existent :

- **agent installé** (service permanent)
- **script exécuté périodiquement** (ex. PowerShell)

Dans le cadre pédagogique, l'approche scriptée est idéale :

- simple
- contrôlable
- transparente

Le serveur central

Le serveur central a pour rôle de :

- recevoir les données des postes
- les stocker (base de données)
- les analyser
- les afficher (dashboard)

Il peut être :

- un serveur Linux ou Windows
- hébergé localement ou dans le cloud
- basé sur des technologies web standards

L’interface d’administration

Elle permet au technicien de :

- visualiser l’état du parc
- filtrer par poste, service ou problème
- consulter l’historique
- déclencher des alertes

Souvent sous forme de :

- dashboard web
- graphiques
- tableaux d’état

Rôle de PowerShell dans un parc Windows

Dans un environnement Windows, **PowerShell est l’outil de référence** pour l’inventaire et la surveillance.

Pourquoi PowerShell ?

- accès natif aux informations système
- commandes orientées **objets**
- automatisation simple
- scripts réutilisables
- compatible avec une architecture client/serveur

PowerShell permet par exemple de :

- interroger le matériel via **CIM/WMI**
- analyser le système de fichiers
- récupérer la configuration réseau
- produire des exports structurés (JSON, CSV)

Principe d’un script d’inventaire

Un script d’inventaire bien conçu suit une logique claire.

Collecte des données

Le script interroge le système via des cmdlets :

- matériel
- OS
- réseau
- stockage
- services
- logiciels

Chaque information est stockée dans des **objets structurés**.

Traitement et filtrage

Avant l'envoi, certaines données peuvent être :

- filtrées (seuils)
- triées (Top N)
- limitées (résumé exploitable)

Cela permet de :

- réduire le volume de données
- se concentrer sur l'essentiel
- faciliter l'analyse côté serveur

Export et transmission

Les données sont ensuite :

- converties en **JSON** (format standard)
- envoyées vers une **API REST**

Ce découplage permet :

- de changer le serveur sans modifier le script
- d'analyser les données avec différents outils
- d'archiver l'historique

Outils open source couramment utilisés

Collecte / agents

- PowerShell (Windows)
- Bash / Python (Linux)
- Scripts planifiés (Task Scheduler / cron)

Serveur et API

- Node.js + Express
- PHP (API REST)
- Python (Flask / FastAPI)

Stockage

- PostgreSQL / MySQL
- MongoDB (données JSON)
- InfluxDB (données temporelles)

Visualisation / monitoring

- Grafana
- Zabbix
- Prometheus
- Netdata
- GLPI (inventaire + parc)

Surveillance passive vs active

Surveillance passive

- collecte périodique
- analyse a posteriori
- peu intrusive

Exemple : inventaire toutes les 24h.

Surveillance active

- alertes en temps réel
- seuils critiques
- actions automatiques

Exemple : alerte si espace disque < 10 %.

Bonnes pratiques professionnelles

- ne collecter que les **données utiles**
- documenter les scripts
- structurer les données
- centraliser les logs
- historiser les informations
- séparer collecte et affichage
- sécuriser les échanges (HTTPS)

Lien avec l’épreuve intégrée

Dans le cadre de l’épreuve :

- vous jouez le rôle d’un **technicien informatique**
- vous concevez un **outil d’inventaire réel**
- vous justifiez vos choix techniques
- vous démontrez votre compréhension du **fonctionnement d’un parc**

👉 Ce type de projet reflète une **situation professionnelle authentique**, bien plus qu’un simple exercice de commande.

© 2025 Mr. Lambrechts https://v2.ttrinfo.be 8 | 9