



# CYBER SÉCURITÉ

Détection d'Anomalies et  
Prévention des Cyberattaques à  
Travers l'Analyse des Logs de  
Sécurité"

Réalisé par : WAFAA EL MAIFI

# Tableau de Contenu

Introduction	3
Objectifs du Projet	4
Chargement des données	5
Analyse descriptive des événements	6
Identification des anomalies dans les données	9
Analyse et Visualisation des anomalies	13
Conclusions et Recommandations	14

# Introduction

Ce projet vise à analyser des logs de sécurité pour détecter des comportements atypiques ou des anomalies potentielles. Les logs contiennent des informations telles que les timestamps, les types d'activité, les utilisateurs, les adresses IP, et les actions effectuées. L'objectif est d'identifier des activités suspectes, des erreurs système, ou des utilisateurs inconnus en utilisant des méthodes statistiques et des techniques de visualisation.

# Objectifs du Projet



1. Chargement et exploration des données : Comprendre la structure des logs et les types d'événements enregistrés.
2. Analyse descriptive : Calculer des statistiques pour identifier les comportements fréquents et les schémas d'activité.
3. Détection des anomalies : Appliquer des tests statistiques et des méthodes d'analyse pour identifier des événements suspects.
4. Analyse temporelle : Examiner les logs par période pour détecter des activités inhabituelles à des moments spécifiques.
5. Visualisation des résultats : Représenter graphiquement les anomalies et les tendances pour une meilleure interprétation.

# Chargement des données

## 1. Chargement des données

### a. Chargement des logs de sécurité

- Méthode** : Les logs ont été chargés à l'aide de la bibliothèque `pandas` en utilisant la fonction `read_csv`.
- Résultat** : Le fichier CSV a été chargé dans un DataFrame, ce qui permet de manipuler et d'analyser les données facilement.

### b. Exploration des données

- Premières lignes** : Les 5 premières lignes du DataFrame ont été affichées pour comprendre la structure des données.
- Types de données** : Les types de données de chaque colonne ont été vérifiés (par exemple, `Timestamp` est un objet, `User_ID` est un entier).
- Résumé des données** : Un résumé des données a été généré pour identifier les valeurs manquantes et les colonnes pertinentes.

```

✓ 0s ⏪ #1.a Chargez les logs de sécurité avec pandas.:
df = pd.read_csv('/content/cybercrime_forensic_dataset.csv')

#1.b Examinez les premières lignes et les types de données
print("Premières lignes du dataset :")
print(df.head())

└ Premières lignes du dataset :
    Timestamp User_ID      IP_Address Activity_Type \
0  2024-09-27 12:53:26.390859     9288   10.174.236.5 File_Modification
1  2024-10-02 15:13:49.741543    1605   172.19.128.216      USB_Insert
2  2024-09-13 14:31:39.163288    3056  192.168.201.132 File_Modification
3  2024-10-02 22:27:57.622008    1564  10.119.122.121 Network_Traffic
4  2024-10-01 08:00:55.017797    3660   172.23.2.38      USB_Insert

    Resource_Accessed      File_Name Action Login_Attempts \
0  /network/logs/new_project.docx new_project.docx Delete      NaN
1          /server/secrets.txt           NaN  Read      NaN
2        /project/document2.docx  document2.docx      NaN      NaN
3        /backup/document2.docx           NaN Delete      NaN
4  /confidential/report1.pdf           NaN  Write      NaN

    File_Size Anomaly_Type      Label
0     30.66 DDoS_Attempt Suspicious
1       NaN  Brute_Force Suspicious
2     21.61 DDoS_Attempt Suspicious
3       NaN        NaN      Normal
4       NaN        NaN      Normal

[9] # + : Type de données
print("\nTypes de données :")
print(df.dtypes)

# + : Résumé des données
print("\nRésumé des données :")
print(df.info())

└ Types de données :
Timestamp          object
User_ID            int64
IP_Address         object
Activity_Type      object
Resource_Accessed object
File_Name          object
Action             object
Login_Attempts    float64
File_Size          float64

```

# Analyse descriptive des événements

## 2. Analyse descriptive des événements

### a. Statistiques descriptives

- **Méthode** : Des statistiques descriptives (moyenne, écart-type, minimum, maximum) ont été calculées pour les colonnes numériques.
- **Résultat** : Ces statistiques donnent une vue d'ensemble des données, comme le nombre moyen de tentatives de connexion par utilisateur.

### b. Événements les plus fréquents

- **Méthode** : La fréquence des types d'activité a été calculée à l'aide de `value_counts()`.
- **Résultat** : Les activités les plus fréquentes ont été identifiées (par exemple, les accès aux fichiers, les tentatives de connexion).

```

0s  ⏎ # 2.a statistiques descriptives :
      print("\nStatistiques descriptives :")
      print(df.describe())

平淡无奇的输出结果，展示了User_ID、Login_Attempts和File_Size三列的基本统计信息，包括计数、平均值、标准差等。
平淡无奇的输出结果，展示了User_ID列中各用户ID对应的事件计数，显示了事件分布的广泛性。

```

# Analyse descriptive des événements

▶ # On filtre les événements de type "Login" et on compte les tentatives de connexion (Login\_Attempts).  
 login\_attempts = df[df['Activity\_Type'] == 'Login']['Login\_Attempts'].value\_counts()  
 print("\nFréquence des tentatives de connexion :")  
 print(login\_attempts)

→ Fréquence des tentatives de connexion :  
 Login\_Attempts  
 4.0 130  
 10.0 111  
 5.0 109  
 1.0 103  
 6.0 102  
 3.0 101  
 7.0 101  
 8.0 100  
 9.0 99  
 2.0 97  
 Name: count, dtype: int64

- On sélectionne uniquement les lignes où la colonne Activity\_Type a la valeur "Login".
- Cela signifie qu'on garde uniquement les événements liés aux connexions.
- On extrait la colonne Login\_Attempts, qui représente le nombre de tentatives de connexion pour chaque événement.
- value\_counts() compte combien de fois chaque valeur de Login\_Attempts apparaît.

Le tableau de sortie affiche les différentes valeurs de Login\_Attempts ainsi que leur fréquence :

Login_Attempts	Fréquence
4.0	130
10.0	111
5.0	109
1.0	103
6.0	102
3.0	101
7.0	101
8.0	100
9.0	99
2.0	97

- Il y a eu 130 événements où l'utilisateur a tenté 4 fois de se connecter avant de réussir ou d'abandonner.
- 111 événements avec 10 tentatives.
- 103 événements avec 1 seule tentative.

# Analyse descriptive des événements

2.c Qu'est-ce que la fréquence des événements peut nous dire sur le comportement normal des utilisateurs ou des systèmes ?

L'analyse de la **fréquence des événements** permet de comprendre le comportement normal des utilisateurs et des systèmes tout en détectant des anomalies et des menaces potentielles.

## 1. Identification des comportements normaux

- Permet de repérer les **activités régulières** et les **schémas d'utilisation** (pics d'activité en journée, baisse le week-end).

## 2. Détection des anomalies

- Une **fréquence anormale** (hausse ou baisse soudaine) peut indiquer une activité suspecte :
  - **Pic de tentatives de connexion échouées** → Possible attaque par force brute.
  - **Diminution d'accès à des fichiers critiques** → Tentative de contournement des contrôles.

## 3. Identification des menaces

- Une **fréquence élevée de connexions répétées** ou **d'accès à des fichiers sensibles** peut signaler une attaque.
- Une **hausse des erreurs système** peut révéler une exploitation de vulnérabilités.

## 4. Analyse du comportement des utilisateurs

- Un utilisateur devenant soudainement **très actif** après une période d'inactivité peut être suspect.
- L'accès à des **fichiers inhabituels** peut indiquer une activité malveillante.

## 5. Optimisation et prévention

- Aide à gérer la **charge du système** et planifier la **maintenance** durant les périodes de faible activité.

## 6. Corrélation avec d'autres indicateurs

- Associer la fréquence des événements à des **facteurs temporels** et à d'autres logs pour identifier des tendances suspectes.
- Exemple : une hausse des **tentatives de connexion à 3h du matin** peut signaler une attaque automatisée.

# Identification des anomalies dans les données

✓ 0s ➔ #3.a Appliquez un test statistique pour détecter des anomalies

```
from scipy.stats import chi2_contingency

# Création de la table de contingence
contingency_table = pd.crosstab(df['Activity_Type'], df['Label'])

# Application du test du chi-deux
chi2, p, dof, expected = chi2_contingency(contingency_table)

# Calcul des résidus standardisés (pour identifier les anomalies)
residuals = (contingency_table - expected) / np.sqrt(expected)

# Affichage des résultats
print(f"\nTest de chi carré : Chi2 = {chi2}, p-value = {p}")
print("\nTableau des résidus standardisés (anomalies potentielles) :")
print(residuals)

# Mise en évidence des valeurs suspectes
threshold = 2 # Seuil d'anomalie (supérieur à 2 ou inférieur à -2)
anomalies = residuals[(residuals > threshold) | (residuals < -threshold)]
print("\nValeurs suspectes dépassant le seuil :")
print(anomalies.dropna(how="all").dropna(axis=1, how="all"))
```

➡ Test de chi carré : Chi2 = 9.287813588317704, p-value = 0.15802688645961668

Tableau des résidus standardisés (anomalies potentielles) :		
Label	Normal	Suspicious
Activity_Type		
File_Access	0.197262	-0.441163
File_Deletion	0.571022	-1.277051
File_Modification	-0.409619	0.916085
Login	0.015277	-0.034167
Network_Traffic	0.233022	-0.521137
Remote_Login	0.313615	-0.701379
USB_Insert	-0.928384	2.076267

1. On calcule la table de contingence pour voir la répartition des `Activity_Type` selon `Label`.
2. On applique le test du chi-deux pour voir s'il y a une relation significative.
3. On calcule les résidus standardisés :
  - Un résidu élevé ( $> 2$ ) signifie qu'une activité est plus fréquente que prévu dans une catégorie.
  - Un résidu négatif très bas ( $< -2$ ) signifie qu'elle est moins fréquente que prévu.
4. On filtre les valeurs suspectes où les écarts sont significatifs.

# Identification des anomalies dans les données

## Interprétation des résultats :

### 1. Test du chi-deux :

- Chi2 = 9.29 et p-value = 0.158 → Il n'y a pas de relation statistiquement significative entre Activity\_Type et Label au seuil de 5% (p > 0.05).
- Cela signifie que, dans l'ensemble, les activités ne montrent pas d'anomalie flagrante.

### 2. Analyse des résidus standardisés :

- La plupart des valeurs sont proches de 0, ce qui indique que les écarts entre les valeurs observées et attendues ne sont pas significatifs.
- Seule l'activité USB\_Insert dans la catégorie "Suspicious" a un résidu standardisé élevé (2.076267), ce qui suggère une anomalie potentielle.

### 3. Anomalie détectée :

- USB\_Insert apparaît plus souvent que prévu dans la catégorie "Suspicious".
  - Cela peut indiquer une activité suspecte liée à l'insertion de périphériques USB, qui pourrait être un vecteur d'attaque (ex. malware, vol de données).
- 
- b Comment les anomalies dans les logs de sécurité peuvent-elles être interprétées ? Peut-on distinguer des événements légitimes des attaques potentielles uniquement avec des tests statistiques ?

Les tests statistiques détectent des anomalies mais ne suffisent pas à distinguer un événement légitime d'une attaque. Un pic de connexions échouées peut être une erreur humaine ou une attaque par force brute. Pour une détection efficace, il faut corrélérer plusieurs indicateurs, analyser le contexte et utiliser des modèles avancés

# Identification des anomalies dans les données

**3.b Comment les anomalies dans les logs de sécurité peuvent-elles être interprétées ? Peut-on distinguer des événements légitimes des attaques potentielles uniquement avec des tests statistiques ?**

## Interprétation des anomalies dans les logs de sécurité

Les anomalies dans les logs sont des événements qui dévient du comportement habituel. Elles peuvent être **bénignes** (ex. un employé qui se connecte depuis un nouvel appareil) ou **malveillantes** (ex. une tentative de piratage).

---

## Les tests statistiques suffisent-ils pour détecter les attaques ?

**✗ Non, seuls les tests statistiques ne peuvent pas confirmer une attaque.**

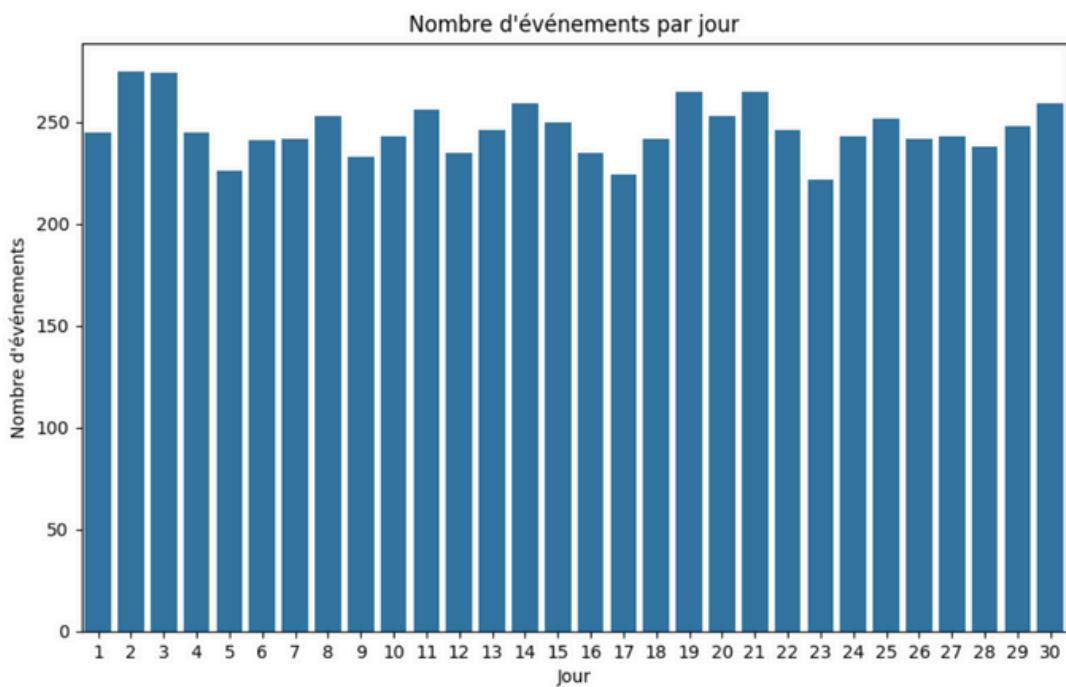
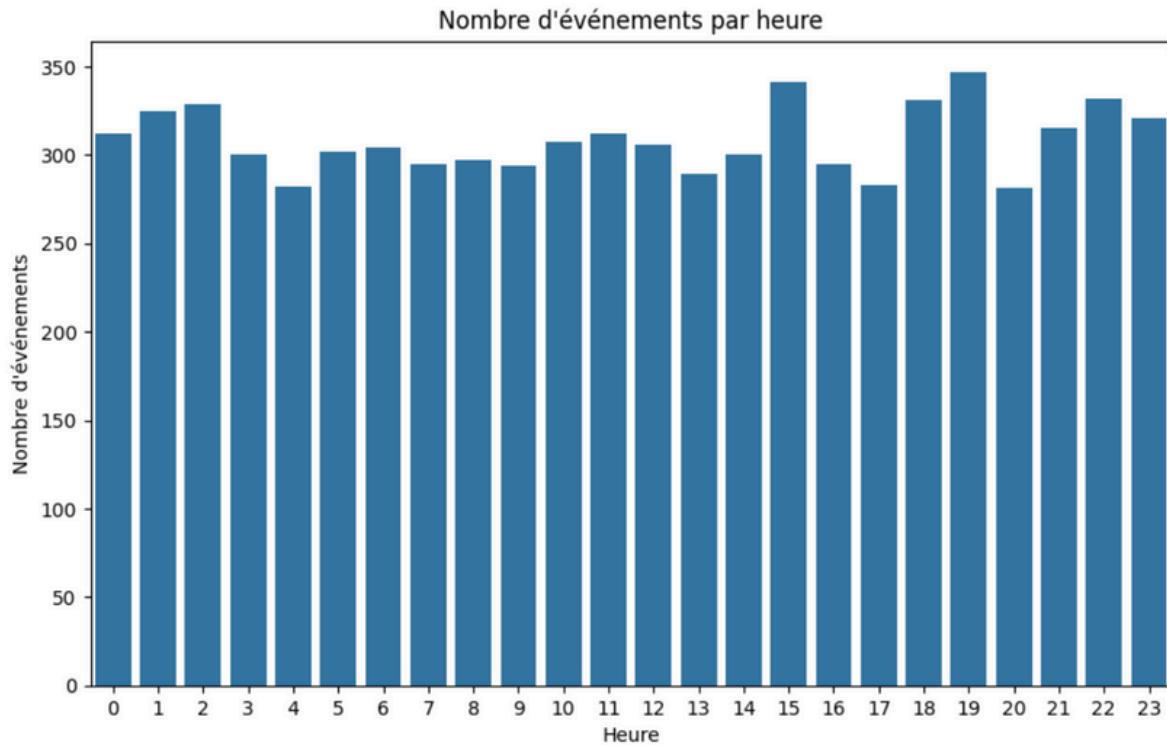
Ils détectent des écarts, mais ne donnent **ni le contexte, ni la cause**.

Exemple :

- ◆ Un pic de connexions échouées peut être **un utilisateur ayant oublié son mot de passe ou une attaque par force brute**.
- ◆ Un accès inhabituel à un fichier peut être **un employé en mission spéciale ou une tentative d'exfiltration de données**.

**Les tests statistiques détectent des anomalies mais ne suffisent pas à distinguer un événement légitime d'une attaque. Un pic de connexions échouées peut être une erreur humaine ou une attaque par force brute. Pour une détection efficace, il faut corrélérer plusieurs indicateurs, analyser le contexte et utiliser des modèles avancés**

# Analyse et Visualisation des anomalies



# Analyse et Visualisation des anomalies

## 1. Périodes sensibles aux attaques ou anomalies

Les périodes où le nombre d'événements est plus élevé peuvent indiquer une activité intense, ce qui peut être normal (horaires de travail) ou anormal (tentatives d'attaques).

Périodes à risque potentiel :

15h - 22h : On observe des pics d'événements, notamment à 15h (341), 18h (331), 19h (347), et 22h (332).

Cela peut être dû à une activité normale accrue en fin de journée.

Mais ces horaires correspondent aussi à des moments où des attaquants pourraient essayer de se cacher dans un trafic élevé.

Heures creuses (0h - 5h) :

L'activité est plus faible mais non nulle (entre 282 et 329 événements).

Un pic anormal la nuit pourrait indiquer des activités suspectes, comme des scans de réseau, tentatives de connexion non autorisées, ou exfiltration de données.

## 2. Corrélation entre événements et horaires

- Les pics d'événements coïncident avec les heures de bureau (8h - 18h)
- Cela est logique, car les employés accèdent aux fichiers, se connectent, et génèrent du trafic réseau.
- Cependant, un pic anormalement élevé à ces heures peut aussi être un signe d'attaque (phishing, mouvements latéraux dans le réseau).
- Un pic en soirée (19h - 22h) peut être suspect
- Normalement, l'activité devrait baisser après la journée de travail.
- Si elle reste élevée, cela peut indiquer des connexions suspectes à distance ou des activités automatisées.
- Activité nocturne (0h - 5h) : un signal d'alerte potentiel
- Un attaquant peut tenter d'exécuter des actions malveillantes pendant ces heures, pensant que personne ne surveille.
- Une hausse inhabituelle d'événements à ces horaires peut être un indicateur d'intrusion.
-

## Conclusion & Recommandations

1. Surveiller particulièrement les périodes de 15h-22h et 0h-5h.
2. Mettre en place des alertes en cas d'activité anormale la nuit.
3. Analyser la nature des événements aux heures de forte activité pour distinguer les comportements légitimes des menaces.
4. Corréler avec d'autres indicateurs de sécurité (adresses IP suspectes, échecs de connexion répétés, etc.).

En résumé, l'analyse des événements par heure permet d'identifier des périodes critiques où des anomalies ou des cyberattaques sont plus probables. Une surveillance continue et l'analyse des tendances sont essentielles pour améliorer la sécurité du système.