# Monitoring Labo 1: Rapport

## Localisation logs

Fichiers intéressants

/var/log/auth.log

/var/log/syslog.1

/var/log/debug.log

/var/log/kern.log

## La taxonomie

/var/log/auth.log → authentification et autorisation, accès au système et aux données

/var/log/syslog.1 → gestion des capacités

/var/log/debug → gestion des capacités

/var/log/kern.log → gestion des capacités, blocage de scans (log fw ufw)

## Échelle des priorités

/var/log/auth.log → high

/var/log/syslog.1 → low

/var/log/debug → low

/var/log/kern.log → low, medium

## Relation avec la sécurité du système d’exploitation

Le fichier *auth.log* est un fichier important à surveiller car il répertorie tous les accès ou tentatives, qui accède à quelles données, à quel moment, …

Les autres sont intéressants car ils permettent de connaître l’état du système à un moment donné.

# 

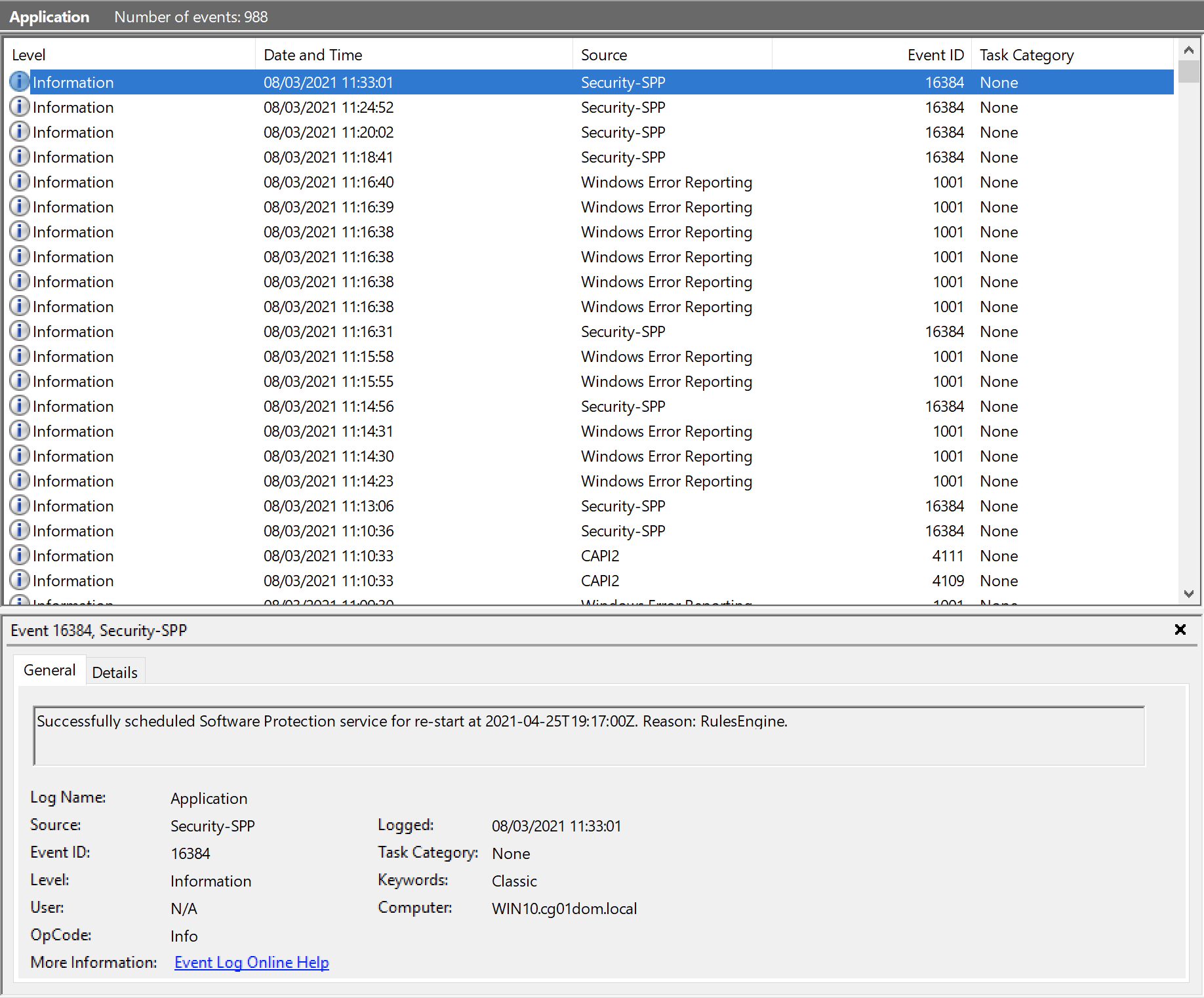
# Monitoring Labo 2: Rapport

## Localisation logs

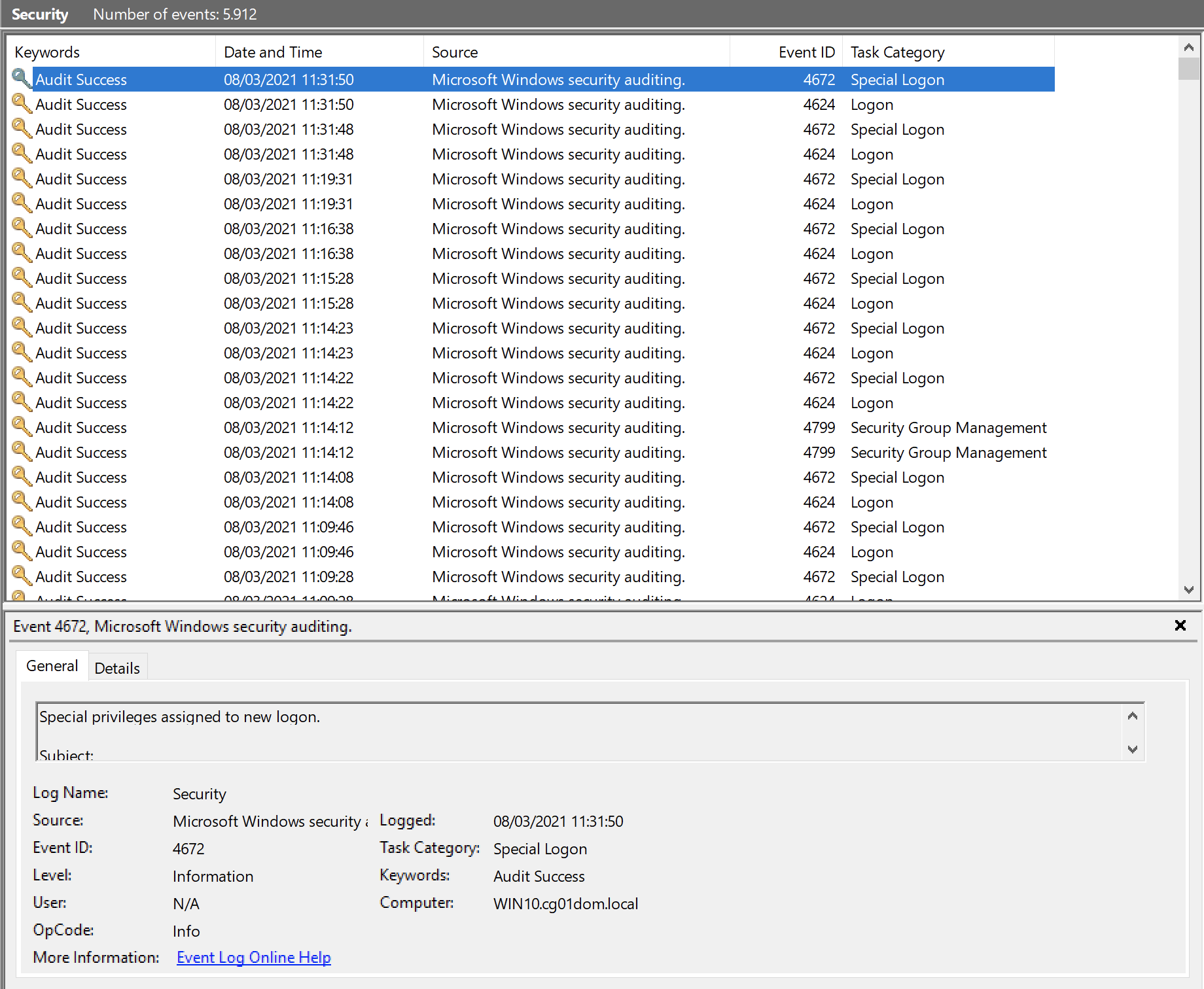
* Event viewer
  + Application
  + Security
  + Setup
  + System
* %SystemRoot%\System32\Winevt\Logs\ → logs généraux
* %SystemRoot%\System32\LogFiles\Firewall → logs du firewall

## La taxonomie

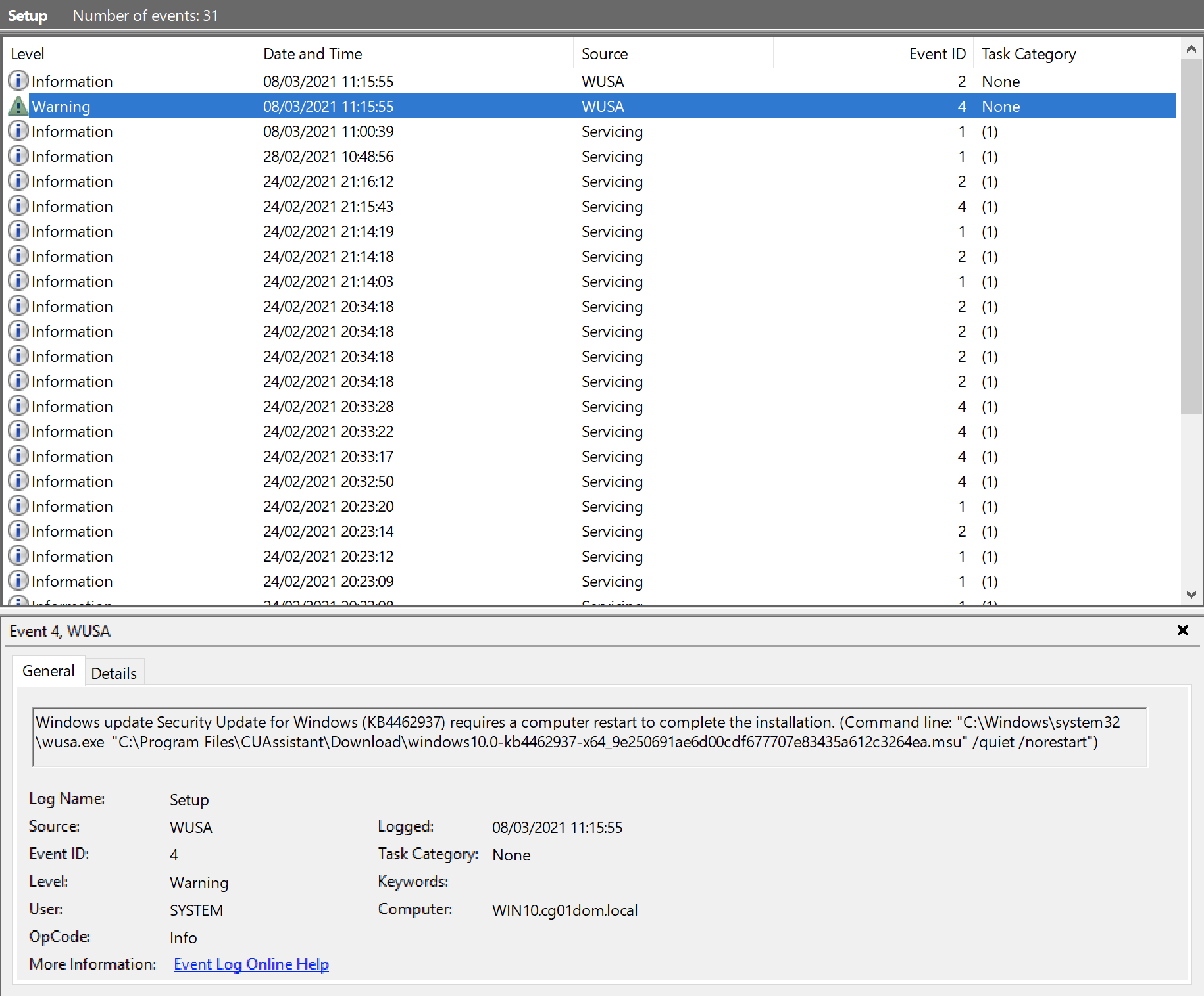
* 1. Application : Ce sont des logs qui sont produits lorsqu’il y a une vérification des clés produit de l’application.



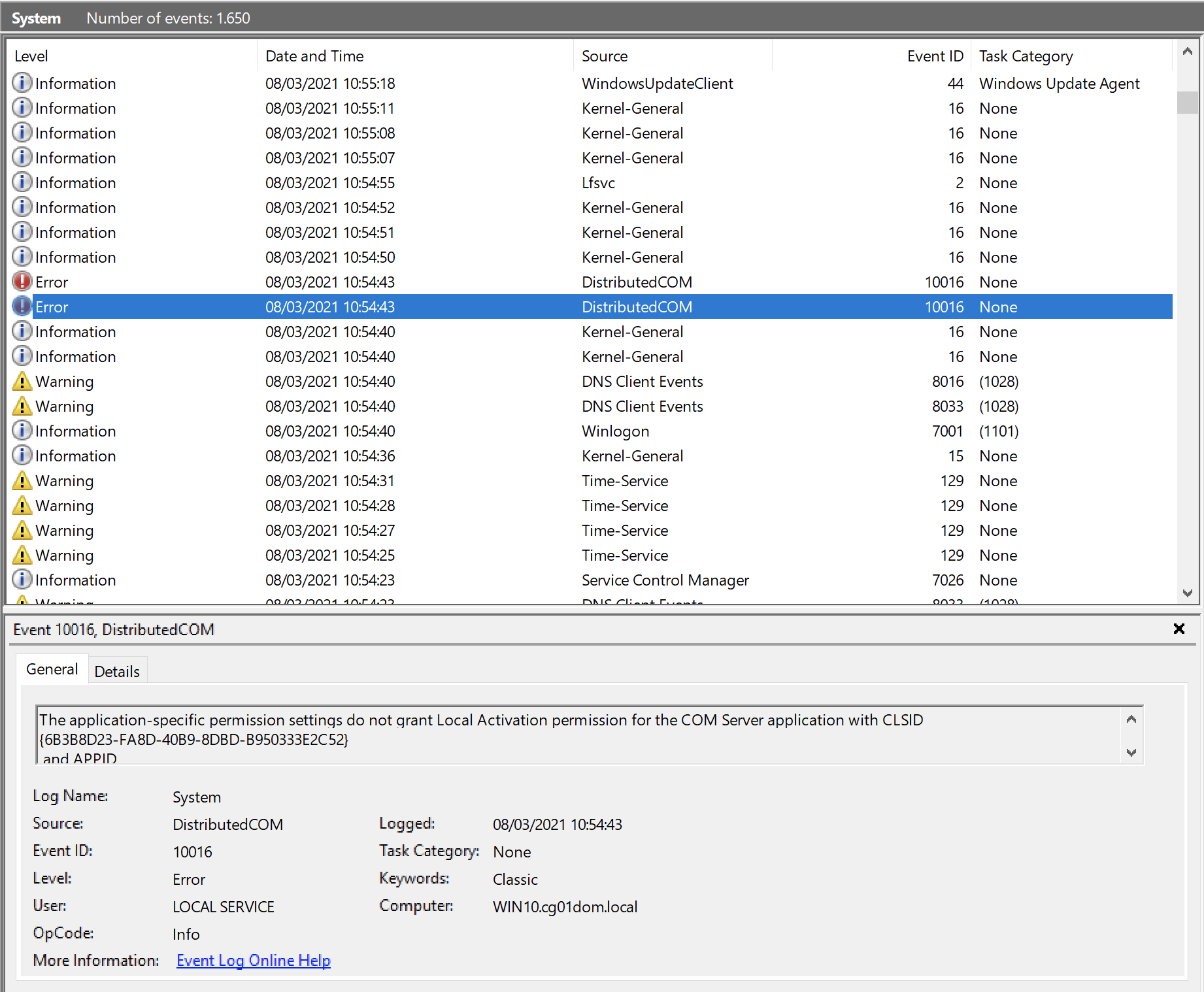
* 1. Security:



* 1. Setup:



* 1. Système: change management, erreurs et défaillance,



## Échelle des priorités

Information < Warning < Error < Critical

1. Relation avec la sécurité du système d’exploitation

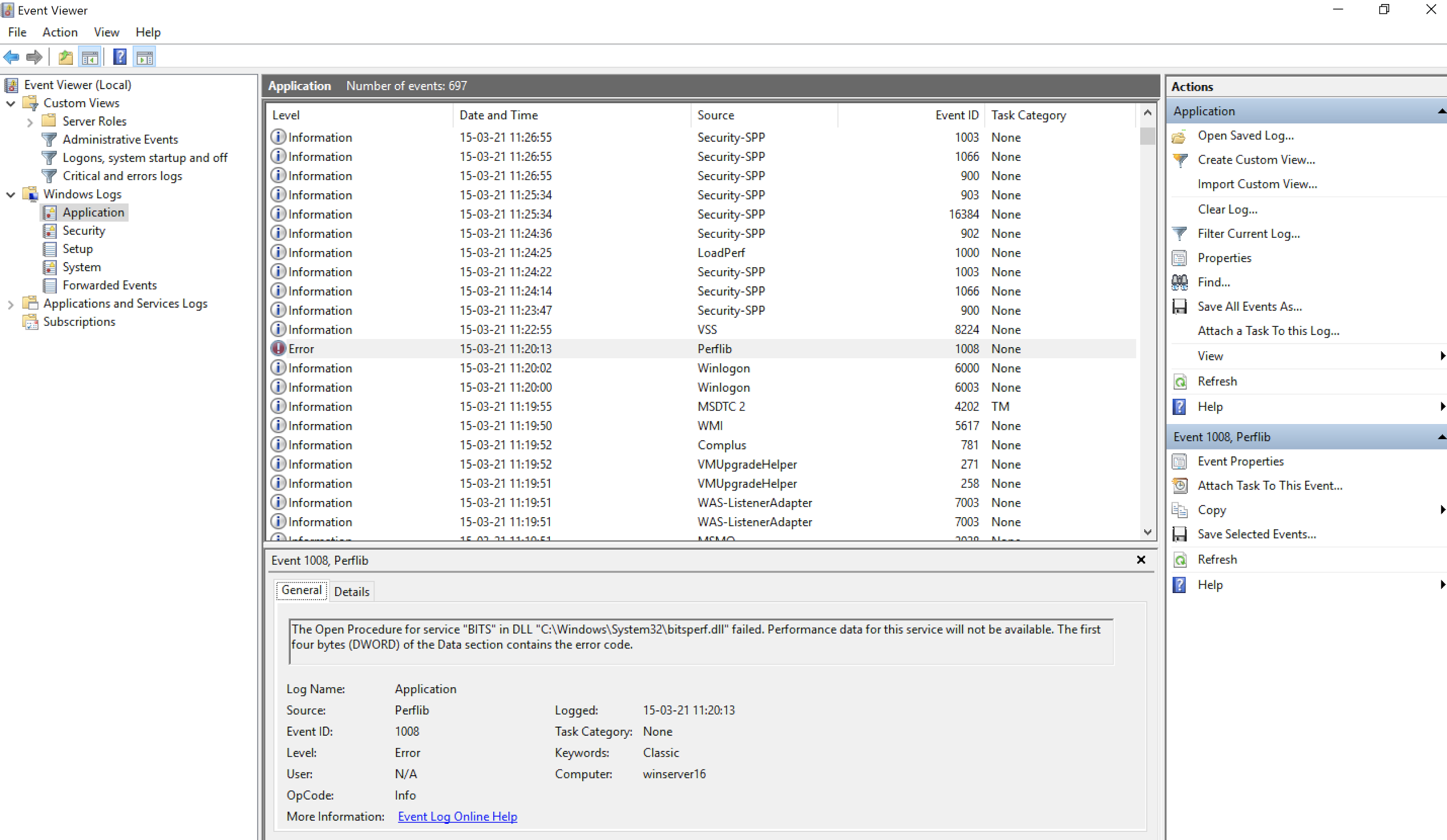
* System:
  + Permissions, erreurs et défaillances du système.
* Setup:
  + Mise à jour, installations de programmes.
* Security :
  + Messages de connexion, déconnexion d’utilisateurs.

1. Comparaison des logs de kali Linux et Windows

Sous Windows, les logs sont en binaires et peuvent être ouverts et lus avec l’Event Log. Ils sont répartis en plusieurs catégories et des filtres personnalisés peuvent être appliqués.  
  
 Sous Linux, les logs sont par défaut en clair dans des fichiers “.log”. Il est possible de corréler les logs entre plusieurs systèmes Linux. Il est plus difficile de catégoriser les logs sous Linux étant donné qu’ils sont divisés en différents fichiers, il faut effectuer cette taxonomie manuellement.

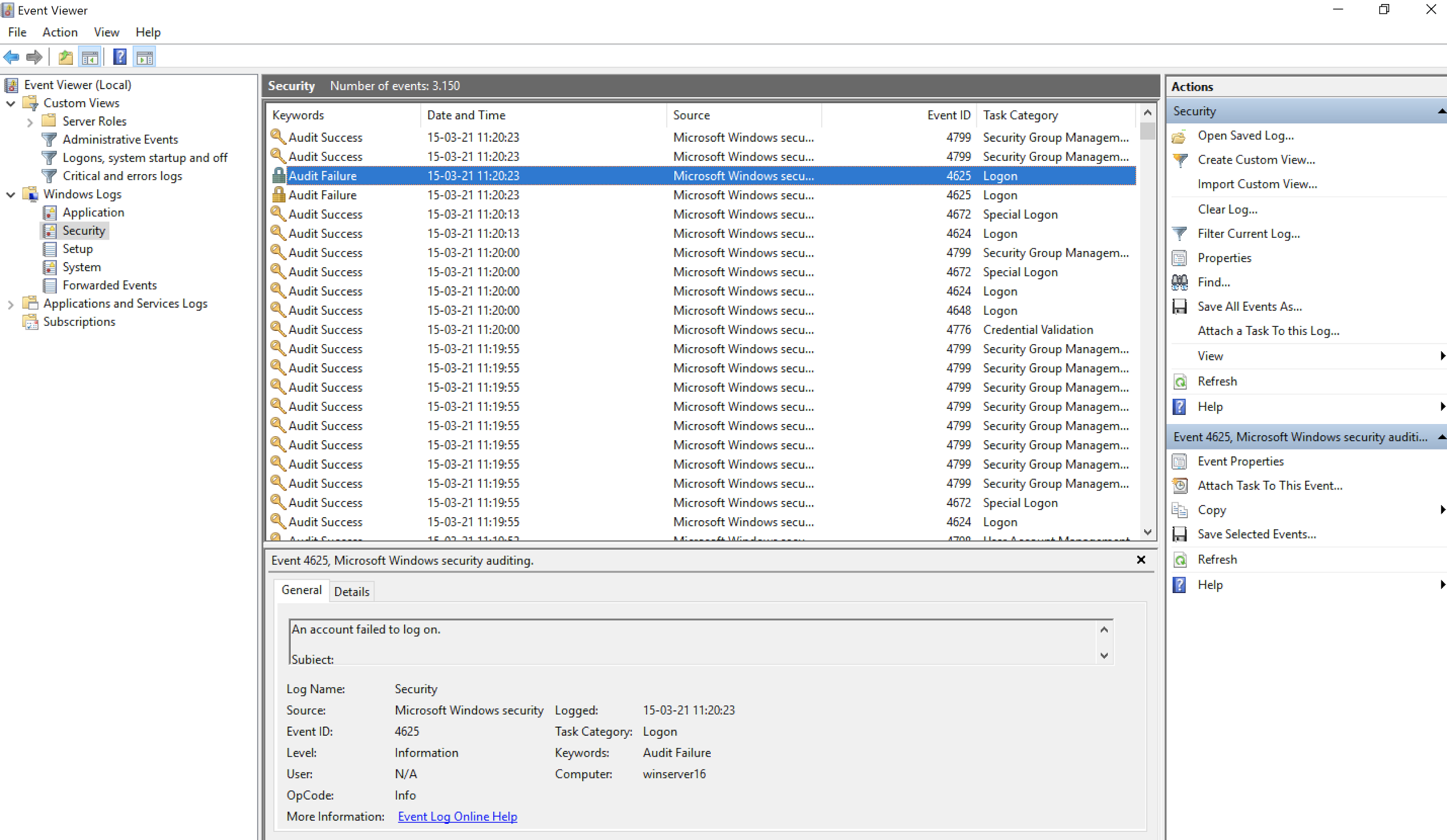
1. Les erreurs identifiées sous Windows

Erreur 1:



The Open Procedure for service ‘BITS’ in dll “C:\Windows\System32\bitsperf.dll” failed. Performance data for this service will not be available. The first four bytes(DWORD) of the Data section contains the error code.

Erreur 2:



An account failed to log on.

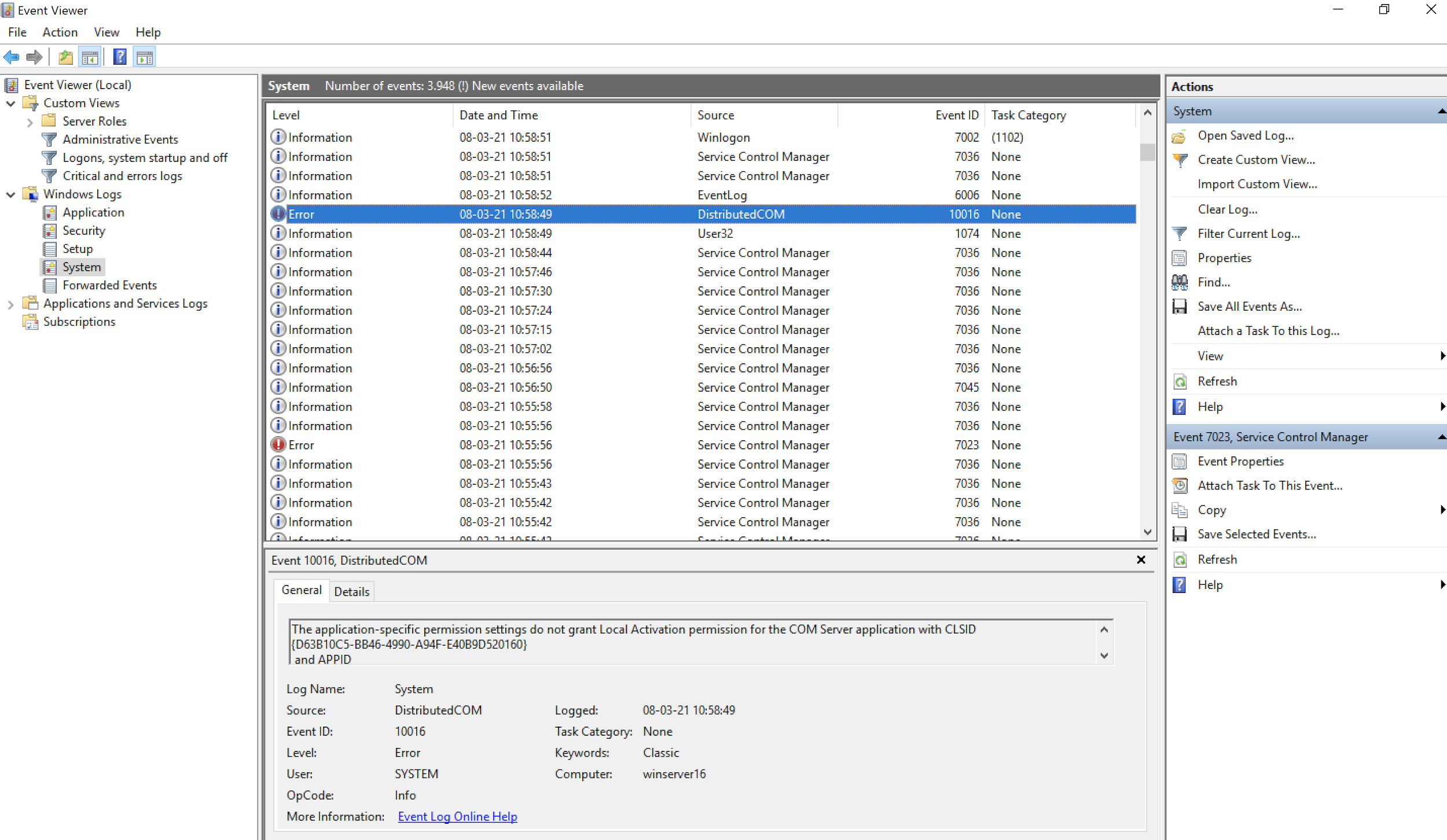
Une connexion au compte utilisateur ratée

Erreur 3:



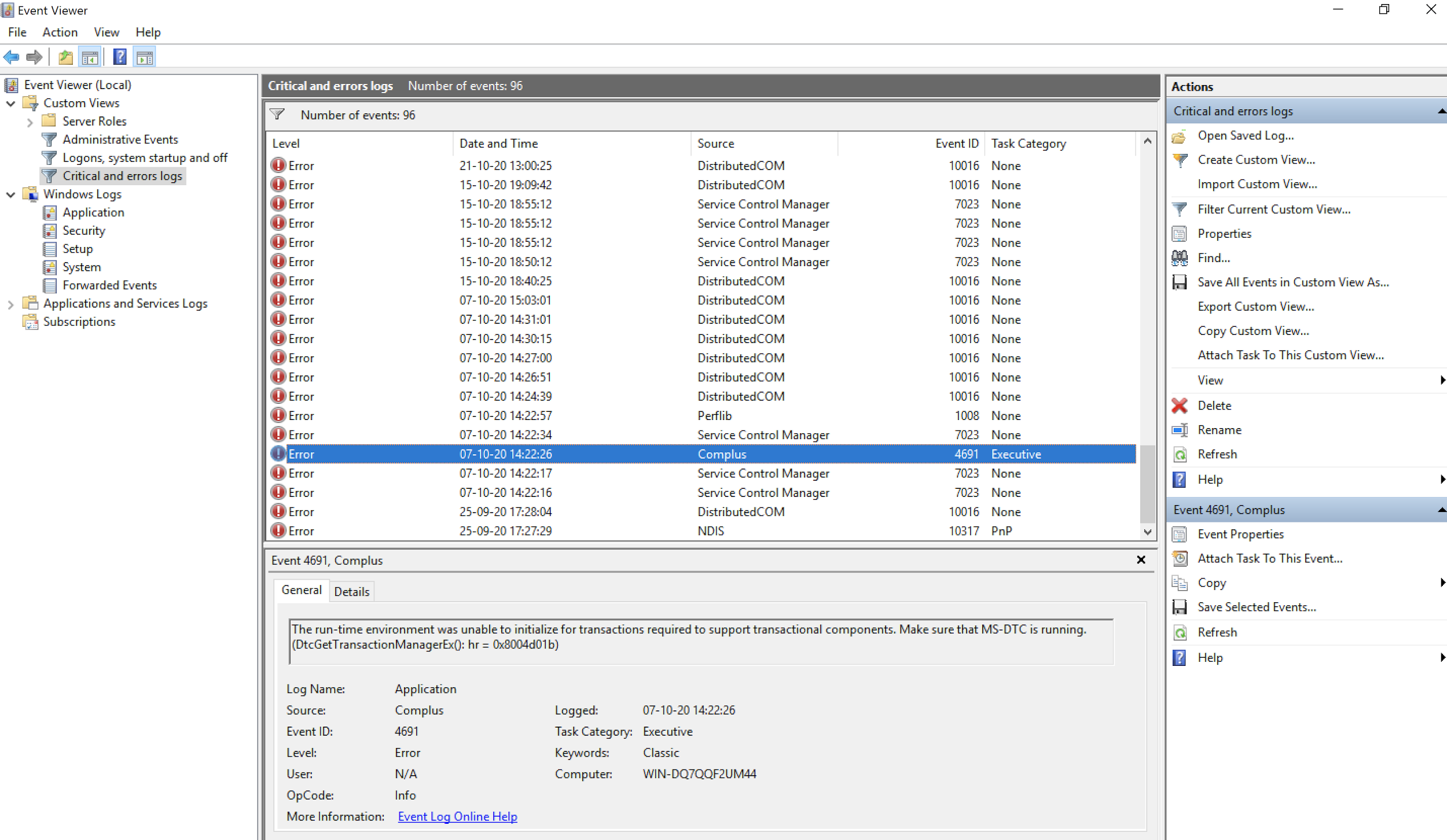
The description for Event ID 7023 from source Service Control Manager cannot be found. Either the component that raises this event is not installed on your local computer or the installation is corrupted. You can install or repair the component on the local computer.

Erreur 4:



This application-specific permission settings do not grant Local Activation permission for the COM Server application with CLSID{D63B1DC5-BB46-4990-A94F-E40B9D520160} and APPID

Erreur 5:



The run-time environment was unable to initialize for transactions components. Make sure that MS-DTC is running.(DtcGetransactionManagerEx(): hr = 0x8004d01b)

# 

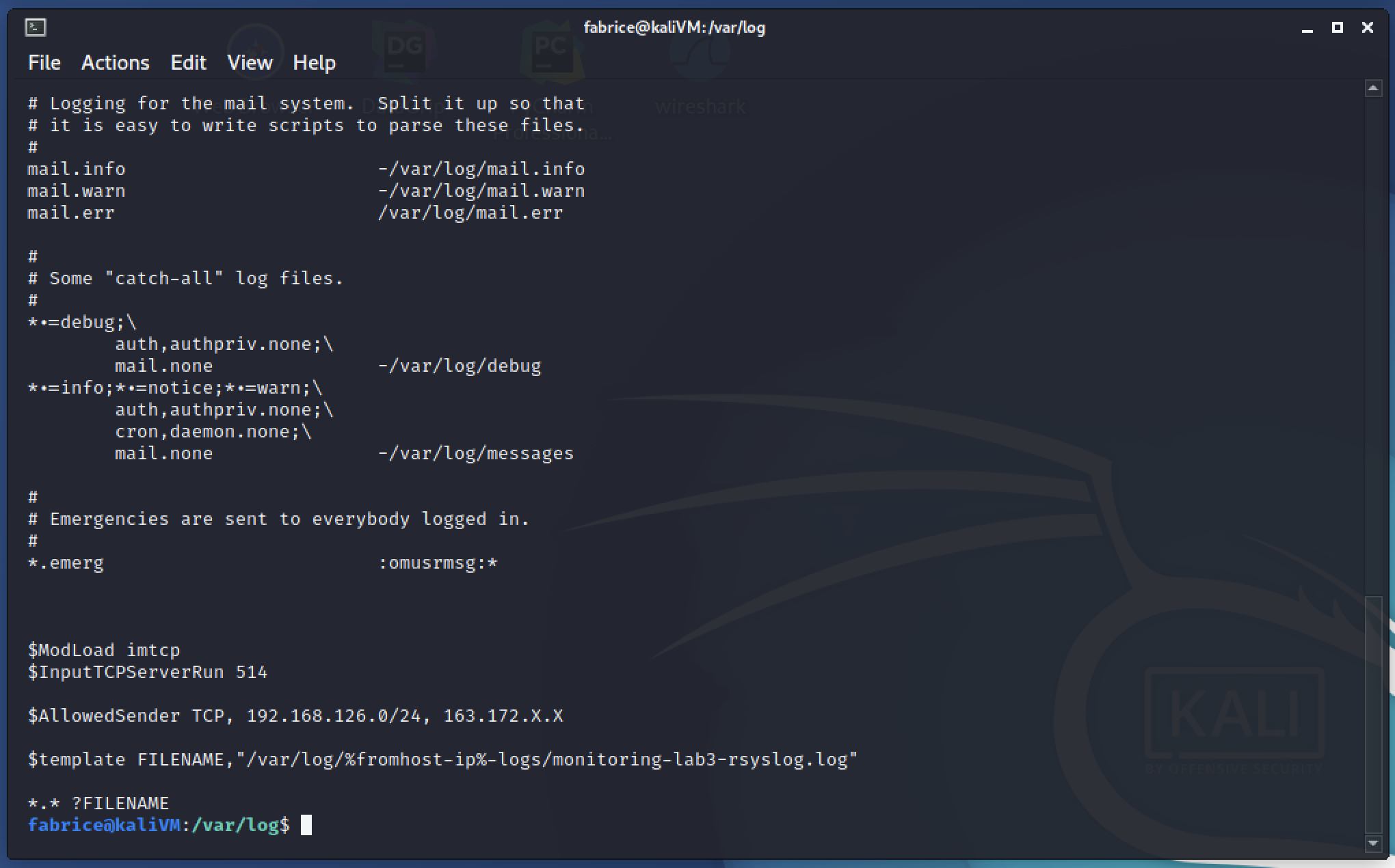
# Monitoring Labo 3 : Rapport

RSyslog sous Linux:

<https://homputersecurity.com/2018/03/01/comment-mettre-en-place-un-serveur-syslog/>

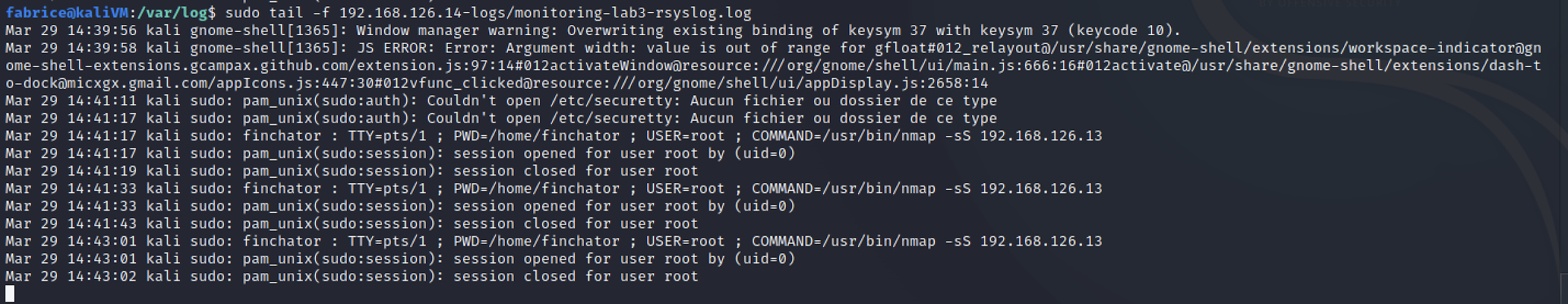
*La configuration du serveur puis du client :*

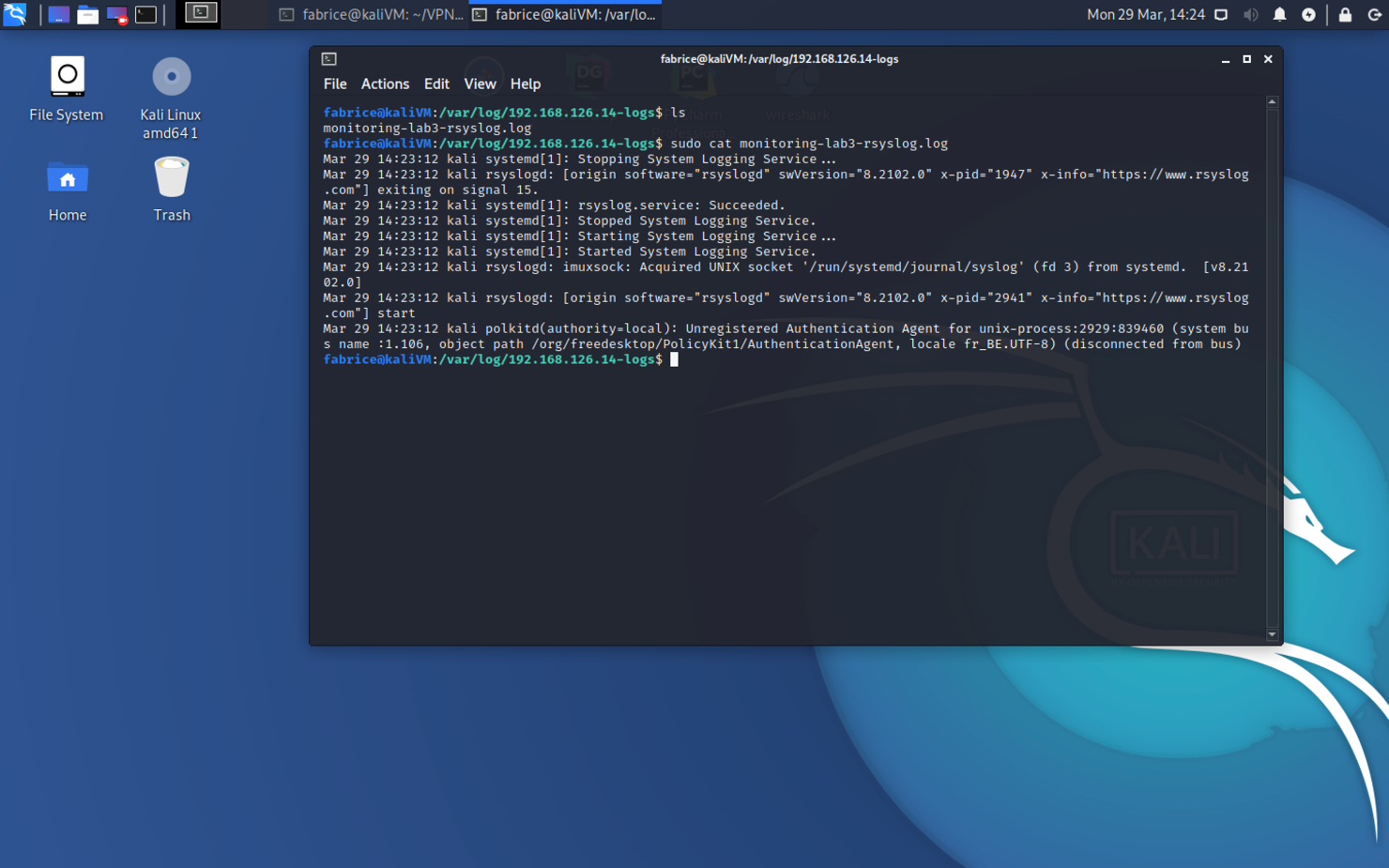
(L’ip du serveur à travers le VPN est 192.168.126.13)





Deux exemples des logs recueillis par le client :





RSyslog sous Windows :

La configuration de l’agent :

Au démarrage de l’agent, il faut créer un nouveau « Rule Set » :

Une image contenant texte

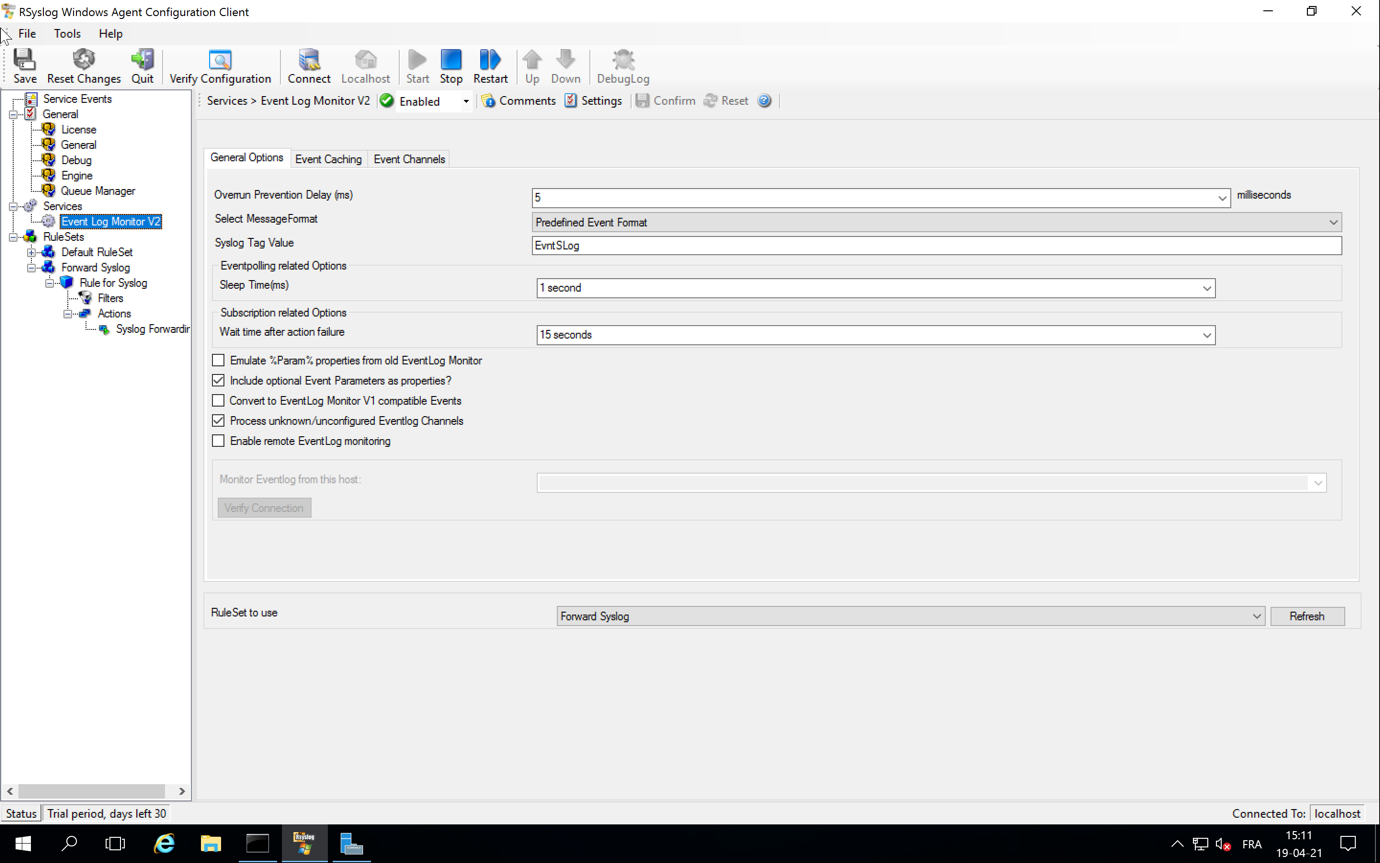
Description générée automatiquement

Ensuite, dans les options de l’action, il faut préciser l’IP du serveur :

Une image contenant texte

Description générée automatiquement

L’onglet « Services > Event Log Monitor V2 » est déjà configurer par défaut, il faut juste sélectionner comme « Rule Set » (en bas de la fenêtre) le « Forward Syslog ».



Cliquer sur « Save » puis sur « Start » pour lancer ler service.

Sur le loghost Linux, il faut restart le service rsyslog (*service restart rsyslog*).

# Monitoring Labo 4 : Rapport

Implémenter chaque action au moins 1 fois pour un des scénarios.

Pour chaque scénario il faut générer un message de log adéquat.

Les étapes du scénario :

1. Produire le scénario  
2. Identifier les traces générées par ce dernier  
3. Etablir, configurer ou construire un outil de détection du scénario  
4. Choisir et programmer une action à entreprendre en cas de détection 5. mettre en place cet outil et le tester.

Les actions possibles :

1. Générer un log : soyez attentifs aux informations de votre message de log : taxonomie, priorité, ...
2. Ecrire un message dans la console d'un ou plusieurs utilisateurs
3. Envoyer un email
4. Activer un script

**Consignes :**

1. **Description de la mise en œuvre des scénarios**
2. **Explication et justification de la méthode de détection employée.**
3. **Exemple de la détection et de l’action.**

**Les scénarios :**

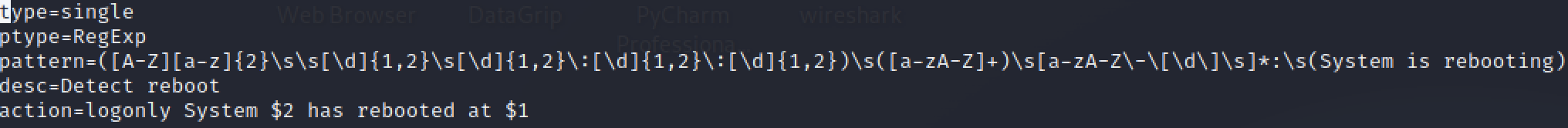
1. Détecter le redémarrage d’un PC. Choisir une méthode.

Nous avons détecter le message ‘System is rebooting » nous allons donc recherche ce message.

Le log :



Notre règle :



La regex :

Nous cherchons un premier groupe quel sera le timestamp. Un deuxième avec le nom de la machine et le troisième avec le message. Entre le 2e et le 3e, nous ne voulons pas prendre ce qu’il y a.

Le résultat :

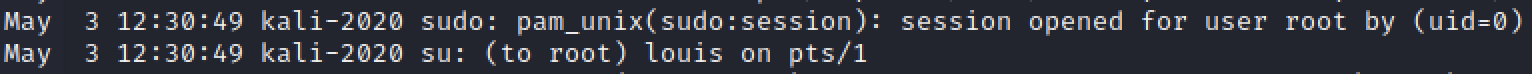
Une image contenant texte, clavier, capture d’écran

Description générée automatiquement

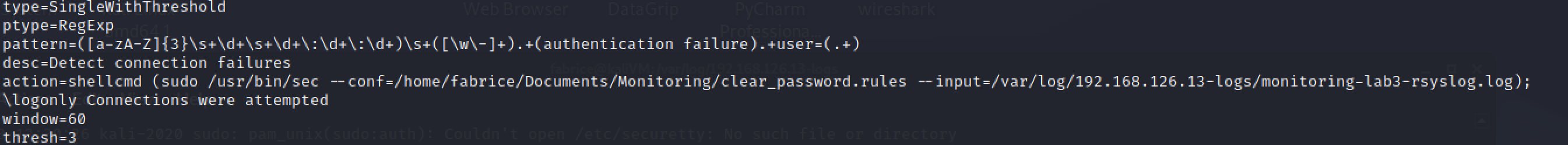
1. Détecter une série de tentatives de connexion avec une réussite à la fin.
   1. Identifier si c’est une attaque et agir en conséquence.

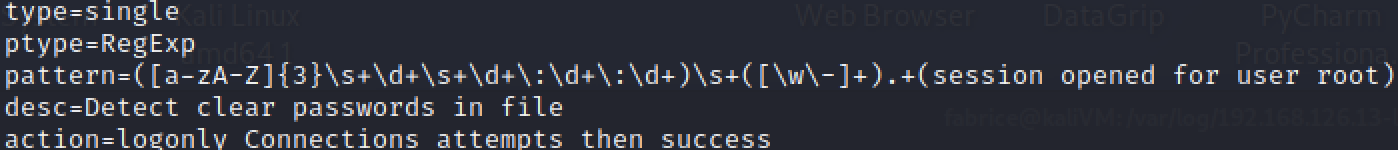
Le log :





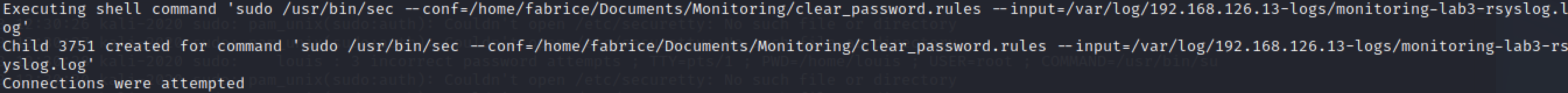
Notre règle :





La regex :

Le résultat :



* 1. Détecter lorsque l’utilisateur a entré son mot de passe à la place de son nom d’utilisateur.

Le log :

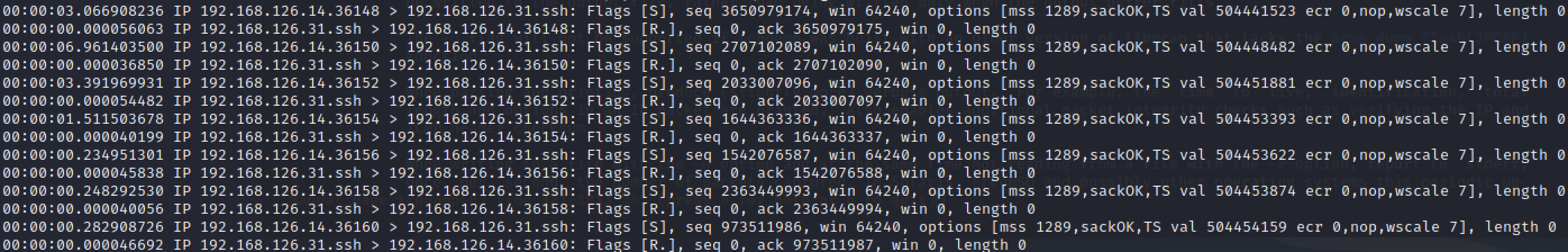
Notre règle :

La regex :

Le résultat :

1. Détecter si un même PC essaie plusieurs connexions vers un même PC/serveur. (tcpdump)

Le log :



Notre règle :

La regex :

Le résultat :

1. Détecter si un « super utilisateur » effectue des actions à distance.

Le log :

Notre règle :

La regex :

Le résultat :

1. Détecter s’il y a une connexion en cascade : pc-1 passe par pc-3 pour se connecter à pc-2.

Le log :

Notre règle :

La regex :

Le résultat :

1. Détecter si la [faille](https://www.it-connect.fr/une-faille-dans-sudo-vieille-de-10-ans-touche-linux/) est utilisée.

Le log :

Notre règle :

La regex :

Le résultat :

1. Détecter si une interface du réseau passe en mode promiscuous.

Le log :

Notre règle :

La regex :

Le résultat :

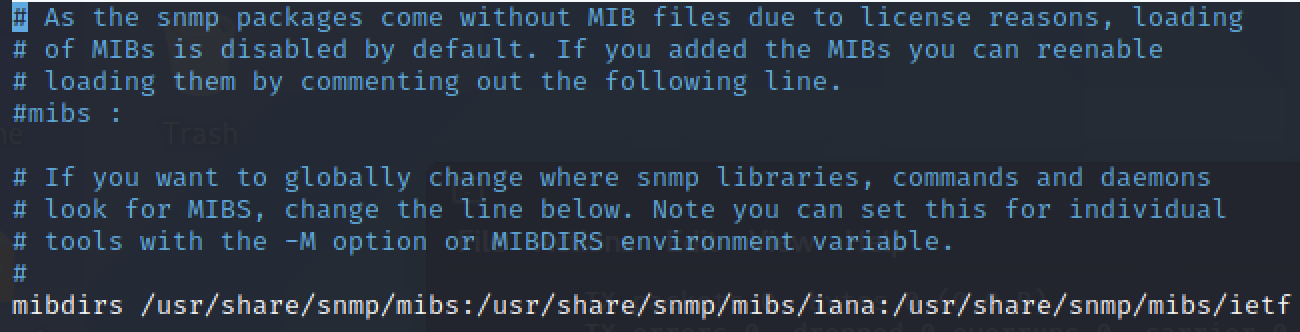
Bonus :

1. Configurer client mail avec mon adresse HELMo. Effectuer des échanges avec les coéquipiers. Détecter les envois hors de HELMo et si possible les pièces jointes.
2. Détecter les scans de ports.

# Monitoring Labo 5 : SNMP

*Configurer SNMP :*

* + Apt-get install snmpd
  + Systemctl start snmpd
  + Modifier snmp.conf :



* + Modifier snmpd.conf (ajouter les OID) :

Une image contenant texte

Description générée automatiquement

* + Restart le service

*Données de configuration :*

* Système :
* **Snmpwalk -v 2c -c public 192.168.133.6 .1.3.6.1.2.1.1**

Résultat :

* Configuration CPU:

Résultat :

* La configuration des mémoires (RAM, disques, …) :

Résultat :

* La configuration des interfaces :

Résultat :

*Données de surveillances :*

Les données à surveiller :

* + Usage du CPU
  + Usage de la RAM
  + Les flux entrants et sortants des interfaces
  + Usage des disques

A présenter via des graphiques (via Excel).

Script qui récolte les données toutes les 4 minutes :