Surveillance des processus en cours d'exécution

Dans ce cas d'utilisation, on configure les agents **Wazuh** pour exécuter des commandes localement afin de surveiller les processus en cours sur les points de terminaison **Windows** et **Linux**. Ensuite, on crée des règles personnalisées sur le serveur **Wazuh** pour générer des alertes lorsqu'un processus particulier est en cours d'exécution ou non.

I - Surveillance des processus en cours sur le point de terminaison Windows

On configure le module de commande pour surveiller les processus en cours d'exécution sur le point de terminaison Windows et alerter si le processus **notepad.exe** est en cours d'exécution.

Configuration du point de terminaison Windows :

1. On crée un script batch nommé **tasklist.bat** dans le répertoire **C:** du point de terminaison **Windows** et on y ajoute le contenu suivant :

```
@Echo Off
setlocal enableDelayedExpansion

for /f "delims=" %%a in ('powershell -command "& tasklist"') do (
   echo tasklist: %%a
)

exit /b
```

2. On ajoute la configuration suivante au fichier de l'agent **Wazuh** :

```
<ossec_config>
  <wodle name="command">
    <disabled>no</disabled>
    <tag>tasklist</tag>
    <command>PowerShell.exe C:\tasklist.bat</command>
    <interval>2m</interval>
    <run_on_start>yes</run_on_start>
    <timeout>10</timeout>
  </wodle>
</ossec_config>
```

3. On redémarre l'agent **Wazuh** pour appliquer les modifications, en utilisant **PowerShell** avec les privilèges d'administrateur :

```
Restart-Service -Name WazuhSvc
```

II – Configuration du serveur Wazuh

1. On ajoute le décodeur suivant au fichier **/var/ossec/etc/decoders/local_decoder.xml** :

```
<decoder name="tasklist">
  <prematch>^tasklist: </prematch>
</decoder>
```

2. On ajoute la règle ci-dessous au fichier **/var/ossec/etc/rules/local_rules.xml** pour générer une alerte lorsque le processus notepad.exe est en cours d'exécution :

```
<group name="process_monitor">
  <rule id="100010" level="6">
    <decoded_as>tasklist</decoded_as>
    <regex type="pcre2">(?i)notepad.exe</regex>
    <description>Notepad.exe is running.</description>
  </rule>
</group>
```

Surveillance des processus en cours d'exécution

3. On redémarre le gestionnaire **Wazuh** pour appliquer les modifications :

```
sudo systemctl restart wazuh-manager
```

III - Surveillance des processus en cours sur le point de terminaison Linux

Les points de terminaison **Linux** exécutent un certain nombre de processus par défaut, notamment le démon **Cron**. Pour ce point de terminaison, on surveille les processus en cours d'exécution à l'aide du module **Logcollector** et on alerte si le **/usr/sbin/cron** processus ne s'exécute pas comme prévu. On utilise la commande **ps** pour obtenir l'état des processus actifs sur le point de terminaison **Linux**.

Configuration point de terminaison Linux :

1. On configure ce point de terminaison pour surveiller l'état des processus en cours d'exécution toutes les deux minutes et transmettre sa sortie au serveur **Wazuh** pour analyse.

2. On ajoute la configuration du module **Logcollector** ci-dessous au **/var/ossec/etc/ossec.conf** fichier de l'agent Wazuh :

```
<ossec_config>
  <localfile>
    <log_format>full_command</log_format>
    <command>ps -auxw</command>
    <frequency>120</frequency>
  </localfile>
</ossec_config>
```

II – Configuration du serveur Wazuh

1. On ajoute les règles suivantes au **/var/ossec/etc/rules/local_rules.xml fichier**. Les règles génèrent une alerte lorsque le **/usr/sbin/cron** processus ne s'exécute pas comme prévu :

```
<group name="process_monitor">
  <rule id="100012" level="6">
    <if_sid>530</if_sid>
    <match>^ossec: output: 'ps -auxw'</match>
    <description>Cron process not running.</description>
  </rule>

  <rule id="100013" level="0">
    <if_sid>100012</if_sid>
    <match>/usr/sbin/cron</match>
    <description>Cron process is running as expected.</description>
  </rule>
</group>
```

2. On redémarre le gestionnaire **Wazuh** pour appliquer les modifications :

```
sudo systemctl restart wazuh-manager
```

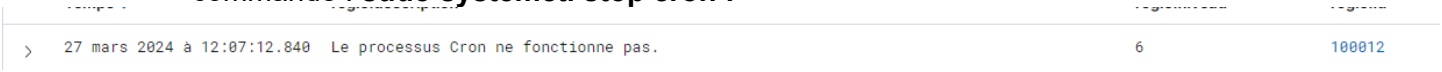
Surveillance des processus en cours d'exécution

Tester la configuration

- Pour déclencher une alerte, on lance l'application correspondante sur chaque point de terminaison puis on accède à l'onglet « **Modules** » > « **Security Events** » sur le tableau de bord **Wazuh** pour visualiser les alertes générées.

    - Par exemple le **Bloc-Notes** de **Windows** :

    - Par exemple en arrêtant le processus **Cron** sur le point de terminaison **Linux** avec la commande : **sudo systemctl stop cron :**



Autres configurations :

Dans ce cas d'utilisation, on montre comment détecter les processus en cours d'exécution avec le module **Wazuh SCA**.

Netcat est un utilitaire réseau polyvalent qui utilise **TCP** et **UDP** pour la transmission de données sur un réseau IP. Il permet d'ouvrir des connexions, d'envoyer des paquets ou encore d'écouter sur les ports **TCP** et **UDP**. Malheureusement, des acteurs malveillants exploitent souvent **Netcat** à des fins néfastes, comme la mise en place d'accès clandestins.

I – Configuration du point de terminaison Linux

1. Créer un nouveau répertoire pour enregistrer les fichiers de stratégie personnalisés :

```
mkdir /var/ossec/etc/custom-sca-files/
```

2. On crée un nouveau fichier de stratégie **SCA /var/ossec/etc/custom-sca-files/processcheck.yml** et ajouter le contenu suivant :

```
policy:
  id: "process_check"
  file: "processcheck.yml"
  name: "SCA use case to detect running processes"
  description: "Guidance for checking running processes on Linux endpoints."
  references:
    - https://documentation.wazuh.com/current/user-manual/capabilities/sec-config-assessment/index.html
    - https://documentation.wazuh.com/current/user-manual/capabilities/sec-config-assessment/creating-custom-
policies.html

requirements:
  title: "Check that the SSH service and password-related files are present on the system"
  description: "Requirements for running the SCA scan against the Unix based systems policy."
  condition: any
  rules:
    - "f:$sshd_file"
    - "f:/etc/passwd"
    - "f:/etc/shadow"

variables:
  $sshd_file: /etc/ssh/sshd_config

checks:
  - id: 10003
    title: "Ensure that netcat is not running on your endpoint"
    description: "Netcat is running on your endpoint."
    rationale: "Threat actors can use netcat to open ports on your endpoints or to connect to remote servers."
    remediation: "Kill the netcat process if confirmed to be malicious after further investigation."
    condition: none
    rules:
      - 'p:nc'
      - 'p:netcat'
```

3. On modifie la propriété du fichier pour que **Wazuh** y ait accès :

```
chown wazuh:wazuh /var/ossec/etc/custom-sca-files/processcheck.yml
```

4. On active le fichier de stratégie en ajoutant les lignes suivantes au bloc **<ossec_config>** du fichier de configuration de l'agent **Wazuh** à l'emplacement **/var/ossec/etc/ossec.conf** :

```
<sca>
  <policies>
    <policy enabled="yes">/var/ossec/etc/custom-sca-files/processcheck.yml</policy>
  </policies>
</sca>
```

5. On installe **netcat** si ce n'est pas déjà fait sur le point de terminaison :

```
apt install netcat
```

6. On exécute **netcat** sur un nouveau terminal et laisser l'écouteur en cours d'exécution :

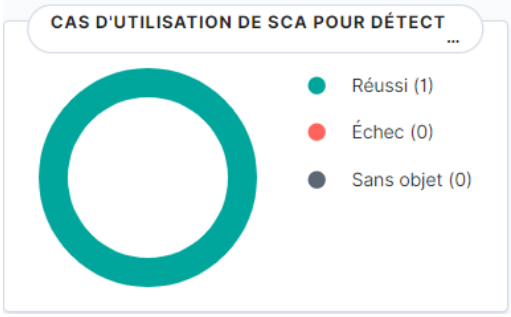
```
netcat -l 4444
```

7. On redémarre l'agent **Wazuh** pour appliquer les modifications et exécuter la nouvelle vérification **SCA** :

```
systemctl restart wazuh-agent
```

Test :

Sur le tableau de bord **Wazuh**, on accède à l'onglet **SCA** et on sélectionne le point de terminaison Linux pour afficher les résultats de la vérification **SCA** personnalisée.



Processus Netcat en cours d'exécution

Surveillance des processus en cours d'exécution

II – Configuration du point de terminaison Windows

1. On exécute **CMD** en tant qu'administrateur et créer un nouveau répertoire pour enregistrer les fichiers de stratégie personnalisés :

```
mkdir "C:\Program Files (x86)\ossec-agent\custom-sca-files"
```

2. On ouvre le **Bloc-notes** en tant qu'administrateur, on crée un nouveau fichier de stratégie **SCA** avec le contenu suivant et l'enregistrer sous : **C:\Program Files (x86)\ossec-agent\custom-sca-files\processcheck.yml**

```
policy:
  id: "process_check"
  file: "processcheck.yml"
  name: "SCA use case to detect running processes"
  description: "Guidance for checking running PowerShell processes on Windows 10 endpoints."
  references:
    - https://documentation.wazuh.com/current/user-manual/capabilities/sec-config-assessment/index.html
    - https://documentation.wazuh.com/current/user-manual/capabilities/sec-config-assessment/creating-custom-policies.html

requirements:
  title: "Check that the Windows platform is Windows 10"
  description: "Requirements to check if it's a Windows 10 (or Windows 11) machine"
  condition: all
  rules:
    - 'r:HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion -> ProductName -> r:^Windows 10'

checks:
  - id: 10004
    title: "Ensure PowerShell is not running on the endpoint"
    description: "PowerShell is running on the endpoint."
      rationale: "PowerShell should be used by only the system administrators. Threat actors can leverage PowerShell for living-off-the-land attacks."
    remediation: "Disable PowerShell for non-admins."
    condition: none
    rules:
      - 'p:powershell.exe'
```

3. On active le fichier de stratégie en ajoutant les lignes suivantes au bloc **<ossec_config>** du fichier de configuration de l'agent **Wazuh** à l'emplacement **C:\Program Files (x86)\ossec-agent\ossec.conf** :

```
<sca>
  <policies>
    <policy enabled="yes">C:\Program Files (x86)\ossec-agent\custom-sca-files\processcheck.yml</policy>
  </policies>
</sca>
```

4. On ouvre une deuxième invite de commande en tant qu'administrateur et on exécute la commande suivante pour générer un processus **PowerShell** caché :
   Il s'agit d'un processus **Powershell** factice qui dort pendant 300 secondes (5 minutes), suffisamment de temps pour que vous puissiez redémarrer l'agent **Wazuh** afin que l'analyse **SCA** s'exécute.

```
powershell -windowstyle hidden -command Start-Sleep -Seconds 300
```

Note : L'invite de commande se ferme après avoir exécuté cette commande et **PowerShell** s'exécute en arrière-plan.

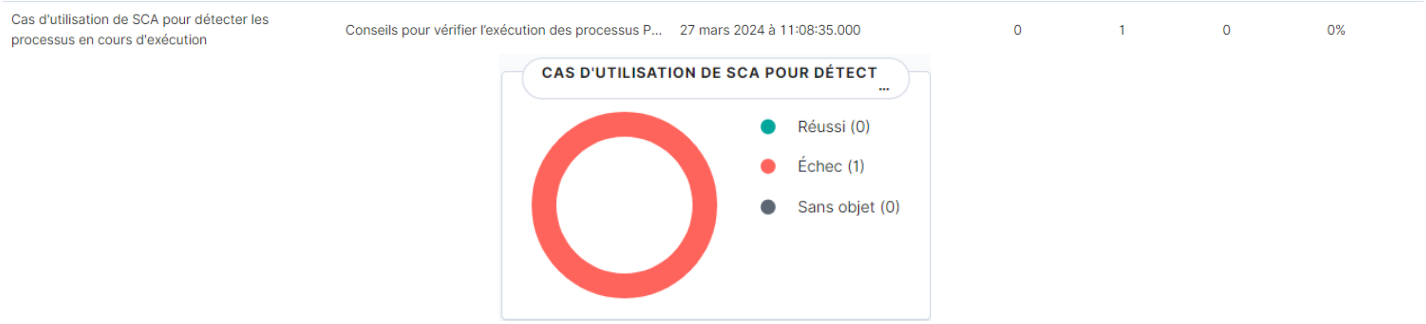
Surveillance des processus en cours d'exécution

5. On exécute les commandes suivantes sur **CMD** en tant qu'administrateur pour redémarrer l'agent **Wazuh** :

```
NET STOP WazuhSvc
NET START WazuhSvc
```

Test :

Sur le tableau de bord **Wazuh**, on accède à l'onglet **SCA** et sélectionner le point de terminaison **Windows** pour afficher les résultats de la vérification **SCA** personnalisée.



PowerShell non exécuté