

TP Thème 1 – Manipulation de fichiers et bibliothèque *random*

Génération de logs synthétiques avec Python (*synthetic_logs.py*)

Objectifs pédagogiques

- Découvrir l'utilisation de la bibliothèque **random** pour générer des données aléatoires.
- Comprendre la construction d'un **script Python modulaire** (constantes, fonctions, boucle principale).
- Produire un **fichier de logs** utilisable pour des exercices de supervision et de monitoring.
- Expliquer en commentaire le rôle de chaque fonction et instruction clé.

Travail demandé

Vous allez compléter un script Python (*synthetic_logs.py*) fourni avec des parties manquantes (indiquées par ??? et par # TODO).

Étapes à suivre

1. Lire le script fourni

- Identifier les constantes de configuration (N_EVENTS, HOSTS, LEVELS...).
Expliquer quelle est la signification de chacune de ces constantes.
- Observer la structure du programme (imports, fonctions utilitaires, boucle de génération, écriture du fichier).
- Expliquer quelles sont les fonctionnalités des bibliothèques importées.

2. Compléter les parties manquantes

- Implémenter les fonctions utilitaires :
 - `random_ip()` : retourne une adresse IPv4 aléatoire.
 - `random_user()` : choisit un utilisateur dans une liste prédéfinie.
 - `random_dev()` : choisit un périphérique disque aléatoire.
- Compléter la boucle de génération :
 - Générer un **horodatage** aléatoire dans une plage donnée.
 - Tirer un hôte, un processus et un niveau de gravité avec `random.choice` ou `random.choices`.

- Construire un **message** en remplissant un modèle (template.format(...)). Quelle est la fonctionnalité de la méthode format() ?

3. Commenter le script

- Ajouter des commentaires expliquant :
 - le rôle de chaque fonction utilitaire,
 - la différence entre random.choice et random.choices,
 - pourquoi on trie les événements avant d'écrire dans le fichier,
 - le format choisi pour les logs.

4. Exécuter le script

- Lancer le script pour générer le fichier synthetic_system.log.
- Vérifier que le fichier contient plusieurs centaines de lignes de logs réalistes.

5. Analyse rapide (bonus)

- Ouvrir le fichier généré et compter :
 - combien de fois apparaît le niveau ERROR,
 - quel hôte est le plus présent,
 - si les horodatages sont bien triés.