

***Ecole Nationale des Sciences Appliquées de Sciences Appliquées de Khouribga***

***Filière : Ingénierie des Réseaux Intelligents & Cybersécurité***

**Rapport du Mini Projet:**

# **Évaluation Efficace du SOC avec MITRE ATT&CK**

**Réalisé par :**

- **EL GHAI T Imane**
- **AMAOUI Imane**

**Encadré par :**

- **Mr MALEH Yassine**

---

# *Table des matières*

---

## Table des matières

### Introduction

|   |    |
|---|----|
| <b>Etape 1 : Analyse de l'Étude de Cas et Identification des TTPs :</b>   | 4  |
| <b>Etape 2 : Définition des SOC Use Cases Basés sur les TTPs :</b>  | 6  |
| <b>Etape 3: Création des Règles de Détection dans Wazuh</b>   | 9  |
| Installation Wazuh sur la VM :  | 9  |
| Access the Wazuh web interface with <a href="https://192.168.8.107">https://192.168.8.107</a> and your credentials: | 10 |
| Ajout de l'agent : Windows 10 : IP :192.168.8.106   | 11 |
| Configuration des règles de Détection en ciblant le comportement des TTPs identifiés :                              | 12 |
| <b>Use Case 1 :Détection Webshell :</b>   | 13 |
| <b>Use Case : Détection PowerShell :</b>  | 16 |
| <b>Etape 4 : Simulation des Use Cases et Validation des Alertes :</b>   | 18 |
| <b>1-Use Case 1 : Détection WebShell :</b>  | 18 |
| 1ere Attack :   | 18 |
| 2eme Attack :   | 19 |
| <b>Use Case 2 : Emulation d'attaque Power Shell :</b>   | 20 |
| (Obfuscation et évaison)  | 20 |
| Command and Scripting Interpreter.  | 20 |

---

## *Introduction*

---

Le présent rapport est réalisé dans le cadre du mini-projet intitulé "**SOC Use Cases Basés sur MITRE ATT&CK**", issu du cours d'introduction à la cybersécurité. L'objectif principal de ce projet est de concevoir et d'implémenter des cas d'usage pour la détection d'activités suspectes dans un environnement SOC (Security Operations Center), en se basant sur les tactiques, techniques et procédures (TTPs) identifiées lors de l'opération "**Soft Cell**".

Ce projet s'appuie sur l'utilisation de la plateforme Wazuh pour configurer des règles de détection permettant de surveiller des comportements malveillants spécifiques, alignés avec les techniques du **Framework MITRE ATT&CK**. À travers une démarche structurée, nous explorons des phases telles que l'accès initial, la persistance, le déplacement latéral et le dumping d'identifiants. Les objectifs sont d'améliorer la sécurité proactive d'un SOC et de renforcer la capacité de réponse en temps réel face aux cybermenaces.

Les résultats escomptés incluent une meilleure compréhension des **TTPs**, une configuration efficace des règles de détection, ainsi que des recommandations concrètes pour l'optimisation des capacités de détection du SOC.

## Etape 1 : Analyse de l'Étude de Cas et Identification des TTPs :

D'après l'analyse du Rapport de la campagne « Operation Soft Cell », on a observé des Tactiques et techniques avec le Framework MITRE ATT&CK :

### THE ATTACK

The initial indicator that 'sparked our imagination' was malicious activity performed by `w3wp.exe`, an IIS process, which was eventually classified as a `webshell` activity. By investigating the `webshell`

(which was later classified as the 'China Chopper' [1, 2] `webshell`), we were able to unravel several attack phases and TTPs used by the attackers.

The attackers leveraged the `webshell` to run reconnaissance commands and credential-stealing activities.

One of the reconnaissance actions was to run a modified `NirSoft NetBIOS scanner` in order to identify available `NetBIOS` name servers locally or over the network.



The credential-stealing tool was a modified `Mimikatz` version, which, when executed, only dumps NTLM hashes (note that it does not require any command line arguments). We renamed this sample `maybemimi.exe`.

Reverse engineering shows the string similarity between `maybemimi.exe` and `Mimikatz`, as shown in Figures 5 and 6.

emerging, as well as reconnaissance and credential-stealing activity. In addition, we uncovered the usage of `PiVv` (Poison Ivy) by the attackers.

### PiVv

Poison Ivy (or `PiVv` for short) is a RAT that is associated with Chinese threat actors. `PiVv` is a powerful, well-featured RAT that allows an attacker to take total control of a machine. Among its containment actions that, later on, were put in place on an ad-hoc basis. We kept track in our research of the different TTPs of the threat actor, and there were no new indicators until the end of 2018. During that time, we detected interesting compressing activity carried out by the attackers in order to exfiltrate data. The attackers used `WinRAR`, which they downloaded from their C2 server, to compress the data they wanted to steal. By leveraging (another) `webshell`, and a renamed `cmd.exe` version, the attackers executed reconnaissance commands, collected data, dropped tools like `portqry.exe` and `hTran`, and performed lateral movement using `net use` and `wmic.exe`.

### hTran

One of the tools that the attackers deployed in order to exfiltrate the data from network segments that were not connected to the Internet was a customized version of `hTran`, a 'connection bounce' tool which allows the attacker to redirect ports and connections between different networks. `hTran`'s code is publicly available on [GitHub](#) [4].

| Tactics           | ID     | Techniques   | ID   | Description (objectif des attaquants )  |
|-------------------|--------|--|--|---|
| Persistence       | T1505  | -Webshells<br><br>-Remote Access Trojan  | T1505.003<br><br>T1219                           | Les attaquants ont utilisé des webshells, comme "China Chopper," pour obtenir un accès initial aux systèmes compromis.<br><br>-Poison Ivy (PiVv) a été installé en tant que RAT (Remote Access Trojan) persistant avec des fonctionnalités avancées   |
| Credential Access | TA0006 | -OS Credential Dumping<br>LSASS Memory   | T1003  | Utilisation de versions modifiées d'outils comme Mimikatz (nommé "maybemimi.exe") pour récupérer des hash NTLM et d'autres informations d'identifiants.   |
| Reconnaissance    | TA0043 | - Network Service Scanning<br><br>- System Network Connections Discovery<br><br>- System Network Configuration Discovery<br><br>-Remote System Discovery | T1046<br><br>T1049<br><br>T1016<br><br>T1018     | Les attaquants ont utilisé des outils comme un scanner NetBIOS de NirSoft pour explorer le réseau, identifier les serveurs de noms et obtenir une cartographie des ressources locales et distantes.   |
| Execution         | TA0002 | -Command and Scripting Interpreter   | T1059  | Des webshells ont été utilisés pour exécuter des commandes directement sur les serveurs compromis.<br>- Windows Command Shell<br><<<cmd.exe   |
| Defense Evasion   | TA0005 | - DLL via Side-Loading.<br><br>- Signed Binary Proxy Execution<br><br>-Obfuscation<br><br>-Masquerading  | T1574.002<br><br>T1218<br><br>T1027<br><br>T1036 | -Les attaquants ont utilisé des logiciels signés et légitimes (comme des outils Samsung) pour charger des DLL malveillantes et éviter la détection.<br>-les attaquants ont modifié les noms et les chaînes de caractères des outils pour échapper à la détection. Mimikatz vers "maybemimi.exe" |
| Exfiltration      | TA0010 | - Archive Collected Data<br><br>-Non-Standard Port   | T1560.001<br><br>T1048.003                       | - Les attaquants ont utilisé WinRAR pour compresser les fichiers avant de les exfiltrer.<br>-Utilisation de l'outil hTran pour rediriger les connexions et exfiltrer  |

|                     |        |  |                            |  |
|---------------------|--------|--|----------------------------|--|
|                     |        | -Exfiltration Over C2 Channel  | T1041                      | des données depuis des segments de réseau isolés.  |
| Command and Control |        | -Application Layer Protocol<br>-Dynamic Resolution: Domain Generation Algorithms | T1071.001<br>T1568.002     |  |
| Collection          | TA0009 | -Input Capture: Keylogging<br><br>-Data from Information Repositories CDR        | T1056.001<br><br>T1213.001 | Keylogging and various other surveillance features<br><br>-One of the most valuable pieces of data that telcos hold are CDRs – call detail records. CDRs are basically a large subset of metadata containing all the details about calls |
| Lateral Movement    | TA0008 | - Remote Services: SMB/Windows Admin Shares                                      | T1021.002                  | CDRversion, the attackers executed reconnaissance commands, collected data, dropped tools like portqry.exe and hTran, and performed lateral movement using net use and wmic.exe  |

## Etape 2 : Définition des SOC Use Cases Basés sur les TTPs :

Intégrer MITRE ATT&CK dans un **SOC (Security Operations Center)** pour améliorer la détection et la réponse aux incidents de sécurité.

Pour chaque TTP identifié, définir un use case de détection dans le contexte SOC et l'objectif de détection et les déclencheurs spécifiques à surveiller.

| Use Case                  | TTP(Technique)       | MITRE ATT&CK ID | Objectif de détection   |
|---------------------------|----------------------|-----------------|---|
| Détecter des webshells    | Webshell             | T1505.003       | Détecter l'activité anormale associée à des webshells : commandes exécutées via des processus inattendus (ex : w3wp.exe pour IIS).  |
| Accès Persistant avec RAT | Remote Access Trojan | T1219           | Identifier l'utilisation continue des outils RAT comme Poison Ivy pour maintenir un contrôle à long terme sur les systèmes compromis, en surveillant les communications régulières avec les serveurs de commande et de contrôle (C2). |

|   |  |           |  |
|---|--|-----------|--|
| Détecter l'utilisation de credential dumping ou d'outils connus, même s'ils sont modifiés ou masqués. | OS Credential Dumping                            | T1003     | Détecter l'activité des attaquants pour extraire les identifiants et les hash de mots de passe stockés dans le système d'exploitation. Les outils comme <b>Mimikatz vers maybemimi.exe</b>         |
| Identifier et bloquer les scans réseau effectués par des attaquants                                   | Network Service Scanning                         | T1046     |  |
| Découverte des connexions réseau système  | System Network Connections Discovery             | T1016     | Identifier les requêtes inhabituelles ou répétées sur des segments réseau spécifiques, notamment via des outils comme <b>NetBIOS</b> scanner ou commandes PowerShell.                              |
| Découverte de la configuration réseau système   | - System Network Configuration Discovery         | T1016     | . Identifier l'exécution de commandes ou d'outils visant à collecter des informations sur la configuration réseau, comme l'utilisation de <b>ipconfig, route</b> ou d'autres utilitaires systèmes. |
| Commandes et scripts interprétés  | Command and Scripting Interpreter                | T1059     | Détecter les commandes encodées ou masquées (ex. : <b>Encoded Command dans PowerShell</b> ), utilisées pour échapper aux systèmes de détection classiques  |
| -Découverte de systèmes distants  | Remote System Discovery                          | T1018     | Identifier les tentatives de reconnaissance sur des hôtes distants en surveillant des outils tels <b>que net use et wmic.exe.</b>  |
| Détecter et bloquer les RATs  | Remote Access Trojan (RAT)                       | T1219     | Détecter les comportements malveillants liés à l'exécution ou au contrôle à distance.  |
| -DLL via Side-Loading   | DLL Side-Loading                                 | T1574.002 | Identifier les tentatives de chargement de DLL non autorisées ou malveillantes via des applications légitimes.   |
| Archivage des données dans des formats standards (.zip .rar.tar ....)                                 | Archive Collected Data                           | T1560.001 | Détection des fichiers d'archives inhabituels  |
| -Utilisation de protocoles d'application pour exfiltration  | -Obfuscation<br>Application Layer Protocol       | T1027     | Identifier les outils ou fichiers malveillants déguisés par modification de chaînes ou d'attributs (ex. changement des noms de fichiers).  |
| Résolution dynamique avec des algorithmes de génération de domaine                                    | Dynamic Resolution: Domain Generation Algorithms | T1568.002 | Détecter les requêtes DNS vers des domaines générés dynamiquement (par exemple, via des algorithmes de génération de domaine - <b>DGA</b> )  |
| Exfiltration de données via C2  | Exfiltration Over C2 Channel                     | T1041     | Détecter les activités de transfert de données volumineuses ou continues vers des domaines ou IPs suspects liés  |

|  |                                     |       |   |
|--|-------------------------------------|-------|---|
|  |                                     |       | aux C2, en utilisant des outils comme hTran.  |
| Exécution par proxy avec des binaires signés | Signed Binary<br>Proxy<br>Exécution | T1218 | Détecter l'utilisation de binaires signés légitimes (comme RunHelp.exe) pour charger des bibliothèques malveillantes (DLL side-loading) |



## Etape 3: Création des Règles de Détection dans Wazuh

### Installation Wazuh sur la VM :

```
$ curl -sO https://packages.wazuh.com/4.9/wazuh-install.sh && sudo bash ./wazuh-i  
ninstall.sh -a
```

```
10.0.2.15      38          96 21    0.57    0.62    0.64 dimr    cluster_manager,data,ingest,remote_cluster_client *  
node-1  
root@ubuntu:/home/imane/Desktop# apt-get -y install wazuh-manager  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
wazuh-manager is already the newest version (4.9.2-1).  
0 upgraded, 0 newly installed, 0 to remove and 341 not upgraded.  
root@ubuntu:/home/imane/Desktop# systemctl daemon-reload  
root@ubuntu:/home/imane/Desktop# systemctl enable wazuh-manager  
Created symlink /etc/systemd/system/multi-user.target.wants/wazuh-manager.service -> /lib/systemd/system/wazuh-manager.service.  
root@ubuntu:/home/imane/Desktop# systemctl start wazuh-manager  
root@ubuntu:/home/imane/Desktop# systemctl status wazuh-manager  
* wazuh-manager.service - Wazuh manager  
   Loaded: loaded (/lib/systemd/system/wazuh-manager.service; enabled; 5  
   Active: active (running) since Thu 2024-11-14 20:28:50 +01; 57min ago  
     Tasks: 101 (limit: 4753)  
    Memory: 288.6M  
       CPU: 2min 59.787s  
    CGroup: /system.slice/wazuh-manager.service  
            └─63583 /var/ossec/framework/python/bin/python3 /var/ossec/a  
            └─63623 /var/ossec/bin/wazuh-authd  
            └─63639 /var/ossec/bin/wazuh-db  
            └─63663 /var/ossec/bin/wazuh-execd  
            └─63667 /var/ossec/framework/python/bin/python3 /var/ossec/a
```

```
INFO: --- Summary ---
```

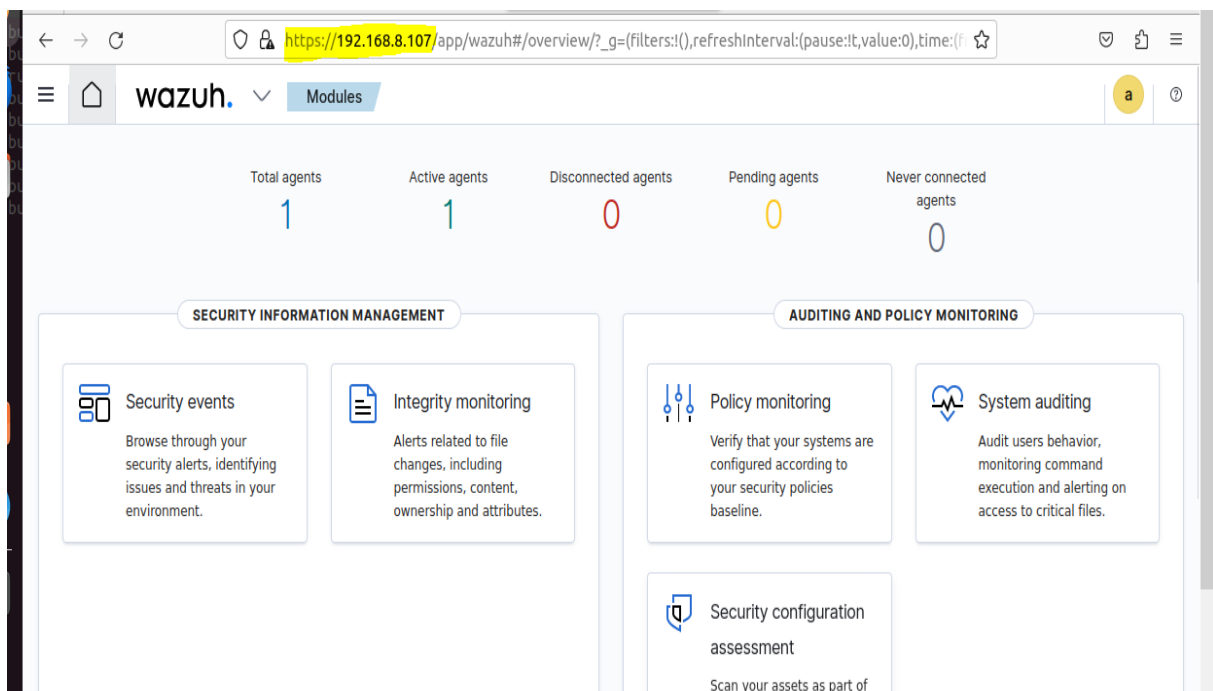
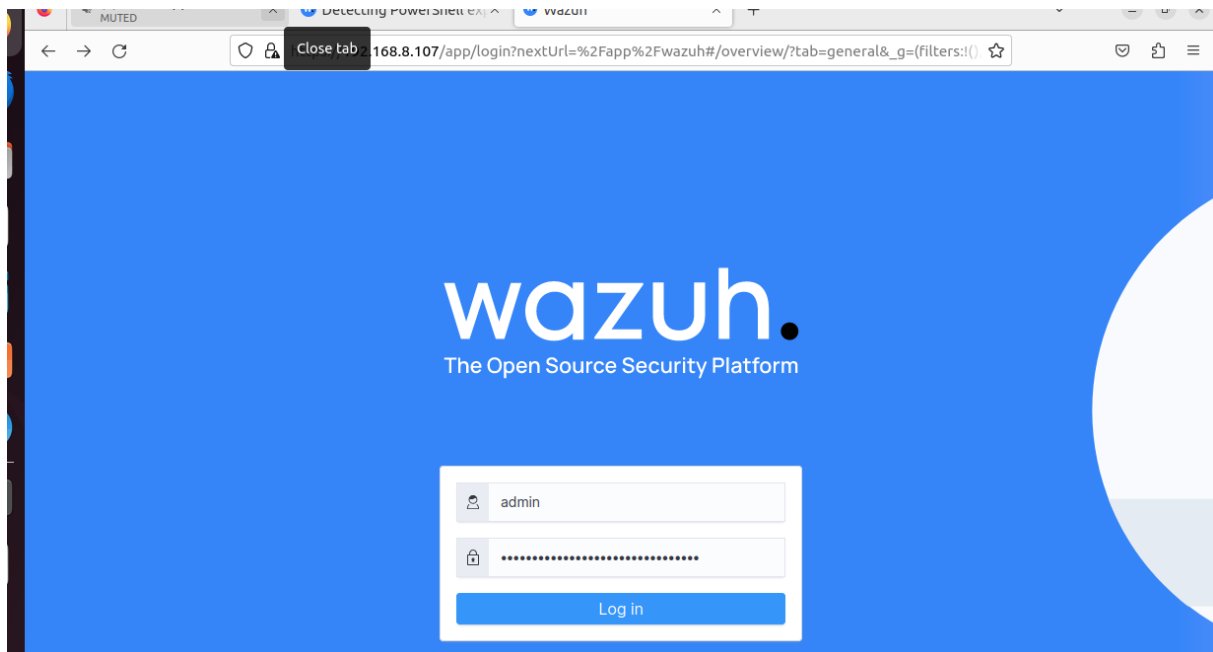
```
INFO: You can access the web interface https://<wazuh-dashboard-ip>
```

```
    User: admin
```

```
    Password: <ADMIN_PASSWORD>
```

```
INFO: Installation finished.
```

Access the Wazuh web interface with <https://192.168.8.107> and your credentials:



## Ajout de l'agent : Windows 10 : IP :192.168.8.106

Refresh

### Deploy new agent

**1 Select the package to download and install on your system:**

**LINUX**

☐ RPM amd64 ☐ RPM aarch64

☐ DEB amd64 ☐ DEB aarch64

**WINDOWS**

☐ MSI 32/64 bits

**macOS**

☐ Intel

☐ Apple silicon

[For additional systems and architectures, please check our documentation.](#)

**2 Server address:**

```
PS C:\Windows\system32> NET START WazuhSvc
Le service demandé a déjà été démarré.

Vous obtiendrez une aide supplémentaire en entrant NET HELPMSG 2182.

PS C:\Windows\system32> Invoke-WebRequest -Uri https://packages.wazuh.com/4.x/windows/wazuh-agent-4.7.5-1.msi -OutFile $(
(env.tmp)\wazuh-agent; msexec.exe /i $(env.tmp)\wazuh-agent /q WAZUH_MANAGER='192.168.8.107' WAZUH_AGENT_GROUP='default'
' WAZUH_REGISTRATION_SERVER='192.168.8.107'
PS C:\Windows\system32> Invoke-WebRequest -Uri https://packages.wazuh.com/4.x/windows/wazuh-agent-4.7.5-1.msi -OutFile $(
(env.tmp)\wazuh-agent; msexec.exe /i $(env.tmp)\wazuh-agent /q WAZUH_MANAGER='192.168.8.107' WAZUH_AGENT_GROUP='default'
' WAZUH_REGISTRATION_SERVER='192.168.8.107'
PS C:\Windows\system32> NET START WazuhSvc
Le service Wazuh démarre.
Le service Wazuh a démarré.

PS C:\Windows\system32>
```

**STATUS**

● Active (1)

● Disconnected (0)

● Pending (0)

● Never connected (0)

**DETAILS**

| Active | Disconnected | Pending | Never connected |
|--------|--------------|---------|-----------------|
| 1      | 0            | 0       | 0               |

Agents coverage  
**100.00%**

Last registered agent  
**windows**

Most active agent  
**windows**

**EVOLUTION**

Last 24 hours

No results found

**Agents (1)** Deploy new agent Refresh Export formatted

WQL Refresh

| ID  | Name    | IP address    | Group(s) | Operating system                          | Cluster node | Version | Status | Actions |
|-----|---------|---------------|----------|---|--------------|---------|--------|---------|
| 001 | windows | 192.168.8.106 | default  | Microsoft Windows 10 Home 10.0.19045.3803 | node01       | v4.7.5  | active |         |

Rows per page: 10

## Configuration des règles de Détection en ciblant le comportement des TTPs identifiés :

Chaque Règle de Détection comprend :

- ID de la règle
- Niveau de criticité (de 0 jusqu'à 10)
- Logique de détection
- La tactique et la technique dont elle appartient

Création d'un fichier **/var/ossec/etc/rules/myrules.xml** dont lequel on ajoutera les règles de détections en se basant sur les uses cases des TTPs identifiées :

```
root@ubuuu: /var/ossec/etc/rules
GNU nano 6.2 myrules.xml *
<!--RAT.XML-->
<rule id="100002" level="10">
  <field name="network.dst_ip">C2 Server IP</field>
  <description>RAT: Communication régulière avec un serveur C2</description>
  <group>rat, persistence</group>
  <mitre>
    <id>T1219</id>
  </mitre>
</rule>

<!--OS Credential Dumping-->
<rule id="100003" level="10">
  <field name="win.eventdata.processName">maybemini.exe</field>
  <description>Credential Dumping: Détection d'outils tels que Mimikatz</description>
  <group>credential_dumping, credential_access</group>
  <mitre>
    <id>T1003</id>
  </mitre>
</rule>

<!--Network service Scanning-->
<rule id="100004" level="8">
  <field name="network.src_ip">Attacker IP</field>
  <description>Scan réseau: Activité de reconnaissance réseau détectée</description>
  <group>network_scanning, discovery</group>
  <mitre>
    <id>T1046</id>
  </mitre>
</rule>
```

```
</mitre>
</rule>

<!--decouverte des cnx reseaux systeme-->
<rule id="100005" level="9">
  <field name="win.eventdata.commandline">*NetBIOS*</field>
  <description>Découverte réseau: Activité liée à NetBIOS détectée</description>
  <group>network_connections_discovery, discovery</group>
  <mitre>
    <id>T1016</id>
  </mitre>
</rule>

<!--commandes et scripts interprété-->
<rule id="100007" level="10">
  <field name="win.eventdata.commandline">EncodedCommand</field>
  <description>Commande encodée: Détection de scripts PowerShell masqués</description>
  <group>command_execution, execution</group>
  <mitre>
    <id>T1059</id>
  </mitre>
</rule>

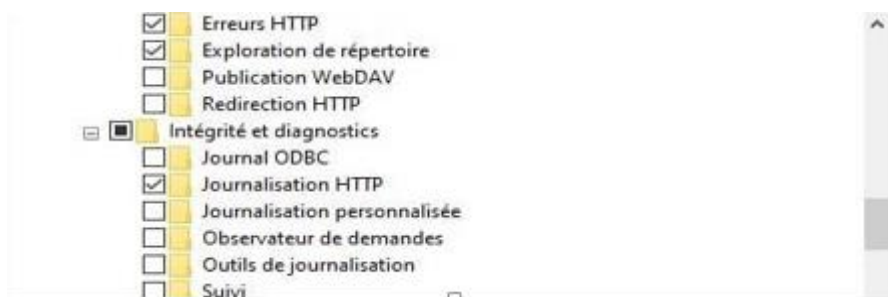
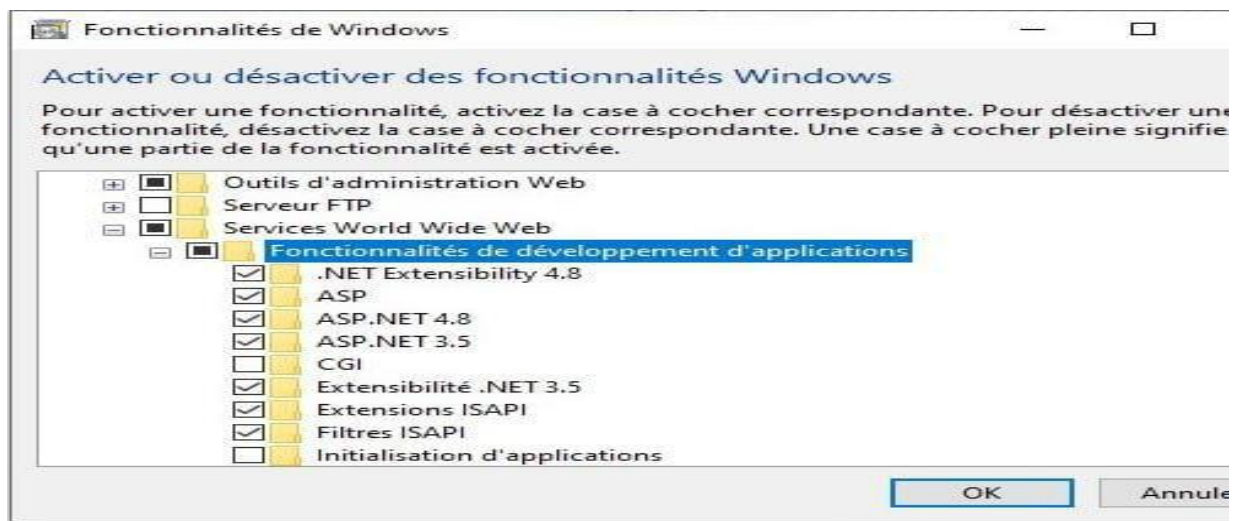
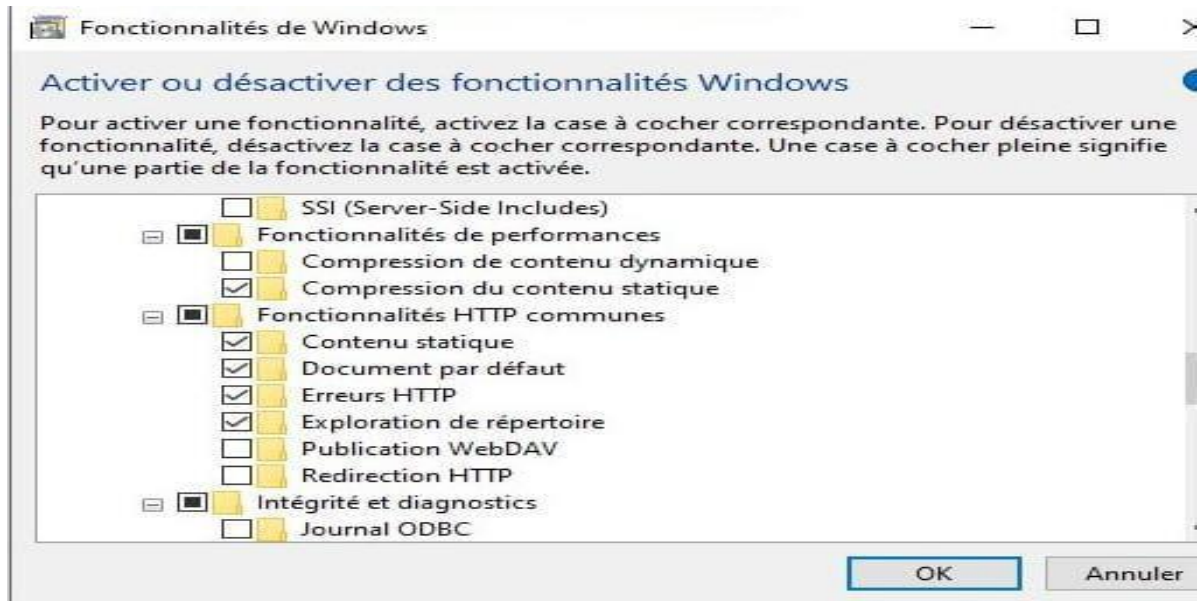
<!--DLL via Side-Loading-->
<rule id="100008" level="10">
  <field name="win.eventdata.processName">RunHelp.exe</field>
  <description>DLL Side-Loading: Utilisation d'une application légitime pour charger une DLL malveillante</description>
  <group>dll_side_loading, defense_evasion</group>
  <mitre>
    <id>T1574.002</id>
  </mitre>
</rule>
```

Et Redémarrer le service Wazuh-manager :

```
root@ubuuu:/var/ossec/etc/rules# sudo systemctl restart wazuh-manager
root@ubuuu:/var/ossec/etc/rules# ls
```

## Use Case 1 :Détection Webshell :

Configuration de **IIS web server** on the Windows Agent :



Ajouter la règle : Dans le fichier webshell\_rules.xml

```
root@ubuuu: /var/ossec/etc/rules
GNU nano 6.2 webshell_rules.xml
<group name="linux, webshell, windows,">
<!-- This rule detects file creation. -->
<rule id="100500" level="12">
<if_sid>554</if_sid>
<field name="file" type="pcr2">(?!).php$|.phtml$|.php3$|.php4$|.php5$|.phps$|.phar$|.asp$|.aspx$|.jsp$|.cshtml$|.vbhtml$</field>
<description>[File creation]: Possible web shell scripting file ($(file)) created</description>
<mitre>
<id>T1105</id>
<id>T1505</id>
</mitre>
</rule>

<!-- This rule detects file modification. -->
<rule id="100501" level="12">
<if_sid>550</if_sid>
<field name="file" type="pcr2">(?!).php$|.phtml$|.php3$|.php4$|.php5$|.phps$|.phar$|.asp$|.aspx$|.jsp$|.cshtml$|.vbhtml$</field>
<description>[File modification]: Possible web shell content added in ($(file))</description>
<mitre>
<id>T1105</id>
<id>T1505</id>
</mitre>
</rule>

<!-- This rule detects files modified with PHP web shell signatures. -->
<rule id="100502" level="15">
<if_sid>100501</if_sid>
<field name="changed_content" type="pcr2">(?!).passthru|exec|eval|shell_exec|assert|str_rot13|system|phpinfo|base64_decode|chmo
<description>[File Modification]: File $(file) contains a web shell</description>
<mitre>
<id>T1105</id>
<id>T1505.003</id>
</mitre>
</rule>
```

Et Redémarrer le service Wazuh Manager

```
systemctl restart wazuh-manager
```

## Installation de Sysmon installer et configurer le fichier sysmonconfig.xml :

File Explorer: Ce PC > Téléchargements > Sysmon

| Nom              | Modifié le         | Type                | Taille   |
|------------------|--------------------|---------------------|----------|
| Eula.txt         | 11/30/2024 9:31 PM | Document texte      | 8 Ki     |
| Sysmon.exe       | 11/30/2024 9:31 PM | Application         | 8,282 Ki |
| Sysmon64.exe     | 11/30/2024 9:31 PM | Application         | 4,457 Ki |
| Sysmon64a.exe    | 11/30/2024 9:31 PM | Application         | 4,877 Ki |
| sysmonconfig.xml | 11/30/2024 9:46 PM | Microsoft Edge H... | 203 Ki   |

```
sysmonconfig.xml - Bloc-notes
Fichier Edition Format Affichage Aide
<!-- NOTICE : This is a balanced generated output of Sysmon-modu
<!-- due to the balanced nature of this configuration there wi
<!-- for more information go to https://github.com/olafhartong,
<Sysmon schemaversion="4.60">
<HashAlgorithms>*</HashAlgorithms>
<!-- This now also determines the file names of the files preserved (String) -->
<CheckRevocation>False</CheckRevocation>
<!-- Setting this to true might impact performance -->
<DnsLookup>False</DnsLookup>
<!-- Disables lookup behavior, default is True (Boolean) -->
<ArchiveDirectory>Sysmon</ArchiveDirectory>
<!-- Sets the name of the directory in the C:\ root where preserved files will be save
<EventFiltering>
<!-- Event ID 1 == Process Creation - Includes -->
<RuleGroup groupRelation="or">
<ProcessCreate onmatch="include">
<ParentImage name="technique_id=T1546.008,technique_name=Accessibility Features" condi
<ParentImage name="technique_id=T1546.008,technique_name=Accessibility Features" condi
<ParentImage name="technique_id=T1546.008,technique_name=Accessibility Features" condi
```



```
> .\Sysmon64.exe -accepteula -i sysmonconfig.xml
```

```
PS C:\> cd 'C:\Users\imanewin\Downloads'
PS C:\Users\imanewin\Downloads> cd 'Sysmon'
PS C:\Users\imanewin\Downloads\Sysmon> .\Sysmon64.exe -accepteula -i sysmonconfig.xml

System Monitor v15.15 - System activity monitor
By Mark Russinovich and Thomas Garnier
Copyright (C) 2014-2024 Microsoft Corporation
Using libxml2. libxml2 is Copyright (C) 1998-2012 Daniel Veillard. All Rights Reserved.
Sysinternals - www.sysinternals.com

Loading configuration file with schema version 4.60
Sysmon schema version: 4.90
Configuration file validated.
The service Sysmon64 is already registered. Uninstall Sysmon before reinstalling.

PS C:\Users\imanewin\Downloads\Sysmon>
```

```
PS C:\Windows\system32> cd 'C:\Program Files (x86)\ossec-agent'
PS C:\Program Files (x86)\ossec-agent> notepad ossec.conf
PS C:\Program Files (x86)\ossec-agent>
```

```

Fichier Edition Format Affichage Aide
<log_format>eventchannel</log_format>
</localfile>

<localfile>
  <location>active-response\active-responses.log</location>
  <log_format>syslog</log_format>
</localfile>

<localfile>
  <location>Microsoft-Windows-Sysmon/Operational</location>
  <log_format>eventchannel</log_format>
</localfile>

<!-- Policy monitoring -->
<rootcheck>
  <disabled>no</disabled>
  <windows_apps>./shared/win_applications_rcl.txt</windows_apps>
  <windows_malware>./shared/win_malware_rcl.txt</windows_malware>
</rootcheck>

<!-- Security Configuration Assessment -->
<sca>
  <enabled>yes</enabled>
  <scan on start>yes</scan on start>

```

Ln 1, Col 1      100%      Windows (CRLF)      UTF-8

```
Restart-Service -Name wazuh
```

- Ajouter dans le fichier webshell\_rules.xml, les règles qui permettent de détection les commandes de web Shell d'exécution et d'établissement de connexion :

```
<!-- Windows Rules. -->
<group name="sysmon, webshell, windows,">
  <!-- This rule detects web shell command execution. -->
  <rule id="100530" level="12">
    <if_sid>61603</if_sid>
    <field name="win.eventdata.parentImage" type="pcr2">(?!w3wp\).exe</field>
    <field name="win.eventdata.parentUser" type="pcr2">(?!IIS\sAPPPOOL\\DefaultAppPool</field>
    <description>[Command execution (${win.eventdata.commandLine})]: Possible web shell attack detected</description>
    <mitre>
      <id>T1505.003</id>
      <id>T1059.004</id>
    </mitre>
  </rule>

  <!-- This rule detects web shell network connections. -->
  <rule id="100531" level="12">
    <if_sid>61605</if_sid>
    <field name="win.eventdata.image" type="pcr2">(?!w3wp\).exe</field>
    <field name="win.eventdata.user" type="pcr2">(?!IIS\sAPPPOOL\\DefaultAppPool</field>
    <description>[Network connection]: Possible web shell attempting network connection on source port: ${win.eventdata.sourcePort}>
    <mitre>
      <id>TA0011</id>
      <id>T1049</id>
      <id>T1505.003</id>
    </mitre>
  </rule>
</group>
```

```
<!-- Linux Rules. -->
<group name="auditd, linux, webshell,">
  <!-- This rule detects web shell command execution. -->
  <rule id="100520" level="12">
    <if_sid>80700</if_sid>
    <field name="audit.key">webshell_command_exec</field>
    <description>[Command execution (${audit.exe})]: Possible web shell attack detected</description>
    <mitre>
      <id>T1505.003</id>
      <id>T1059.004</id>
    </mitre>
  </rule>

  <!-- This rule detects web shell network connections. -->
  <rule id="100521" level="12">
    <if_sid>80700</if_sid>
    <field name="audit.key">webshell_net_connect</field>
    <description>[Network connection via ${audit.exe}]: Possible web shell attack detected</description>
    <mitre>
      <id>TA0011</id>
      <id>T1049</id>
      <id>T1505.003</id>
    </mitre>
  </rule>
</group>
```

Redémarrer le service wazuh-manager :

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

### Use Case : Détection PowerShell :

- Activation de la journalisation PowerShell pour avoir une journalisation détaillée dans PowerShell :
- Ouvrir Powershell en tant qu'administrateur :



```

Administrateur : Windows PowerShell
Testez le nouveau système multiplateforme PowerShell https://aka.ms/pscore6

PS C:\Windows\system32> function Enable-PSLogging {
>> # Define registry paths for ScriptBlockLogging and ModuleLogging
>> $scriptBlockPath = 'HKLM:\Software\Policies\Microsoft\Windows\PowerShell\ScriptBlockLogging'
>> $moduleLoggingPath = 'HKLM:\Software\Policies\Microsoft\Windows\PowerShell\ModuleLogging'
>>
>> # Enable Script Block Logging
>> if (-not (Test-Path $scriptBlockPath)) {
>>     $null = New-Item $scriptBlockPath -Force
>> }
>> Set-ItemProperty -Path $scriptBlockPath -Name EnableScriptBlockLogging -Value 1
>>
>> # Enable Module Logging
>> if (-not (Test-Path $moduleLoggingPath)) {
>>     $null = New-Item $moduleLoggingPath -Force
>> }
>> Set-ItemProperty -Path $moduleLoggingPath -Name EnableModuleLogging -Value 1
>>
>> # Specify modules to log - set to all (*) for comprehensive logging
>> $moduleNames = @('*') # To specify individual modules, replace * with module names in the array
>> New-ItemProperty -Path $moduleLoggingPath -Name ModuleNames -PropertyType MultiString -Value $moduleNames -Force
>>
>> Write-Output "Script Block Logging and Module Logging have been enabled."
>> }
PS C:\Windows\system32>
PS C:\Windows\system32> Enable-PSLogging

ModuleNames : {*}
PSPPath      : Microsoft.PowerShell.Core\Registry::HKEY_LOCAL_MACHINE\Software\Policies\Microsoft\Windows\PowerShell\Mo
               duleLogging
PSParentPath : Microsoft.PowerShell.Core\Registry::HKEY_LOCAL_MACHINE\Software\Policies\Microsoft\Windows\PowerShell
PSChildName  : ModuleLogging
PSDrive      : HKLM
PSProvider   : Microsoft.PowerShell.Core\Registry

Script Block Logging and Module Logging have been enabled.

PS C:\Windows\system32>

```

Le transfert des journaux PowerShell au serveur Wazuh pour les analyser ce fait comme suit :

```

Script Block Logging and Module Logging have been enabled.

PS C:\Windows\system32> cd 'C:\Program Files (x86)\ossec-agent'
PS C:\Program Files (x86)\ossec-agent> notepad ossec.conf
PS C:\Program Files (x86)\ossec-agent> cd ..
PS C:\Program Files (x86)\> cd ..
PS C:\> Restart-Service -Name wazuh
PS C:\>

```

Ajouter les règles de détection dans le fichier `/var/ossec/etc/rules/local_rules.xml` :

```
GNU nano 6.2 local.rules.xml *
</group>
<group name="windows,powershell">

<rule id="100201" level="8">
  <if_sid>60009</if_sid>
  <field name="win.eventdata.payload" type="pcr2">(?!)(CommandInvocation</field>
  <field name="win.system.message" type="pcr2">(?!)(EncodedCommand|FromBase64String|EncodedArguments|-e\b|-enco\b|-en\b</field>
  <description>Encoded command executed via PowerShell.</description>
  <mitre>
    <id>T1059.001</id>
    <id>T1562.001</id>
  </mitre>
</rule>

<rule id="100202" level="4">
  <if_sid>60009</if_sid>
  <field name="win.system.message" type="pcr2">(?!)(blocked by your antivirus software</field>
  <description>Windows Security blocked malicious command executed via PowerShell.</description>
  <mitre>
    <id>T1059.001</id>
  </mitre>
</rule>

<rule id="100203" level="10">
  <if_sid>60009</if_sid>
  <field name="win.eventdata.payload" type="pcr2">(?!)(CommandInvocation</field>
  <field name="win.system.message" type="pcr2">(?!)(Add-Persistence|Find-AVSignature|Get-GPPAutologon|Get-GPPPassword|Get-HttpSta
  <description>Risky CMDLet executed. Possible malicious activity detected.</description>
  <mitre>
    <id>T1059.001</id>
  </mitre>
</rule>
```

Et Redémarrer le service Wazuh-manager :

```
systemctl restart wazuh-manager
```

## Etape 4 : Simulation des Use Cases et Validation des Alertes :

### 1-Use Case 1 : Détection WebShell :

Lancer le Power Shell au tant qu'Administrateur et exécuter la commande suivante :

1ere Attack :

- Création d'un fichier dans le répertoire du serveur web : webshell-script.aspx  
C:\inetpub\wwwroot

```
Administrateur : Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. Tous droits réservés.

Testez le nouveau système multiplateforme PowerShell https://aka.ms/pscore6

PS C:\Windows\system32> New-Item -Path 'C:\inetpub\wwwroot\webshell-script.aspx' -ItemType File

Répertoire : C:\inetpub\wwwroot

Mode                LastWriteTime         Length Name
----                -
-a----          11/30/2024 10:20 PM              0 webshell-script.aspx

PS C:\Windows\system32>
```

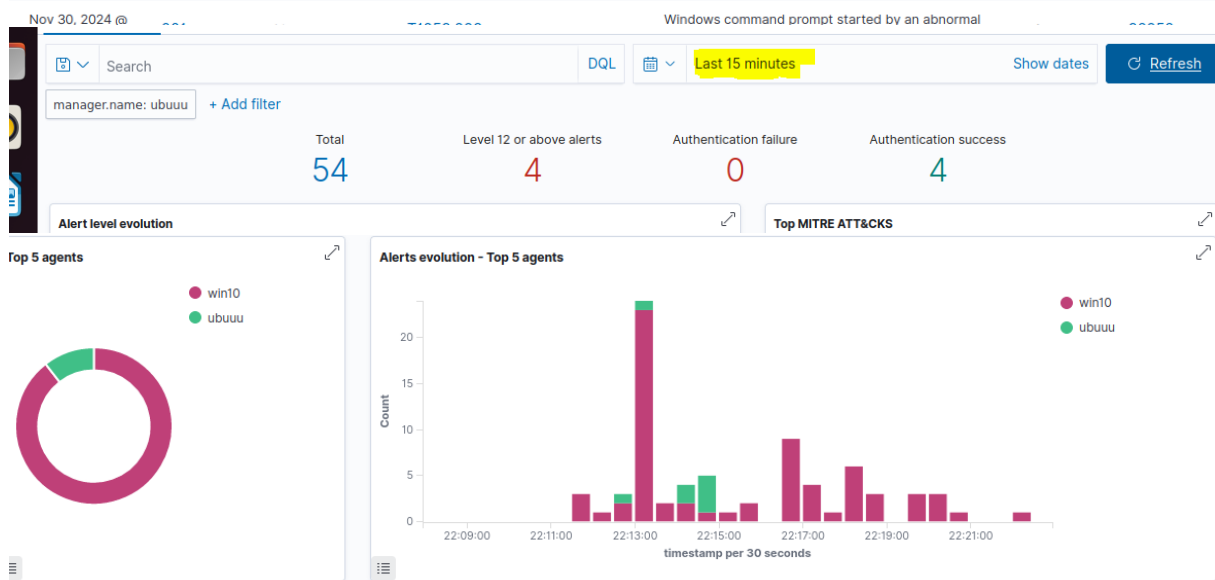
Accéder à l'interface du server Wazuh :

Security Events > Alertes : (pour les dernières 15 minutes)

| Time ↓                        | Agent | Agent name | Technique(s)           | Tactic(s)  | Description   | Level | Rule ID |
|-------------------------------|-------|------------|------------------------|--|---|-------|---------|
| > Nov 30, 2024 @ 22:24:58.031 | 001   | win10      | T1574.001<br>T1574.002 | Persistence, Privilege Escalation, Defense Evasion | Possible DLL search order hijack by C:\Windows\SoftwareDistribution\Download\debd60358786516fd3e6fc91795ca227\Metadata\dpx.dll created in Windows root folder         | 6     | 92219   |
| > Nov 30, 2024 @ 22:24:57.984 | 001   | win10      | T1574.001<br>T1574.002 | Persistence, Privilege Escalation, Defense Evasion | Possible DLL search order hijack by C:\Windows\SoftwareDistribution\Download\debd60358786516fd3e6fc91795ca227\Metadata\UpdateAgent.dll created in Windows root folder | 6     | 92219   |
| > Nov 30, 2024 @ 22:22:20.128 | 001   | win10      |                        |  | CIS Microsoft Windows 10 Enterprise Benchmark v1.12.0: Ensure 'Network access: Allow anonymous SID/Name translation' is set to 'Disabled'.                            | 3     | 19009   |
| > Nov 30, 2024 @ 22:20:51.655 | 001   | win10      | T1105<br>T1505         | Command and Control, Persistence                   | [File creation]: Possible web shell scripting file (c:\inetpub\wwwroot\webshell-script.aspx) created  | 12    | 100500  |
| > Nov 30, 2024 @ 22:20:24.677 | 001   | win10      | T1105                  | Command and Control                                | Executable file dropped in folder commonly used by malware  | 15    | 92213   |
| > Nov 30, 2024 @ 22:20:13.281 | 001   | win10      | T1053.005              | Execution, Persistence, Privilege Escalation       | Process loaded taskschd.dll module. May be used to create delayed malware execution   | 4     | 92154   |
| > Nov 30, 2024 @ 22:20:12.340 | 001   | win10      |                        |  | Software protection service scheduled successfully.   | 3     | 60642   |

#### Security Alerts

| Time ↓                      | Agent | Agent name | Technique(s)       | Tactic(s)                        | Description  | Level | Rule ID |
|-----------------------------|-------|------------|--------------------|----------------------------------|--|-------|---------|
| Nov 30, 2024 @ 22:31:02.216 | 001   | win10      | T1087<br>T1059.003 | Discovery, Execution             | Suspicious Windows cmd shell execution   | 3     | 92032   |
| Nov 30, 2024 @ 22:31:01.402 | 001   | win10      | T1105<br>T1505     | Command and Control, Persistence | [File modification]: Possible web shell content added in c:\inetpub\wwwroot\webshell-script.aspx | 12    | 100501  |



#### 2eme Attack :

- Modification du fichier crée en ajoutant un texte aléatoire 'hello World'

```
PS C:\Windows\system32> Set-Content -Path 'C:\inetpub\wwwroot\webshell-script.aspx' -Value 'Hello world!'
PS C:\Windows\system32> Invoke-WebRequest -OutFile 'C:\Users\Public\Downloads\webshell-script.aspx' -Uri https://psidave
```

Accéder à l'interface Wazuh pour voir l'alerte :

|                            |     |       |           |                                  |  |    |        |
|----------------------------|-----|-------|-----------|----------------------------------|--|----|--------|
| 16:31:14.578               | 001 | win10 | T1059.003 | Execution                        | Windows command prompt started by an abnormal process  | 4  | 92052  |
| Dec 1, 2024 @ 16:31:14.575 | 001 | win10 | T1105     | Command and Control, Persistence | [File modification]: Possible web shell content added in c:\inetpub\wwwroot\webshell-script.aspx | 12 | 100501 |
| Dec 1, 2024 @ 16:31:14.250 | 001 | win10 | T1505     |                                  |  |    |        |

Rows per page: 10

Use Case 2 : Emulation d'attaque Power Shell :

On fait une attaque qui permet de masquer l'intention des scripts et échapper la détection.

(Obfuscation et évasion)

Ouvrir PowerShell en tant qu'Administrateur :

```

+ FullyQualifiedErrorId : WebException

PS C:\> powershell.exe -EncodedCommand "VwByAGKADAB1AC0ATwB1AHQAcAB1AHQAIAAIAEgAZQBzAGwAbwAgAFcAbwByAGwAZAAIAA=="
Hello World
PS C:\>
  
```

Accéder à l'interface wazuh pour voir l'alerte :

| Time ↓                     | Agent | Agent name | Technique(s)   | Tactic(s)                             | Description  | Level | Rule ID |
|----------------------------|-------|------------|----------------|---------------------------------------|--|-------|---------|
| Dec 1, 2024 @ 16:30:58.545 | 001   | win10      | T1105<br>T1505 | Command and Control, Persistence      | [File creation]: Possible web shell scripting file (c:\inetpub\wwwroot\webshell-script.aspx) created                         | 12    | 100500  |
| Dec 1, 2024 @ 16:30:45.408 | 001   | win10      | T1055          | Defense Evasion, Privilege Escalation | Explorer process was accessed by C:\Users\lmanewin\AppData\Local\Microsoft\OneDrive\OneDrive.exe, possible process injection | 12    | 92910   |
| Dec 1, 2024 @ 16:27:30.502 | 001   | win10      | T1105          | Command and Control                   | Executable file dropped in folder commonly used by malware   | 15    | 92213   |

Rows per page: 10

Command and Scripting Interpreter.

On fait une attaque qui permet d'exécuter des commandes malveillantes :

```

Hello World
PS C:\> echo "whoami" > "C:\Program Files (x86)\ossec-agent\active-response\bin\test.ps1"
PS C:\> Powershell -ExecutionPolicy bypass -File "C:\Program Files (x86)\ossec-agent\active-response\bin\test.ps1"
win10\lmanewin
PS C:\>
  
```

Accéder à l'interface Wazuh pour voir l'alerte :

| Security Alerts              |       |            |              |                     |  |       |         |
|------------------------------|-------|------------|--------------|---------------------|--|-------|---------|
| Time ↓                       | Agent | Agent name | Technique(s) | Tactic(s)           | Description  | Level | Rule ID |
| > Dec 1, 2024 @ 19:30:50.358 | 001   | win10      | T1105        | Command and Control | Executable file dropped in folder commonly used by malware | 15    | 92213   |
| > Dec 1, 2024 @ 19:30:50.079 | 001   | win10      | T1059.001    | Execution           | PowerShell execution policy set to bypass.                 | 5     | 100205  |

---

## Conclusion

---

Ce mini-projet a permis d'explorer et de mettre en œuvre des cas d'usage de détection basés sur des TTPs issus de l'opération **Soft Cell**, en utilisant le Framework **MITRE ATT&CK**. À travers une approche méthodique, nous avons identifié des techniques spécifiques utilisées par des acteurs malveillants, défini des objectifs de détection clairs et configuré des règles adaptées dans un environnement SOC à l'aide de **Wazuh**.

Les principales réalisations incluent :

- L'identification et le mapping des TTPs avec les IDs **MITRE ATT&CK** appropriés.
- La définition et la configuration de **règles XML** permettant de surveiller et détecter des comportements suspects.
- La validation des règles par des simulations et l'ajustement des paramètres pour réduire les faux positifs.

Ces travaux ont permis d'améliorer la compréhension des mécanismes d'attaque sophistiqués et de renforcer la capacité de détection proactive du SOC. Cependant, il est important de souligner que la cybersécurité est une discipline dynamique. Les attaquants évoluent constamment, ce qui nécessite une mise à jour régulière des règles et des techniques de détection.

Enfin, ce projet met en évidence l'importance d'une approche structurée et basée sur des Framework reconnus comme MITRE ATT&CK pour construire une défense robuste. Les recommandations incluent l'intégration de sources de renseignement sur les menaces (Threat Intelligence) et la formation continue des équipes SOC pour maintenir une posture de sécurité optimisée face à des menaces en constante évolution.