PROJET EN ENTREPRISE ÉPREUVE UE52

Surveillance d'un serveur WEB et d'un serveur VSFTPD avec une SIEM

BRET Sébastien Licence Pro. MIASSR RE
PROEF

SOMMAIRE

Résumé

Le projet se résume en la préparation et l'installation en amont d'un serveur Apache 2, d'un serveur Vsftpd et de WAZUH qui est un HIDS avec 2 modules de SIEM qui va surveiller nos 2 serveur pour voir s'il y a une tentative de rentrer par force ou une modification de fichier qui peut être soit d'ajouter, de supprimer ou de changer ce qui se trouve dans le fichier.

Abstract

The project sums up on the preparation and upstream installation of an Apache 2 server, a Vsfttpd server, and WAZUH which is a SIEM which will monitor our 2 servers to see if there is an attempt to brute force log or modify a file which can either be to add, remove or change what is in the file.

1. Introduction

Le projet consiste en la préparation et l'installation d'un serveur WAZUH qui va surveiller 2 serveurs un serveur Web (apache2) et un serveur VSFTPD (FTP). Je présenterai dans un premier temps ce qu'est un HIDS et le serveur WAZUH avec c'est 2module de SIEM. Pour la deuxième partie ce sera la partie technique et simulation intrusion. Je terminerai par une conclusion en résumant les points à améliorer et les améliorations possibles.

2. Présentation de l’entreprise d’accueil PROEF France

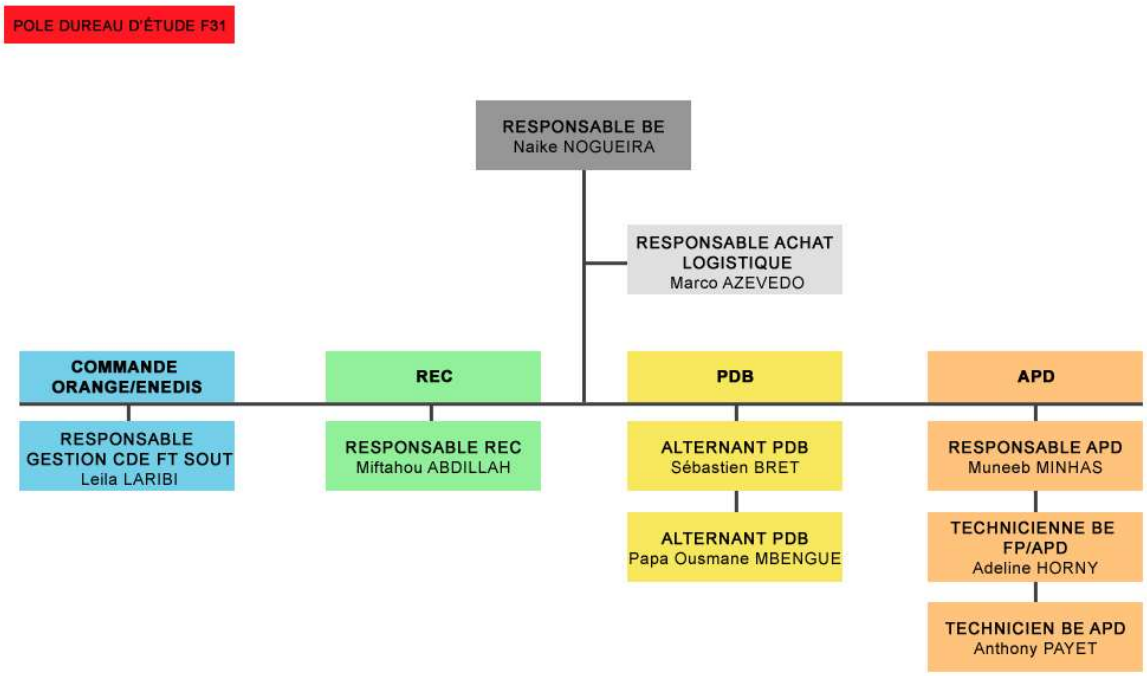
PROEF France est une **S**ociété par **A**ctions **S**implifiée (SAS) en activité depuis 10 ans.

Implantée à BUSSY-SAINT-MARTIN (77600), elle est spécialisée dans le secteur d'activité de la construction de réseaux électriques et de télécommunications.

Son effectif est compris entre 20 et 49 salariés. Sur l'année 2019 elle réalise un chiffre d'affaires de 21 999 400,00 €.

Le total du bilan a augmenté de 55,84% entre 2018 et 2019. Societe.com recense 3 établissements et 3 événements notables depuis un an. Jean-Jacques PANY, est président du conseil d'administration de l'entreprise PROEF FRANCE SAS.

2.1Organigramme du bureau d’étude

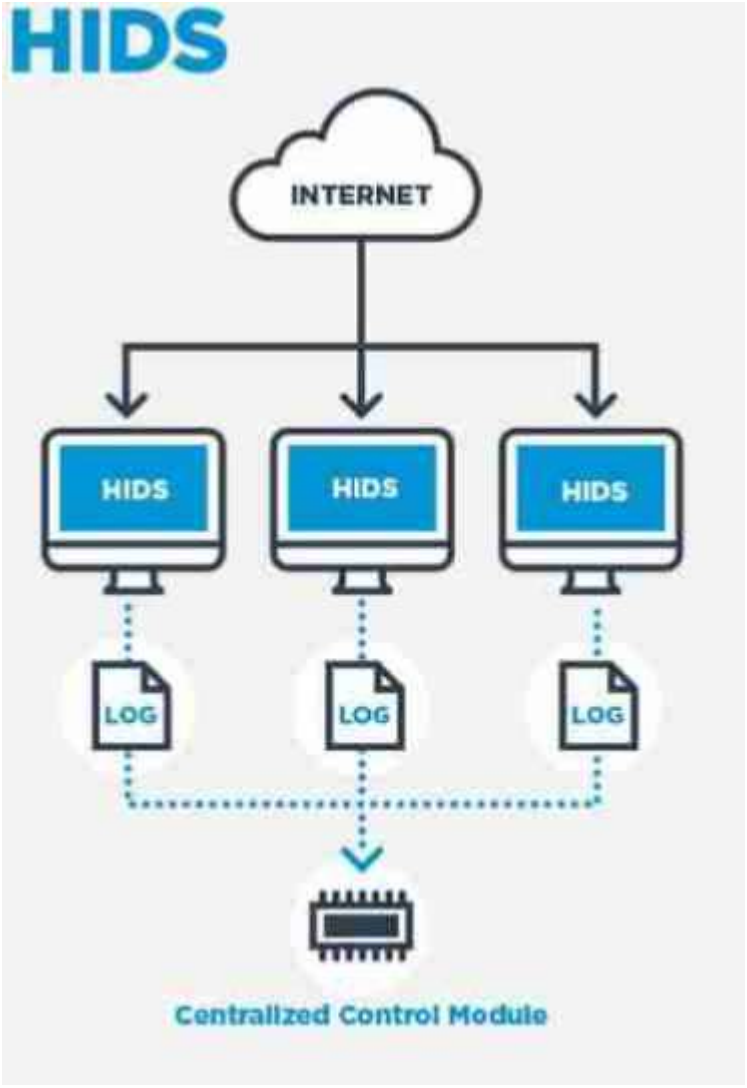


Je remercie toute l’équipe de Fibre 31 et l’équipe losange qui font partie de PROEF France.

Qui mon accueilli pendant ces 3 ans et qui m’ont appris beaucoup. Je remercie Naike mon manageur qui est ma tutrice. Elle m’a suivi tout le long de ma formation avec mes Collègues Muneeb, Anthony, Adeline, Leila, Miftahou et Papa.

3. Présentation d’un HIDS

HIDS : Host-based intrusion detection system



Un HIDS est un système de surveillance il permet de surveiller les systèmes sur lequel il est installé dans le but de constater une intrusion, ou une tentative et de prévenir le SOC qui pourra intervenir en fonction du type d’alerte.

SOC : Security Operation Center est une division d’une entreprise qui assure la sécurité de l’organisation avec de la supervision grâce à des logiciels de traitement de données.

3.1 WAZUH

WAZUH est un HIDS Open source qui est utilisé pour la prévention et la détection de menace grâce a beaucoup de modules. WAZUH possède 2 modules de SIEM.

Le module « Security event » et le module « Integrity monitoring »

3.1.1 Présentation de la SIEM EVENT (Security event)

La SIEM EVENT permet de remonter toutes les alertes de sécurité et se concentrer sur les logs grâce à un agent sur le serveur.

3.1.2 Présentation de la SIEM FIM (Integrity monitoring)

La SIEM FIM permet de surveiller et de remonter si un fichier est créé, effacé ou modifié grâce à un agent sur le serveur. La FIM peut être configurée de différentes manières afin d'être plus ou moins réactive, là on l'utilise en temps réel mais de base elle est configurée pour contrôler les modifications une fois par jour par exemple, normalement cela ne se fais pas en temps réel.

3.1.3 Présentation des agents WAZUH

Les agents WAZUH sont des collecteurs d'informations qu'ils transmettent au serveur WAZUH. Les agents collectent les informations seulement sur les serveurs où ils sont installés. Les agents peuvent être associés à des groupes et suivent des règles qui sont inscrites sur le serveur WAZUH. Les agents WAZUH remontent les logs en temps réel vers le serveur WAZUH qui va pouvoir générer des alertes en fonction des règles qui sont préétablies par la communauté. Les alertes sont considérées comme un event de sécurité, event qui peut être consulté grâce à l'interface graphique. Grâce au fait que WAZUH est open source la plupart des technologies utilisées ont déjà des règles préétablies et ne demandent pas de configuration spécifique. Les agents peuvent être configurés en temps réel et en masse grâce à leur groupe d'affectation, on peut déployer de nouvelles règles sur tous les agents d'un même groupe sans redémarrer leur service et sans devoir faire la modification à la main sur chaque serveur.

4. Contexte

L'entreprise PROEF voulait pouvoir sécuriser ses deux serveurs, un serveur WEB et un serveur VSFTPD. Elle a pensé installer un serveur SIEM pour permettre d'être prévenu au cas où il y aurait une attaque par log ou une modification de fichier.

5. Problématique

La Problématique étant de trouver une SIEM qui pourrait avoir deux modules, un qui pourrait repérer toute infraction par log, et un autre module qui surveillerait l'intégrité des fichiers.

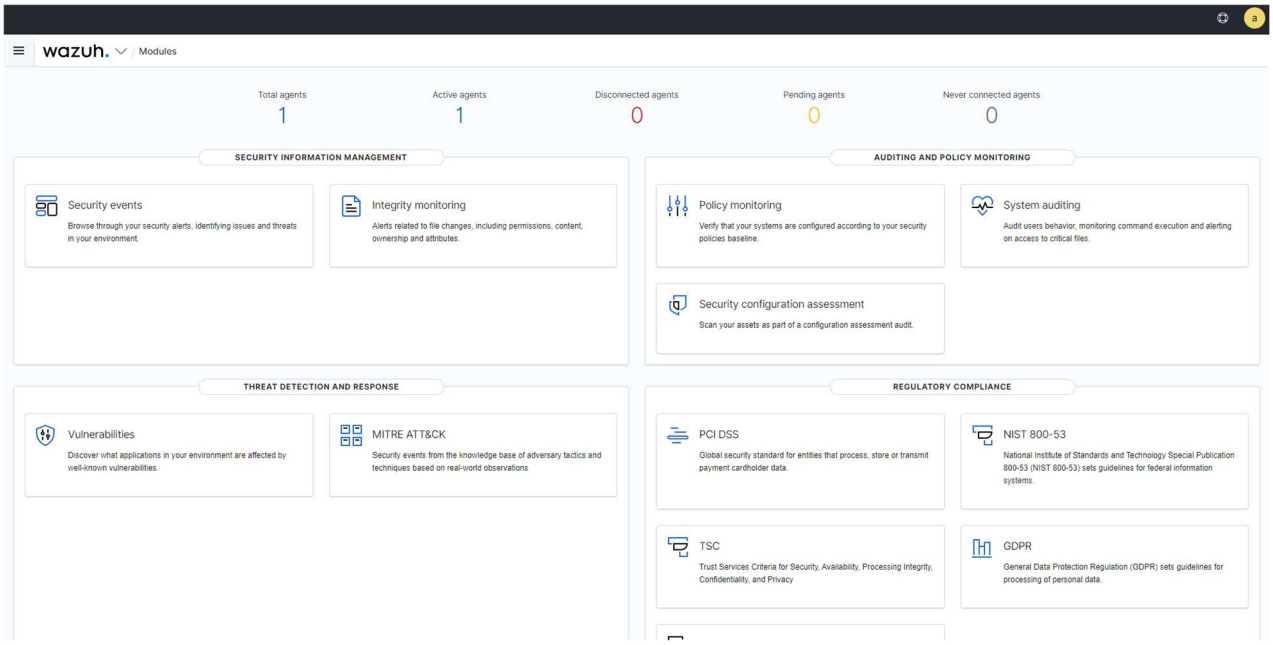
- Enjeux et risques :

Le but étant de surveiller les deux serveurs afin d'éviter tout risque de fuites d'informations ou de données erronées.

6. Solution

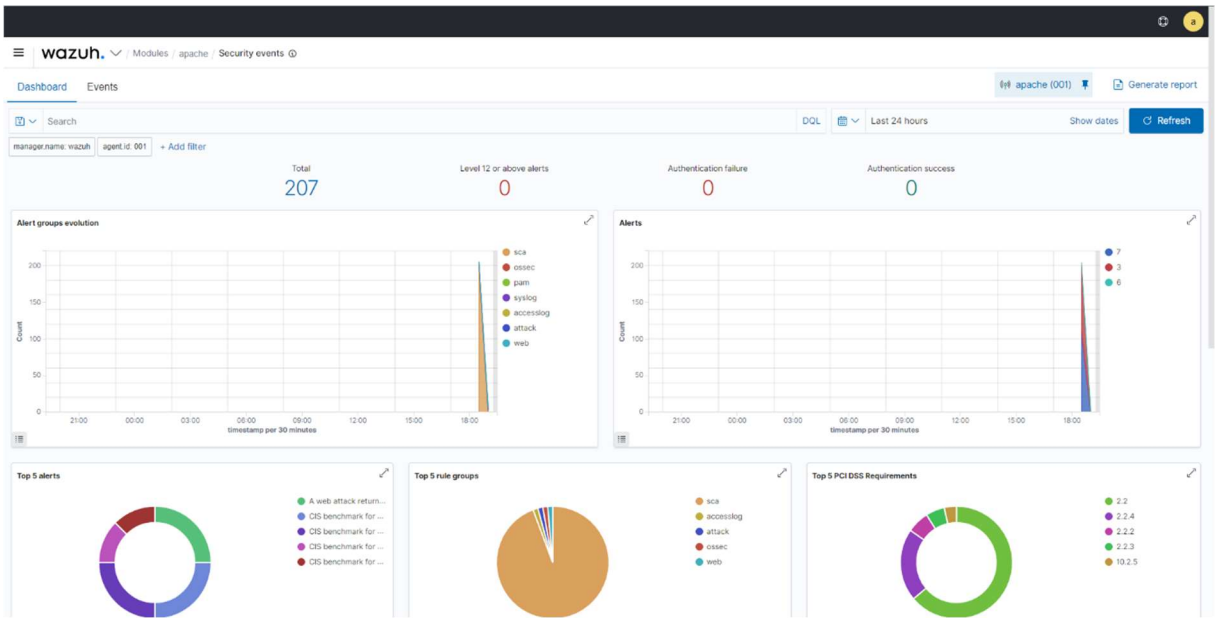
La solution mise en place est de créer un nouveau serveur WAZUH qui a deux modules, un module security event et un module integrity monitoring et de mettre en place des agents de WAZUH sur les deux serveurs web et VSFTPD et tester la solution en essayant de pirater pour voir si les infractions sont bien remontées.

7. Surveillance du serveur apache (WEB)



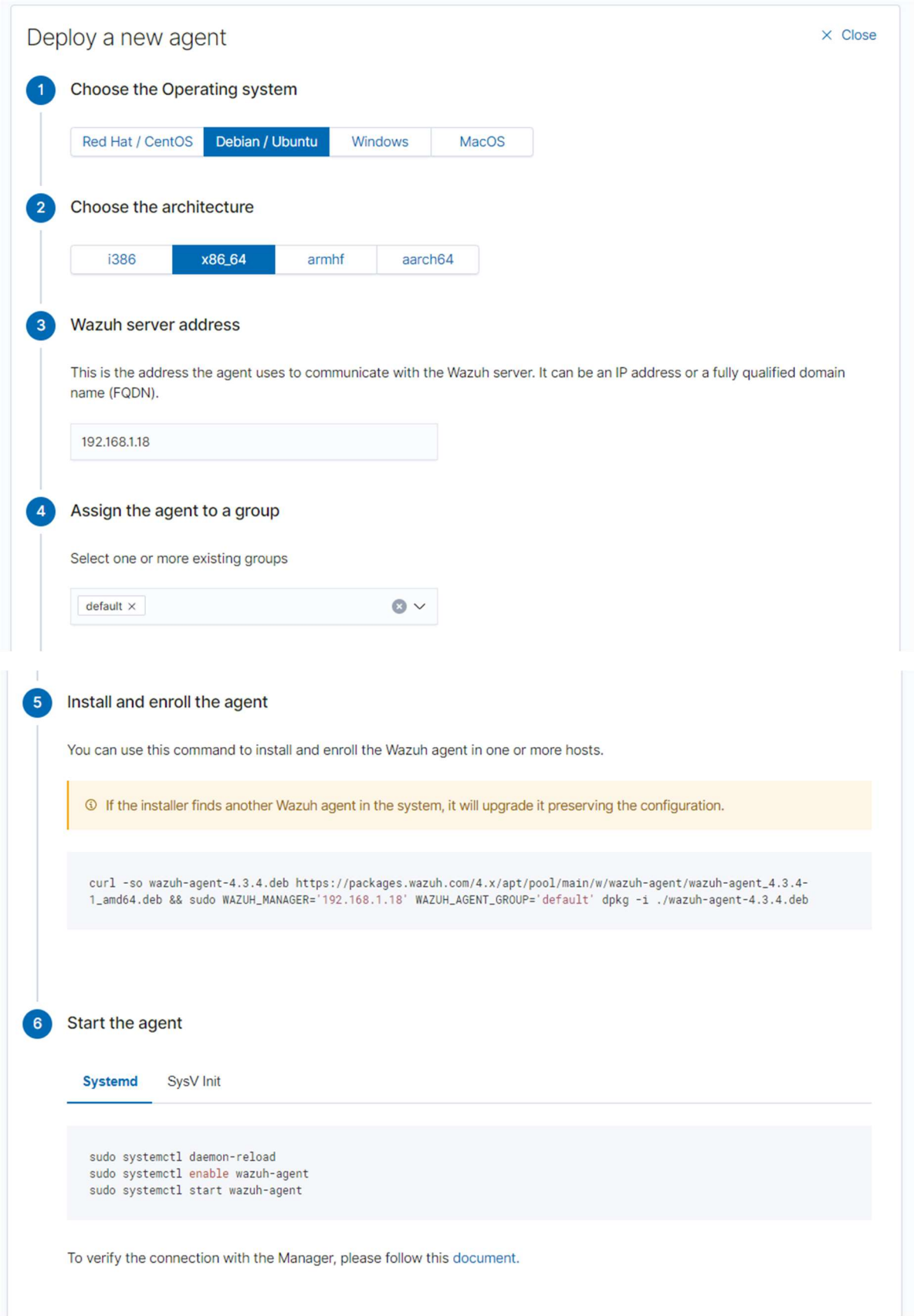
Sur l'interface graphique de WAZUH nous pouvons voir qu'on a plusieurs modules.

On va s'intéresser à deux modules plus spécialement à « Security event » et « integrity monitoring ». On peut retrouver toutes les règles sur le serveur WAZUH dans le dossier « /var/ossec/ruleset/rules » dans des fichiers xml qui sont classées par famille.



Sur l'interface WAZUH dans security event nous pouvons voir les alertes en mode graphique.

Pour commencer à surveiller le premier serveur WEB il nous faut installer un agent sur le serveur WEB. Sur l'interface graphique il faut créer un nouvel agent.

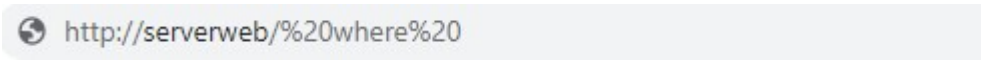


- L'étape 1 : nous devons dire sur quel système est notre serveur.
- L'étape 2 : on choisit l'architecture x86_64
- L'étape 3 : nous devons indiquer l'adresse IP du serveur WAZUH
- L'étape 4 : nous choisissons un groupe, on a mis groupe par défaut
- L'étape 5 : nous pouvons voir une ligne de code qu'il faudra entrer sur le serveur web pour le mettre en place.
- L'étape 6 : nous avons les lignes de code pour démarrer l'agent.

Avec la commande « sudo systemctl statut wazuh-agent »

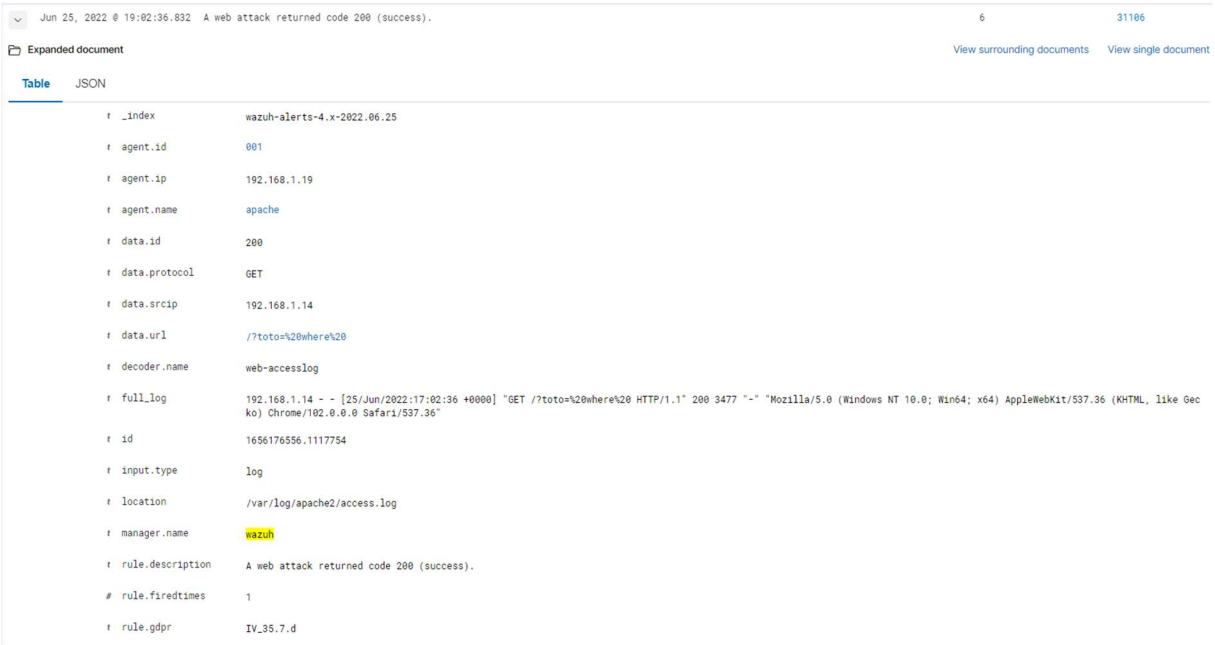
Nous pouvons voir si l'agent est activé.

Pour pouvoir tester si l'agent nous remonte l'information d'une tentative d'intrusion ou d'une réussite :



Sur la barre de recherche nous simulons une action qui pourrait faire penser à une injection SQL.

Sur le module security event nous avons bien une alerte injection SQL.



Sur le serveur WAZUH nous avons bien les règles qui nous permettent de voir s'il y a une injection SQL.

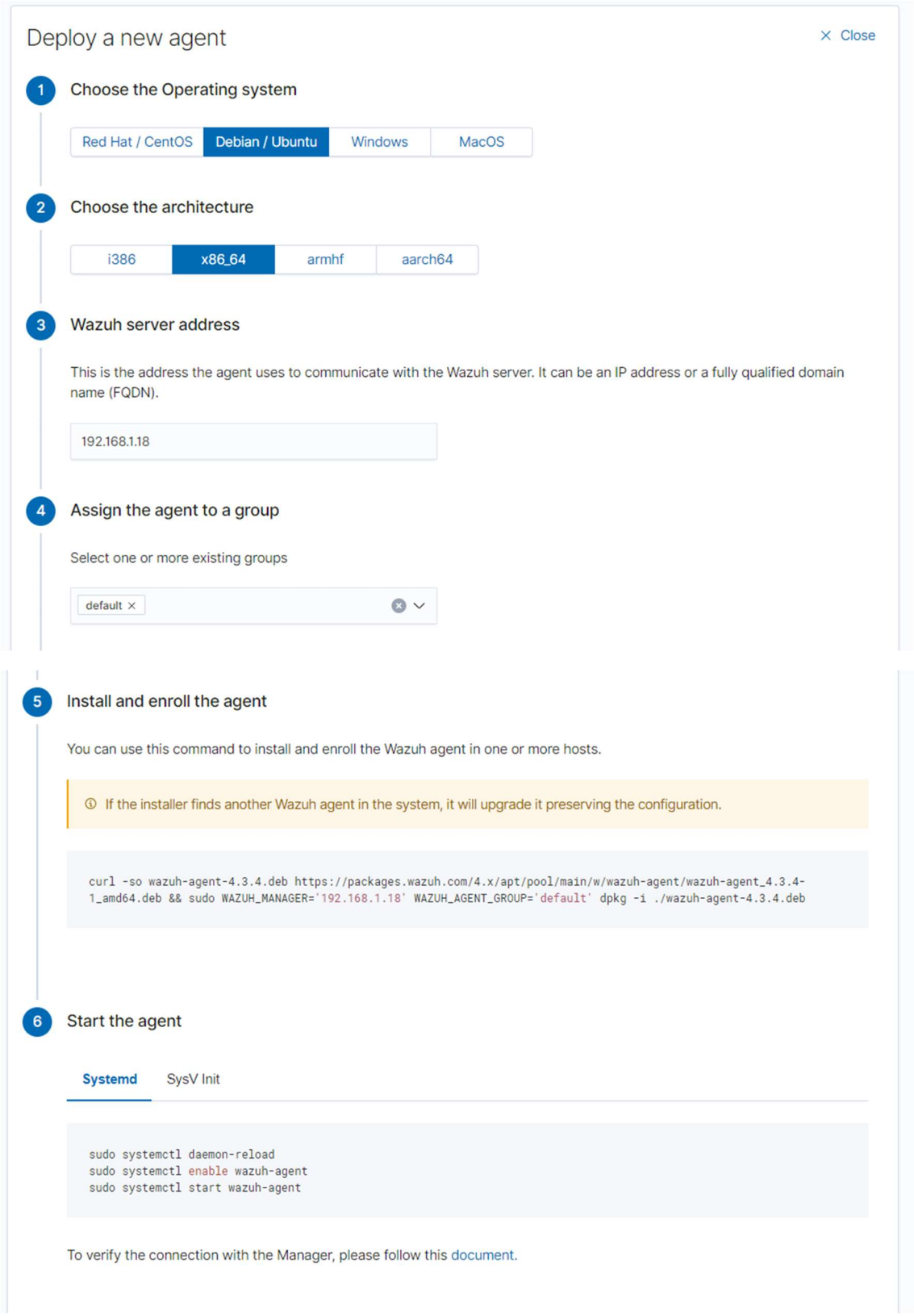
Les règles qui me font remonter l'information sont :

```
  <rule id="31106" level="6">
    <if_sid>31103, 31104, 31105</if_sid>
    <id>^200</id>
    <description>A web attack returned code 200 (success).</description>
    <mitre>
      <id>T1190</id>
    </mitre>
    <group>attack,pci_dss_6.5,pci_dss_11.4,gdpr_IV_35.7.d,nist_800_53_SA.11,
.6,tsc_CC7.1,tsc_CC8.1,tsc_CC6.1,tsc_CC6.8,tsc_CC7.2,tsc_CC7.3,</group>
  </rule>
```

```
  <rule id="31103" level="7">
    <if_sid>31100,31108</if_sid>
    <url>=select%20|select+|insert%20|%20from%20|%20where%20|union%20|</url>
    <url>union+|where+|null,null|xp_cmdshell</url>
    <description>SQL injection attempt.</description>
    <mitre>
      <id>T1190</id>
    </mitre>
    <group>attack,sql_injection,pci_dss_6.5,pci_dss_11.4,pci_dss_6.5.1,gdpr_IV_35.7.d,nist_800_53_SA
11,nist_800_53_SI.4,tsc_CC6.6,tsc_CC7.1,tsc_CC8.1,tsc_CC6.1,tsc_CC6.8,tsc_CC7.2,tsc_CC7.3,</group>
  </rule>
```

8. Surveillance d’un serveur VSFTPD (FTP)

Pour surveiller le deuxième serveur VSFTPD il faut installer un nouvel agent sur ce même serveur. Sur l’interface graphique il faut créer un nouvel agent.



C’est la même méthode que pour le premier agent.

Nous allons tester l’agent et voir s’il remonte des infractions par 3 méthodes.

- La première méthode par les logs en essayant de forcer avec la méthode brute force, en imaginant que j’ai le mot de passe mais pas le nom de compte.

  J’essaye de me connecter plusieurs fois au minimum huit fois dans un délai de deux minutes.

  J’obtiens bien une alerte dans security event.

›	Jun 25, 2022 @ 20:48:53.550	vsftpd: FTP brute force (multiple failed logins).
›	Jun 25, 2022 @ 20:48:51.550	vsftpd: FTP session opened.
›	Jun 25, 2022 @ 20:48:5 ⊕ ⊖	PAM: Multiple failed logins in a small period of time.
›	Jun 25, 2022 @ 20:48:49.545	vsftpd: Login failed accessing the FTP server.
›	Jun 25, 2022 @ 20:48:47.643	PAM: User login failed.
›	Jun 25, 2022 @ 20:48:47.643	vsftpd: FTP session opened.
›	Jun 25, 2022 @ 20:48:47.643	vsftpd: Login failed accessing the FTP server.
›	Jun 25, 2022 @ 20:48:45.549	vsftpd: FTP session opened.
›	Jun 25, 2022 @ 20:48:45.542	vsftpd: Login failed accessing the FTP server.
›	Jun 25, 2022 @ 20:48:45.540	PAM: User login failed.
›	Jun 25, 2022 @ 20:48:43.541	vsftpd: Login failed accessing the FTP server.
›	Jun 25, 2022 @ 20:48:43.539	vsftpd: FTP session opened.
›	Jun 25, 2022 @ 20:48:43.536	PAM: User login failed.

Table JSON

t	_index	wazuh-alerts-4.x-2022.06.25
t	agent.id	002
t	agent.ip	192.168.1.21
t	agent.name	vsftpd
t	data.dstuser	vsftpseb54
t	data.srcip	::ffff:192.168.1.14
t	data.status	FAIL LOGIN
t	decoder.name	vsftpd
t	decoder.parent	vsftpd
t	full_log	Sat Jun 25 18:48:51 2022 [pid 4058] [vsftpseb54] FAIL LOGIN: Client "::ffff:192.168.1.14"
t	id	1656182933.1716071
t	input.type	log
t	location	/var/log/vsftpd.log
t	manager.name	wazuh
t	previous_output	Sat Jun 25 18:48:49 2022 [pid 4056] [vsftpseb54] FAIL LOGIN: Client "::ffff:192.168.1.14" Sat Jun 25 18:48:46 2022 [pid 4054] [vsftpseb54] FAIL LOGIN: Client "::ffff:192.168.1.14" Sat Jun 25 18:48:43 2022 [pid 4052] [vsftpseb3] FAIL LOGIN: Client "::ffff:192.168.1.14" Sat Jun 25 18:48:43 2022 [pid 4050] [vsftpseb] FAIL LOGIN: Client "::ffff:192.168.1.14" Sat Jun 25 18:48:35 2022 [pid 4048] [vsftpseb] FAIL LOGIN: Client "::ffff:192.168.1.14" Sat Jun 25 18:48:25 2022 [pid 4045] [vsftp2] FAIL LOGIN: Client "::ffff:192.168.1.14" Sat Jun 25 18:48:17 2022 [pid 4042] [vsftp2] FAIL LOGIN: Client "::ffff:192.168.1.14"

Là on peut voir que j'ai testé plusieurs logins différents comme seb54, seb3, seb.

Les règles qui me font remonter l'information sont :

```
<rule id="11403" level="5">
  <if_sid>11400</if_sid>
  <match>FAIL LOGIN: </match>
  <description>vsftpd: Login failed accessing the FTP server.</description>
  <group>authentication_failed,pci_dss_10.2.4,pci_dss_10.2.5,gpg13_7.1,gdpr_IV_35.7.d,gdpr_IV_32
</rule>
```

```
<rule id="11451" level="10" frequency="8" timeframe="120">
  <if_matched_sid>11403</if_matched_sid>
  <same_source_ip />
  <description>vsftpd: FTP brute force (multiple failed logins).</description>
  <mitre>
    <id>T1110</id>
  </mitre>
  <group>authentication_failures,pci_dss_10.2.4,pci_dss_10.2.5,pci_dss_11.4,gpg13_7.1,gdpr_IV_35.
</rule>
```

On peut voir dans la deuxième règle si une personne essaye de se connecter huit fois ou plus en moins de deux minutes on appelle cela un « brute force ».

- Pour la deuxième méthode nous allons imaginer que j'ai réussi à rentrer dans le serveur FTP et que j'ai des intentions de nuire en modifiant des fichiers de configuration présents dans un dossier d'admin qui ne devraient pas être accessibles depuis le ftp.

  Dans un premier temps je vais sur le WAZUH pour rajouter une règle à mes agents.

  Dans la configuration de l'agent je rentre l'emplacement du fichier de logs de VSFTPD car il n'est pas présent dans la configuration de base. Cela permet à l'agent de signaler si il y a une interaction avec ce fichier.

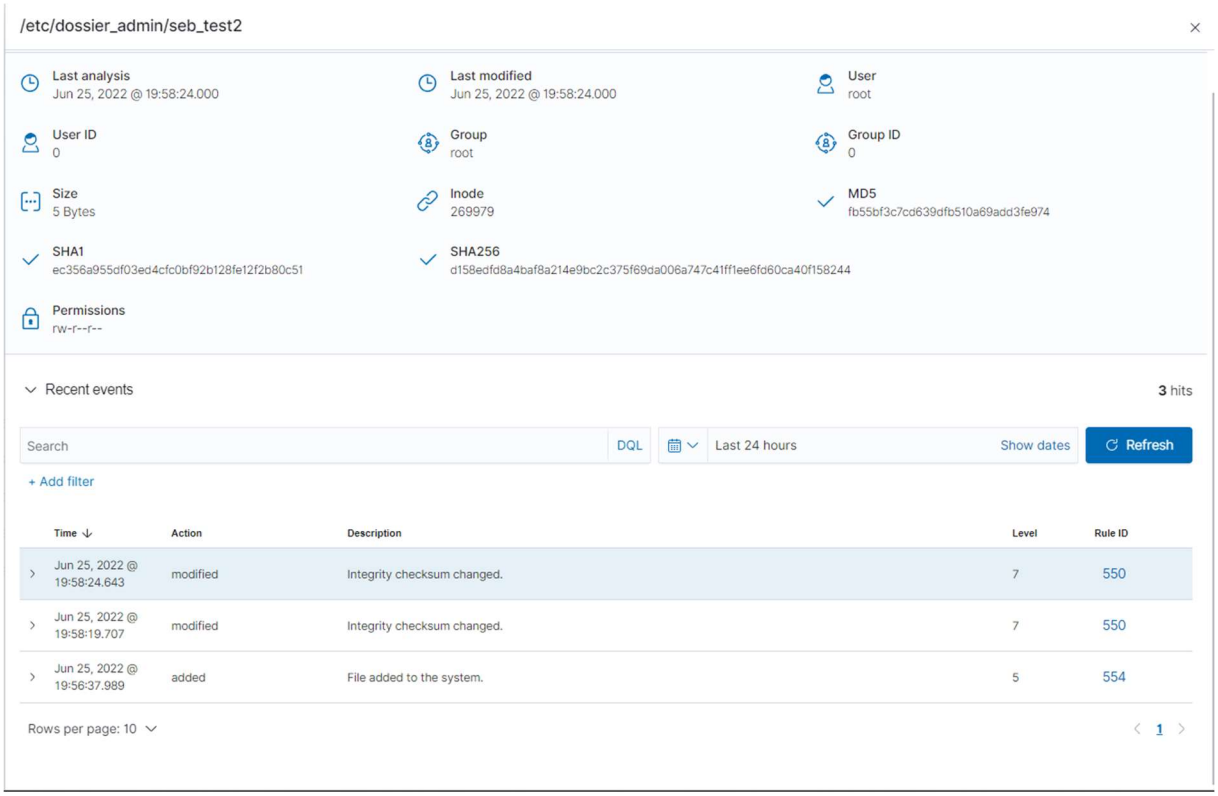
```
<agent_config>

  <!-- Shared agent configuration here -->

  <localfile>
     <location>/var/log/vsftpd.log</location>
     <log_format>syslog</log_format>
  </localfile>

  <syscheck>
     <alert_new_files>yes</alert_new_files>
     <directories check_all="yes" realtime="yes">/etc/dossier_admin</directories>
  </syscheck>

</agent_config>
```

Cette règle sera appliquée sur tous mes agents pour voir si un dossier sera modifié en temps réel.

Si le dossier est modifié WAZUH va réussir à le repérer grâce à un algorithme qui est le « md5 » c'est une chaine de caractères et de chiffres qui ne changera pas si le texte reste identique mais va changer s'il y a une différence.

```
root@wazuh:/tmp# echo "toto" > seb
root@wazuh:/tmp# md5sum seb
11a3e229084349bc25d97e29393ced1d  seb
root@wazuh:/tmp# echo "tata" > seb
root@wazuh:/tmp# md5sum seb
6ccef1b25ea58fb8be3ca1a1a744ea53  seb
```

Sur l'interface graphique de WAZUH dans le module integrity monitoring on peut voir si le dossier à subit des modifications.



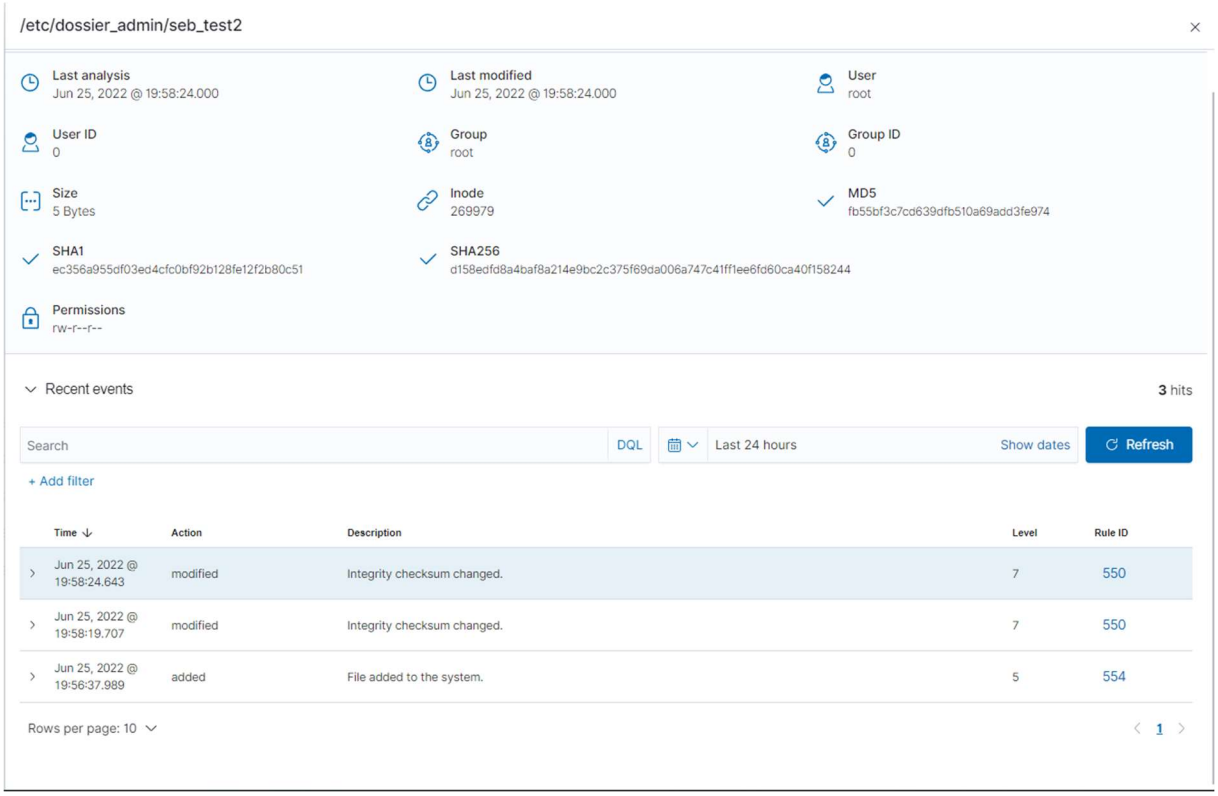
syscheck.md5_after	fb55bf3c7cd639dfb510a69add3fe974
syscheck.md5_before	11a3e229084349bc25d97e29393ced1d

Quand on regarde la chaine avant et après, elle a changé, cela veut dire qu'il y a eu une modification.

Voici la règle qui a permis à mon agent de remonter l'information :

```
<agent_config>

  <!-- Shared agent configuration here -->

  <localfile>
     <location>/var/log/vsftpd.log</location>
     <log_format>syslog</log_format>
  </localfile>

  <syscheck>
     <alert_new_files>yes</alert_new_files>
     <directories check_all="yes" realtime="yes">/etc/dossier_admin</directories>
  </syscheck>

</agent_config>
```

- Pour la troisième méthode nous allons imaginer que j'ai réussi à rentrer dans le serveur FTP et que j'ai des intentions de nuire en ajoutant une image vérolée. J'ajoute l'image sur le serveur FTP.



On peut voir sur la dernière ligne dans l'interface graphique de WAZUH, dans le module integrity monitoring, qu'une image a été détectée sur le serveur FTP

Voici la règle qui m'a permis de pouvoir remonter l'information :

```
<rule id="11404" level="5">
  <if_sid>11400</if_sid>
  <match>OK UPLOAD: </match>
  <description>vsftpd: FTP server file upload.</description>
</rule>
```

On a augmenté le niveau de la gravité pour qu'elle puisse apparaitre, car ce dossier est sensible et on ne doit pas pouvoir mettre un dossier ou effacer un dossier sans avoir une alerte.

9. Défaut de la technologie SIEM

Le plus gros point négatif de cette technologie c'est qu'il y a beaucoup d'alertes et qu'elles n'ont pas les mêmes niveaux de danger. Pour pouvoir remédier à ce problème il faut une équipe dédiée qu'on appelle le SOC (Security Operation Center) qui soit consacrée à traiter les alertes.

10. Point fort de la technologie SIEM de WAZUH

Le plus gros point fort de cette technologie c'est que le logiciel est open source et qu'il ajoute régulièrement du nouveau contenu pour pouvoir repérer les hackeurs car les hackeurs cherchent toujours de nouvelles façons de trouver une brèche.

11. Améliorations possibles

Les améliorations possibles sont :

- Pouvoir ajouter une adresse mail aux alertes de haute gravité.

```
<ossec_config>
  <global>
    <jsonout_output>yes</jsonout_output>
    <alerts_log>yes</alerts_log>
    <logall>no</logall>
    <logall_json>no</logall_json>
    <email_notification>no</email_notification>
    <smtp_server>smtp.example.wazuh.com</smtp_server>
    <email_from>wazuh@example.wazuh.com</email_from>
    <email_to>recipient@example.wazuh.com</email_to>
    <email_maxperhour>12</email_maxperhour>
    <email_log_source>alerts.log</email_log_source>
    <agents_disconnection_time>10m</agents_disconnection_time>
    <agents_disconnection_alert_time>0</agents_disconnection_alert_time>
  </global>

  <alerts>
    <log_alert_level>3</log_alert_level>
    <email_alert_level>12</email_alert_level>
  </alerts>
```

- Pouvoir ajouter des règles personnalisables.
- Intégrer une application tierce comme Teams ou Discord pour recevoir les alertes.