

Titulación: CFGS en Administración de Sistemas Informáticos en Red

Proyecto: Monitorizando la red con Wazuh

Curso: 2023 / 2024

Alumno:

Tutor:

Índice

Estudio del problema y análisis del sistema	3
Introducción. Breve descripción del proyecto.	3
Finalidad. Qué queremos conseguir con el sistema a implementar.	3
Objetivos. Qué servicios ofrecerá el sistema una vez implementado.	4
Modelado de la solución	5
Recursos humanos	5
Recursos hardware	5
Recursos software	6
Planificación	9
Ejecución	10
Acceso panel control	10
Implementación cliente Windows y Ubuntu	11
Configuración	13
Firewall del servidor	13
Grupos	13
Vulnerabilidades	14
Virusotal	14
Correo	16
Script para autoborrado	19
Slack	23
Optimización	27
Fase de pruebas	34
Login incorrecto	34
Detección archivo malicioso y auto borrado con virustotal	34
Detección de fuerza bruta	35
Vulnerabilidades	36
Conclusiones finales	36
Grado de cumplimiento de los objetivos fijados.	36
Propuesta de modificaciones o ampliaciones futuras del sistema implementado.	36
Modulo Regulatory Compliance	36
Modulo Cloud security monitoring	37
Documentación final	38
Docker	38
Contenedor	38
Estructura	38
Manual de Instalación	39
Manual de uso	41
Bibliografía	44

Estudio del problema y análisis del sistema

La seguridad de la información es primordial para cualquier organización, especialmente ante el aumento de ciberataques. La complejidad de gestionar múltiples dispositivos, cuentas, etc incrementa el riesgo de accesos no autorizados a nuestros datos. Implementar Wazuh en la red nos proporciona una solución para la detección y respuesta a amenazas, mejorando la protección y el cumplimiento normativo.

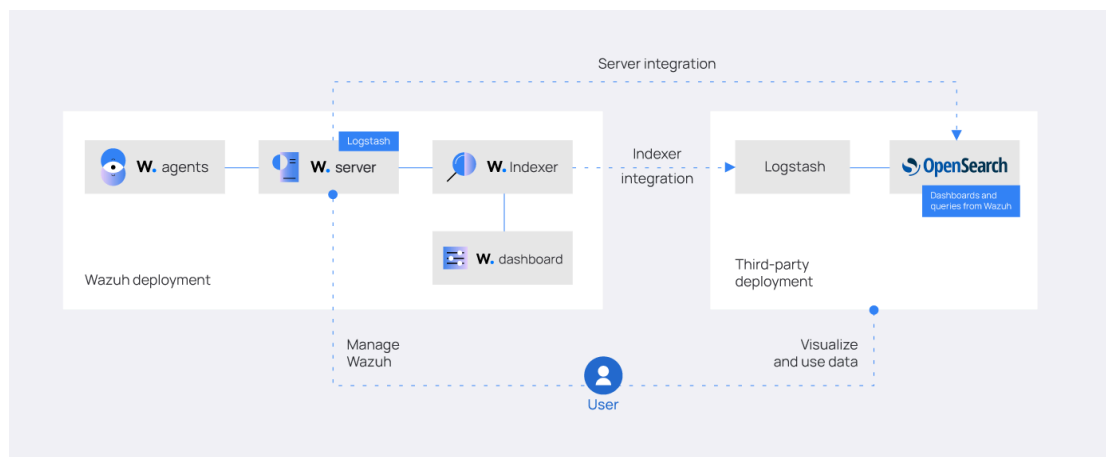
Introducción. Breve descripción del proyecto.

¿Qué es Wazuh?

Es una plataforma de seguridad de código abierto que sirve para monitorizar la red, detectar amenazas, y dar respuestas ante incidentes. Se utiliza para proteger la infraestructura informática de una organización.

Quiero implementar y poner en práctica un sistema de monitorización y análisis de todos los endpoints (clientes). Con ello, podremos ahorrarnos tanto tiempo como dinero. Para ello utilizaré Wazuh. Desde mi servidor podré monitorizar el servidor donde estará instalado Wazuh y podrá escanear todos los demás equipos donde instale el Wazuh cliente.

Un ejemplo visual del proyecto sería así:



Finalidad. Qué queremos conseguir con el sistema a implementar.

La finalidad de este proyecto es fortalecer la seguridad de nuestra empresa. Mediante la implementación de Wazuh:

Fortalecer seguridad:

Este sistema nos permitirá una detección temprana y precisa de amenazas, identificando y enumerando los registros anómalos que se nos podrían escapar.

Respuesta rápida y efectiva ante Incidentes:

Capacidad para detectar, responder y recuperarse de un ataque lo más rápido posible.

Análisis de la actividad en el sistema:

Proporciona una una visión detallada y contextualizada de la actividad que hay en los sistemas.

Objetivos. Qué servicios ofrecerá el sistema una vez implementado.

Recopilación continua de eventos de clientes

Estará constantemente guardando eventos sobre los endpoints de dentro de la red. Tendremos información detallada sobre actividades, registros, cambios en archivos confidenciales y mucho más.

Biblioteca de eventos forenses

Tendremos una biblioteca donde se guardarán todos los eventos organizados. Con esto nos permitirá la investigación y el análisis de eventos forenses.

Búsqueda de amenazas personalizables

Podremos realizar búsquedas personalizables para encontrar diferentes amenazas.

Almacenamiento central de eventos indefinidamente

Los eventos se pueden guardar de manera centralizada indefinidamente.

Instalar plugins para que Wazuh nos muestre información adicional

Estos agentes nos permitirán recabar mucha más información que la recopilación de por defecto.

Investigaciones rápidas

Nos facilitará realizar investigaciones en semanas.

Modelado de la solución

Recursos humanos

Un administrador de sistemas. Será el encargado de planificar y desplegar todo este sistema. El profesional, a parte de los conocimientos de informática, también deberá conocer cómo funciona la seguridad informática de la empresa.

El administrador de sistemas será el responsable, el solo, de desplegar todo el proyecto.

Escala de tiempo del proyecto			
Actividad	Duración (Días)	Horas	Coste (15€/h)
Investigación	3	10	150,00€
Planificación	3	10	150,00€
Implementación	25	100	1.500,00€
Optimización	15	60	900,00€
Fase de pruebas	10	40	600,00€
Documentación	7	25	375,00€
Evaluación final	2	7	105,00€
Entrega	1	3	45,00€
Presupuesto total del proyecto:			3.825,00€

Recursos hardware

Los recursos hardware que se utilizan para el desarrollo de este proyecto será:

- Linux server. Donde se aloja Wazuh server y se gestionará. Para trabajar en la configuración tanto de Wazuh como del servidor. Se recomienda acceder remotamente a él.

Requisitos mínimos del servidor:

- RAM: 4GB
 - Almacenamiento: 500GB
 - Sistema operativo Ubuntu Server 22.04 LTS
- Clientes. Se podrán añadir todos los clientes que queramos, el único requisito es que el Sistema Operativo sea compatible con Wazuh

Recursos hardware	Coste
Servidor Linux	650€

Recursos software

En recursos software no es necesario ningún gasto. Todas las tecnologías y aplicaciones son libres. En las aplicaciones que se necesitaba cuenta he usado la gratuita. Estos son los recursos que he utilizado:



Ubuntu Server 22.04 LTS es un sistema operativo basado en Linux desarrollado para producir un entorno robusto y seguro para servidores. Será donde alojaré todo mi proyecto.



Docker es un proyecto open-source que automatiza el despliegue de aplicaciones y virtualiza sistemas operativos dentro de contenedores. Levantaré la herramienta Wazuh en un docker.



Python es un lenguaje de programación de código abierto, simple y versátil. En mi caso lo voy a utilizar en la configuración de autoborrado de los archivos maliciosos.



Bash es un lenguaje de scripting de código abierto utilizado en sistemas Linux y UNIX. Se suele utilizar para automatizar tareas del sistema, escribir scripts... Lo utilizare para crear los scripts para añadir nuevos clientes.



Ansible es una herramienta que automatiza una gran cantidad de procesos informáticos. Se utiliza para instalar software, automatizar tareas, aplicar parches a los sistemas y muchos más. La herramienta utiliza el protocolo SSH para conectarse a los otros equipos y ejecutar las tareas. Para ello se crean los playbooks, que son archivos .yml o .yaml en donde se incluyen las tareas que queremos realizar. Será la encargada de la configuración e instalación de los nuevos clientes Linux.



Slack es una aplicación de mensajería para empresas que conecta a las personas con la información que necesitan. Lo uniré a mi Wazuh para que cuando llegue un tipo de alerta en específico me llegue un mensaje en slack.



La herramienta de VirusTotal es una herramienta multiplataforma que permite examinar archivos o URL de páginas web, analizándolas con más de 90 antivirus online en busca de software malintencionado oculto en ellos. Lo configuraré para que examine unas rutas específicas de mis clientes.



GitHub es una plataforma online de desarrollo de software que se usa para almacenar, supervisar y trabajar con proyectos de software. Facilita el intercambio de archivos de código y trabajar en proyectos colaborativos de código abierto. Lo utilizare para alojar mis scripts.

Planificación

Investigación

El apartado de investigación será recopilar toda la información que necesito para realizar el trabajo.

Planificación

La fase de planificación será preparar detalladamente la ejecución del proyecto. Se establecerán los objetivos a alcanzar y determinará los recursos que necesito desarrollar el proyecto.

Implementación de Wazuh

Durante la fase de implementación, se empezará a construir el sistema. Se desplegará toda la infraestructura necesaria, instalando tanto servidor como clientes. Se implementarán scripts y herramientas para automatizar tareas de administración y seguridad. Se implementarán también otras herramientas ajenas a Wazuh para aumentar la productividad de la herramienta.

Optimización

El objetivo de la optimización será mejorar la eficiencia y el rendimiento de Wazuh. Se realizarán ajustes y mejoras en las configuraciones para optimizar los tiempos de ejecución.

Fase de pruebas

Durante la fase de pruebas aseguraremos que la herramienta funciona y cumple con todos los requerimientos. Se realizarán pruebas adicionales para verificar todas las configuraciones. Se evaluará el rendimiento sobre situaciones reales de uso.

Documentación

Se creará toda la documentación detallada sobre el sistema y su implementación. Se crearán guías y manuales para otros usuarios.

Evaluación Final

Se revisará y evaluará el proyecto en todas las partes para su correcto funcionamiento.

Entrega

Se preparará y realizará una presentación del proyecto con los aspectos más importantes y los resultados obtenidos.

Diagrama de Gantt	Marzo	Abril	Mayo	Junio
<i>Investigación</i>				
<i>Planificación</i>				
<i>Implementación</i>				
<i>Optimización</i>				
<i>Fase de pruebas</i>				
<i>Documentación</i>				
<i>Evaluación final</i>				
<i>Entrega</i>				

Ejecución

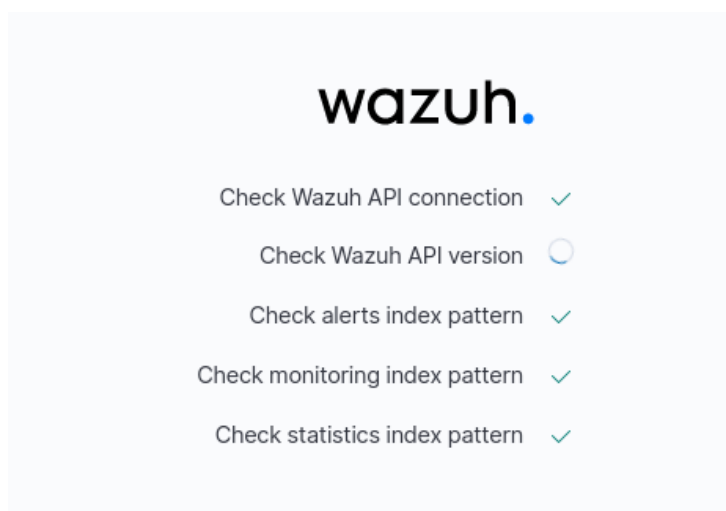
Acceso panel control

Para acceder al panel de control (dashboard) de Wazuh, que previamente habremos levantado en docker (ver [manual de instalación](#)), debemos ir a otro equipo que esté en la misma red que el servidor. Y en el navegador ponemos la IP del servidor.

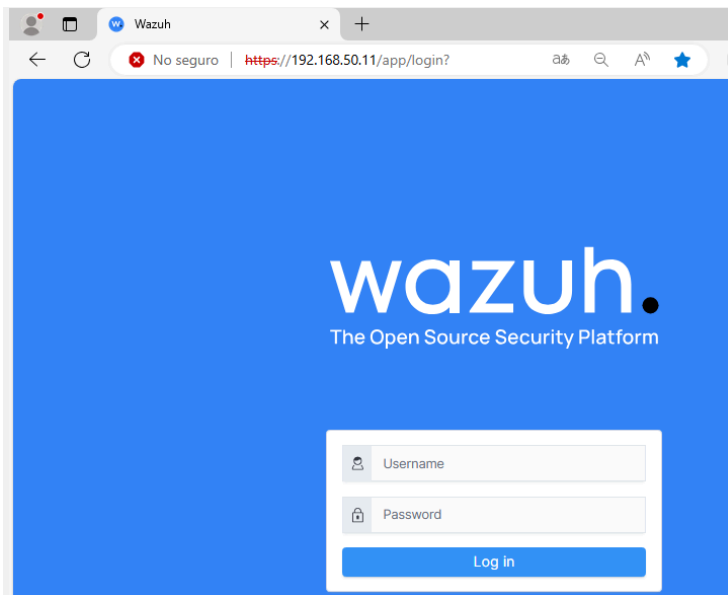
Las credenciales por defecto de Wazuh-Dashboard son: admin/SecretPassword

La contraseña del API es: MyS3cr37P450r.*-

La primera vez que accedemos se quedará bastante tiempo en esta pantalla ya que tiene que configurar todo por primera vez.



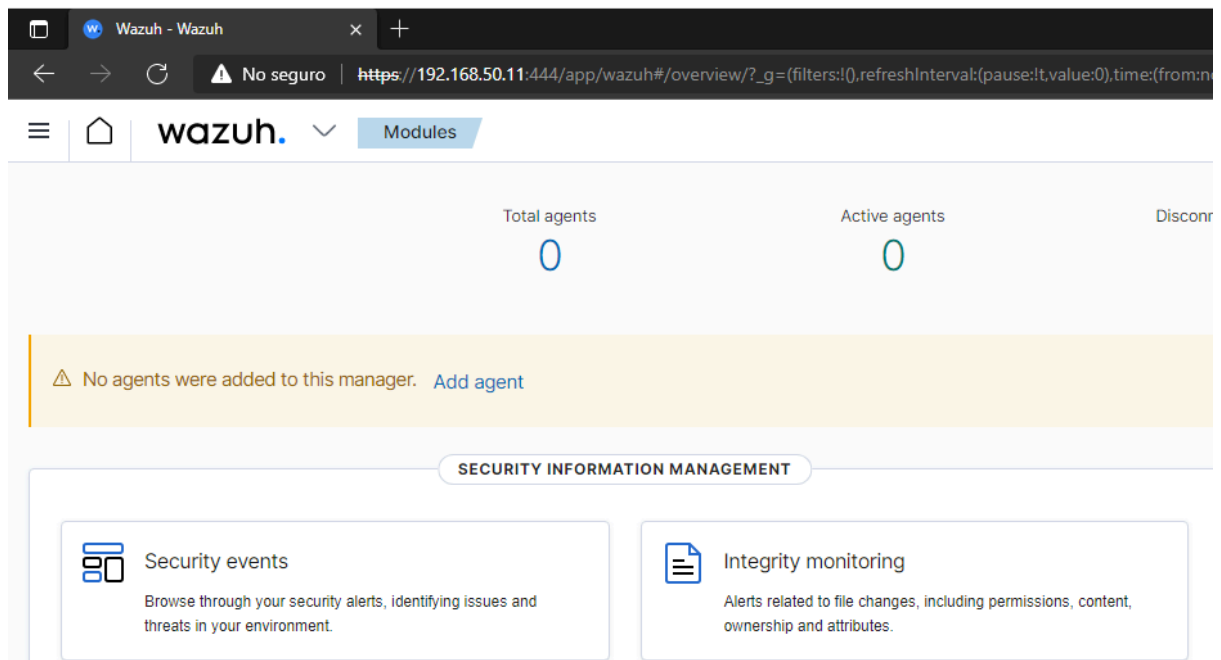
Una vez finalice la configuración ya nos saldrá el panel para loguearnos.



Implementación cliente Windows y Ubuntu

Windows:

Si no tenemos ningún cliente instalado, nada más acceder al panel de control nos saldrá el siguiente mensaje para añadir nuevos clientes, así que le daremos a añadir agente.



Seleccionamos el sistema operativo que tiene el cliente donde lo vamos a instalar y ponemos la IP del servidor. A continuación, ya nos dará los comandos que debemos ejecutar en nuestro cliente para hacer la instalación.

☐ RPM amd64 ☐ RPM aarch64
☐ DEB amd64 ☐ DEB aarch64

☒ MSI 32/64 bits

☐ Intel
☐ Apple silicon

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

✓ **Server address:**

This is the address the agent uses to communicate with the server. Enter an IP address or a fully qualified domain name (FDQN).

Assign a server address: ?

192.168.50.11

Al ser un cliente Windows vamos a powershell, ejecutamos el primer comando para que haga la instalación y el siguiente para arrancar el servicio. Y ya estaría configurado el cliente en Windows.

```
Administrador: Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Prueba la nueva tecnología PowerShell multiplataforma https://aka.ms/pscore6

PS C:\Windows\system32> Invoke-WebRequest -Uri https://packages.wazuh.com/4.x/windows/wazuh-agent-4.7.3-1.msi -OutFile $(env:tmp)\wazuh-agent; msixec.exe /i $(env:tmp)\wazuh-agent /q WAZUH_MANAGER='192.168.50.11' WAZUH_AGENT_GROUP='Windows'
PS C:\Windows\system32> NET START WazuhSvc

El servicio de Wazuh se ha iniciado correctamente.
PS C:\Windows\system32>
```

Ubuntu:

Deberemos realizar el primer paso igual que en Windows pero seleccionando como sistema operativo Ubuntu. Y nos dará los comandos que debemos meter en la terminal.

```
usuariodesk@usuariodesk-VirtualBox: ~
usuariodesk@usuariodesk-VirtualBox:~$ wget https://packages.wazuh.com/4.x/apt/pool/main/w/wazuh-agent/wazuh-agent_4.7.3-1_amd64.deb && sudo WAZUH_MANAGER='192.168.50.11' WAZUH_AGENT_GROUP='Linux' dpkg -i ./wazuh-agent_4.7.3-1_amd64.deb
--2024-05-02 22:04:10-- https://packages.wazuh.com/4.x/apt/pool/main/w/wazuh-agent/wazuh-agent_4.7.3-1_amd64.deb
Resolviendo packages.wazuh.com (packages.wazuh.com)... 18.154.48.117, 18.154.48.95, 18.154.48.50, ...
Conectando con packages.wazuh.com (packages.wazuh.com)[18.154.48.117]:443... conectado.
Petición HTTP enviada, esperando respuesta... 200 OK
Longitud: 9362524 (8,9M) [binary/octet-stream]
Guardando como: "wazuh-agent_4.7.3-1_amd64.deb"

wazuh-agent_4.7.3-1_amd64. 85%[=====] 7,63M 2,64MB/s
```

Habilitamos el servicio y lo arrancamos. Y ya estaría instalado y configurado en el cliente ubuntu.

```
Configurando wazuh-agent (4.7.3-1) ...
Procesando disparadores para systemd (245.4-4ubuntu3) ...
usuariodesk@usuariodesk-VirtualBox:~$ sudo systemctl daemon-reload
usuariodesk@usuariodesk-VirtualBox:~$ sudo systemctl enable wazuh-agent
Created symlink /etc/systemd/system/multi-user.target.wants/wazuh-agent.service → /lib/systemd/s
ystem/wazuh-agent.service.
usuariodesk@usuariodesk-VirtualBox:~$ sudo systemctl start wazuh-agent
usuariodesk@usuariodesk-VirtualBox:~$
```

Configuración

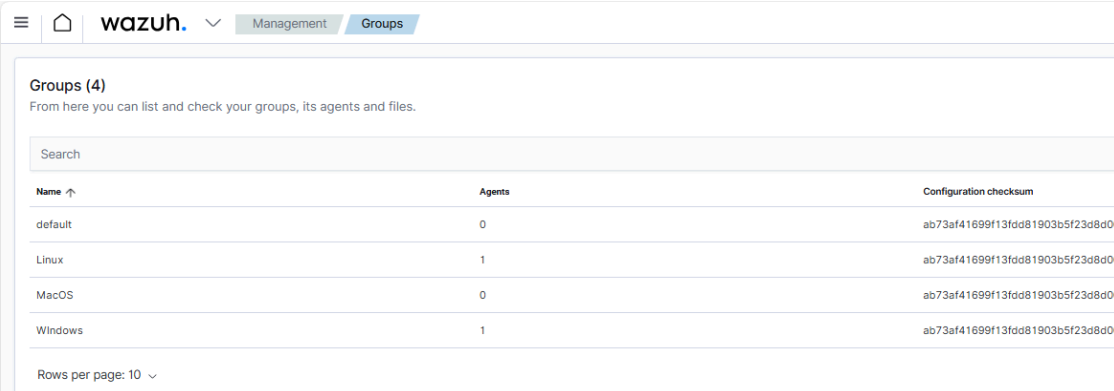
Firewall del servidor

Deberemos habilitar los siguientes puertos en el firewall del servidor para que Wazuh funcione correctamente:

- 22: Puerto SSH, lo habilitó porque el uso del servidor vía SSH es más cómodo. También lo necesitare para la comunicación de Ansible con los clientes Linux.
- 53: Monitoriza y analiza el tráfico DNS.
- 68: Monitoriza y analiza actividades relacionadas con el DHCP.
- 443: Puerto HTTPS será por el que accederemos al panel de control.
- 514,1514,1515,9200 y 55000: Envío de registros de actividades del sistema y de aplicaciones.

Grupos

Para tener mejor organizados a los clientes. Creare los siguientes grupos para organizarlos por sistemas operativos.



Name ↑	Agents	Configuration checksum
default	0	ab73af41699f13fdd81903b5f23d8d00
Linux	1	ab73af41699f13fdd81903b5f23d8d00
MacOS	0	ab73af41699f13fdd81903b5f23d8d00
Windows	1	ab73af41699f13fdd81903b5f23d8d00

Vulnerabilidades

El módulo de vulnerabilidades escanea todos los equipos clientes para identificar fallos de seguridad conocidos que podrían ser explotados por los atacantes. Lo deberemos activar en el archivo de configuración ossec.conf. Deberá quedar así:

```
98 <vulnerability-detector>
99   <enabled>yes</enabled>
100   <interval>5m</interval>
101   <min_full_scan_interval>6h</min_full_scan_interval>
102   <run_on_start>yes</run_on_start>
103
104 <!-- Ubuntu OS vulnerabilities -->
```

Virustotal

Vamos a implementar el módulo de virustotal que nos permitirá buscar virus en los archivos de los clientes.

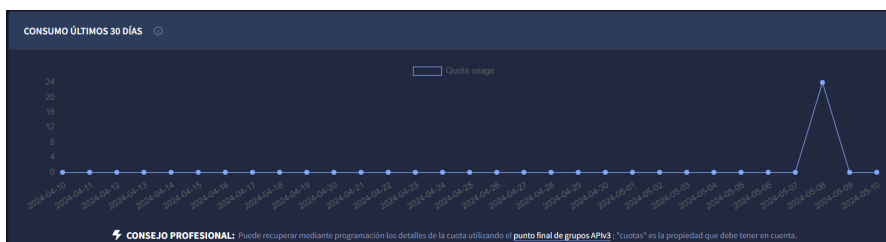
1. Primero deberemos dar de alta en Virustotal para conseguir la API.

<https://www.virustotal.com/gui/join-us>

La API gratuita esta limitada a lo siguiente (La premium tiene las búsquedas ilimitadas):

- 4 búsquedas por minuto
- 500 búsquedas por día
- 15500 búsquedas al mes

Si nos logueamos en virustotal. Tenemos un gráfico de las búsquedas que llevamos en el último mes.



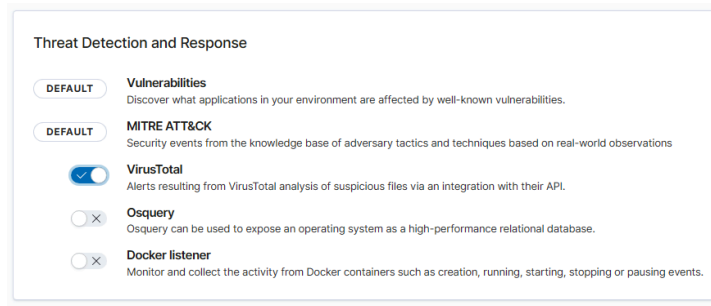
2. Una vez tenemos la API vamos al archivo de configuración de wazuh y deberemos añadir el siguiente código. Poniendo la API que nos han dado.

```
<ciscat_path>wodles/ciscat</ciscat_path>
</wodle>

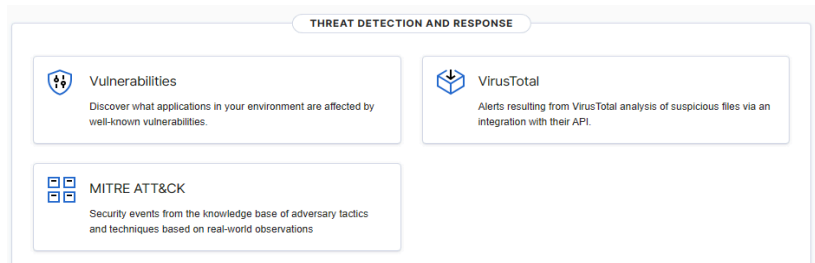
<integration>
  <name>virustotal</name>
  <api_key>4a1f3e915a34d65d03e97267<api_key> <!-- Replace with your VirusTotal API key -->
  <group>syscheck</group>
  <alert_format>json</alert_format>
</integration>

<!-- Osquery integration -->
<wodle name="osquery">
```

3. Ahora en Settings > Modules Debemos habilitar el módulo de virustotal.



4. Una vez configurado y activado el módulo ya nos aparecerá en el panel central.



5. En el archivo de configuración de Wazuh podremos configurar lo que queremos que nos escanee, la frecuencia del escaneo. Todo lo que está entre <syscheck> es la configuración del módulo de virustotal.

Podremos tanto especificar las rutas que queramos que nos escanee cómo poner otras rutas.

```
184 <!-- File integrity monitoring -->
185 <syscheck>
186   <disabled>no</disabled>
187   <!-- Frequency that syscheck is executed default every 12 hours -->
188   <frequency>43200</frequency>
189   <scan_on_start>yes</scan_on_start>
190   <!-- Generate alert when new file detected -->
191   <alert_new_files>yes</alert_new_files>
192   <!-- Don't ignore files that change more than 'frequency' times -->
193   <auto_ignore frequency="10" timeframe="3600">no</auto_ignore>
194   <!-- Directories to check (perform all possible verifications) -->
195   <directories>/etc,/usr/bin,/usr/sbin</directories>
196   <directories>/bin,/sbin,/boot</directories>
197   <!-- Files/directories to ignore -->
198   <ignore>/etc/mtab</ignore>
199   <ignore>/etc/hosts.deny</ignore>
200   <ignore>/etc/mail/statistics</ignore>
201   <ignore>/etc/random-seed</ignore>
202   <ignore>/etc/random.seed</ignore>
203   <ignore>/etc/adjtime</ignore>
204   <ignore>/etc/httpd/logs</ignore>
205   <ignore>/etc/utmpx</ignore>
206   <ignore>/etc/wtmpx</ignore>
207   <ignore>/etc/cups/certs</ignore>
208   <ignore>/etc/dumpdates</ignore>
209   <ignore>/etc/svc/volatile</ignore>
210   <!-- File types to ignore -->
211   <ignore type="sregex">.log$.swp$</ignore>
```

Correo

He intentado configurar el correo electrónico para que me mande alertas al correo cuando salte una alerta en Wazuh, pero no ha sido posible. Estos son los pasos que he seguido para la instalación y para la configuración.

Debemos entrar dentro del contenedor donde tenemos toda la configuración de Wazuh.

```
/tcp
usuario@server:~/wazuh-docker/single-node$ sudo docker exec -it single-node-wazuh.manager-1 /bin/bas
h
```

Actualizamos e instalamos los siguientes paquetes.

```
root@wazuh:/# apt-get update && apt-get install postfix mailutils libsasl2-2 ca-certificates libsasl
2-modules_
```

Aquí elegimos la opción 2 para poder mandar correos electrónicos a internet.

```
Please select the mail server configuration type that best meets your needs.

No configuration:
Should be chosen to leave the current configuration unchanged.
Internet site:
Mail is sent and received directly using SMTP.
Internet with smarthost:
Mail is received directly using SMTP or by running a utility such
as fetchmail. Outgoing mail is sent using a smarthost.
Satellite system:
All mail is sent to another machine, called a 'smarthost', for delivery.
Local only:
The only delivered mail is the mail for local users. There is no network.

1. No configuration      3. Internet with smarthost  5. Local only
2. Internet Site        4. Satellite system

General type of mail configuration: 2_
```

Ahora vamos al archivo de configuración de postfix. Lo configuramos, indicando el puerto 587 del smtp las rutas de los certificados, la ruta donde se encuentra la contraseña...

```
GNU nano 4.8 /etc/postfix/main.cf

# TLS parameters
smtpd_tls_cert_file=/etc/ssl/certs/ssl-cert-snakeoil.pem
smtpd_tls_key_file=/etc/ssl/private/ssl-cert-snakeoil.key
smtpd_tls_security_level=may

smtp_tls_CApath=/etc/ssl/certs
smtp_tls_security_level=may
smtp_tls_session_cache_database = btree:${data_directory}/smtp_scache

smtpd_relay_restrictions = permit_mynetworks permit_sasl_authenticated defer_unauth_destination
myhostname = wazuh.manager
alias_maps = hash:/etc/aliases
alias_database = hash:/etc/aliases
myorigin = /etc/mailname
mydestination = $myhostname, jontfg.com, wazuh.manager, localhost.manager, localhost
relayhost = [smtp.gmail.com]:587
smtp_sasl_auth_enable = yes
smtp_sasl_password_maps = hash:/etc/postfix/sasl_passwd
smtp_sasl_security_options = noanonymous
smtp_tls_CAfile = /etc/ssl/certs/ca-certificates.crt
smtp_use_tls = yes
smtpd_relay_restrictions = permit_mynetworks, permit_sasl_authenticated, defer_unauth_destination
mynetworks = 127.0.0.0/8 [::ffff:127.0.0.0]/104 [::1]/128
mailbox_size_limit = 0
recipient_delimiter = +
inet_interfaces = all
inet_protocols = all
```


Debo crear un gmail con doble factor y pedir la [contraseña de aplicación](#) y se la pasamos.

```
root@wazuh:/# echo [smtp.gmail.com]:587 jontfg2024@gmail.com:xioy shom xeoj ryea > /etc/postfix/sasl_passwd
root@wazuh:/# postmap /etc/postfix/sasl_passwd
sasl/          sasl_passwd
root@wazuh:/# postmap /etc/postfix/sasl_passwd
postmap: warning: /etc/postfix/main.cf, line 48: overriding earlier entry: smtpd_relay_restrictions=
permit_mynetworks permit_sasl_authenticated defer_unauth_destination
root@wazuh:/# chmod 400 /etc/postfix/sasl_passwd
root@wazuh:/# _
```

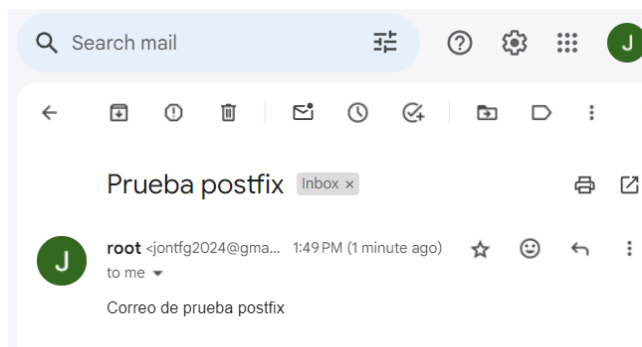
Protegemos nuestro archivo donde se encuentra la contraseña.

```
root@wazuh:/# chown root:root /etc/postfix/sasl_passwd /etc/postfix/sasl_passwd.db
root@wazuh:/# chown 0600 /etc/postfix/sasl_passwd /etc/postfix/sasl_passwd.db
root@wazuh:/# _
```

Reiniciamos postfix y ya estaría configurado. Mandamos un correo de prueba para ver si lo tenemos bien configurado.

```
root@wazuh:/# echo "Correo de prueba postfix" | mail -s "Prueba postfix" -r jontfg2024@gmail.com jontfg2024@gmail.com
root@wazuh:/# _
```

Vemos que hemos recibido el correo de prueba.



Ahora vamos a implementar el correo en wazuh. En el archivo de configuración especificamos lo siguiente. Cuando haya una alerta de nivel 7 nos notificará al correo electrónico.

Edit **ossec.conf** of **Manager**

```
1 <ossec_config>
2 <global>
3   <jsonout_output>yes</jsonout_output>
4   <alerts_log>yes</alerts_log>
5   <logall>no</logall>
6   <logall_json>no</logall_json>
7   <email_notification>yes</email_notification>
8   <smtp_server>localhost</smtp_server>
9   <email_from>jontfg2024@gmail.com</email_from>
10  <email_to>jontfg2024@gmail.com</email_to>
11  <email_maxperhour>12</email_maxperhour>
12  <email_log_source>alerts.log</email_log_source>
13  <agents_disconnection_time>10m</agents_disconnection_time>
14  <agents_disconnection_alert_time>0</agents_disconnection_alert_time>
15 </global>
16
17
18 <alerts>
19   <log_alert_level>3</log_alert_level>
20   <email_alert_level>7</email_alert_level>
21 </alerts>
```

Vemos que nos salta una alerta de nivel 7.

timestamp per 30 minutes						
Time	agent.name	rule.mitre.id	rule.mitre.tactic	rule.description	rule.level	rule.id
May 28, 2024 @ 19:29:28.231	DESKTOP-WIN10	T1070, 004, T1485	Defense Evasion, Impact	File deleted.	7	553

Y si vemos los logs nos intenta notificar por correo pero nos da un error de smtp.

```
2024/05/28 17:50:54 wazuh-maild: ERROR: (1765): RCPT TO not accepted by server - 'jontfg2024@gmail.com'.
2024/05/28 17:50:54 wazuh-maild: ERROR: (1263): Error Sending email to 127.0.0.1 (smtp server)
2024/05/28 17:57:24 wazuh-maild: ERROR: (1765): RCPT TO not accepted by server - 'jontfg2024@gmail.com'.
2024/05/28 17:57:24 wazuh-maild: ERROR: (1263): Error Sending email to 127.0.0.1 (smtp server)
```

He probado todas las siguientes opciones, pero no he conseguido solucionar el error:

- Probar con otros dominios de correo electrónico. Gmail, ProtonMail, Hotmail.
- Niveles de alertas más bajos.
- Alertas personalizadas.
- Probar con otro correo con dominio gmail por si el error podía estar en la contraseña de aplicación.
- Configuración de nateo de puertos en el docker compose.
- Añadir configuración adicional en el archivo ossec.conf.
- Diferentes configuraciones en la configuración de postfix.

Después de ver muchos vídeos, páginas y foros, creo que el error puede ser que estoy usando Wazuh en docker. Ya que en casi todos los sitios se usa la aplicación instalada en el servidor en vez de un docker.

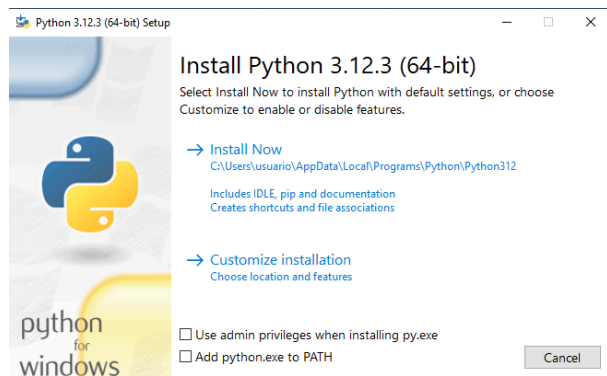
Script para autoborrado

Voy a configurar un script para que cuando VirusTotal detecte un archivo malicioso lo borre automáticamente.

Windows:

Para poder configurarlo en los clientes Windows deberemos tener instalado Python.

[Download Python | Python.org](https://www.python.org/downloads/windows/)



Una vez instalado, abrimos powershell e instalamos python.

```
PS C:\Windows\system32> pip install pyinstaller
Collecting pyinstaller
  Downloading pyinstaller-6.6.0-py3-none-win_amd64.whl.metadata (8.3 kB)
Collecting setuptools>=42.0.0 (from pyinstaller)
  Downloading setuptools-69.5.1-py3-none-any.whl.metadata (6.2 kB)
Collecting altgraph (from pyinstaller)
  Downloading altgraph-0.17.4-py2.py3-none-any.whl.metadata (7.3 kB)
Collecting pyinstaller-hooks-contrib>=2024.3 (from pyinstaller)
  Downloading pyinstaller_hooks_contrib-2024.6-py2.py3-none-any.whl.metadata (16 kB)
Collecting packaging>=22.0 (from pyinstaller)
  Downloading packaging-24.0-py3-none-any.whl.metadata (3.2 kB)
Collecting pefile>=2022.5.30 (from pyinstaller)
  Downloading pefile-2023.2.7-py3-none-any.whl.metadata (1.4 kB)
Collecting pywin32-ctypes>=0.2.1 (from pyinstaller)
  Downloading pywin32-ctypes-0.2.2-py3-none-any.whl.metadata (3.8 kB)
Collecting pyinstaller-6.6.0-py3-none-win_amd64.whl (1.3 MB)
  Downloading pyinstaller-6.6.0-py3-none-win_amd64.whl (1.3 MB)
  Downloading packaging-24.0-py3-none-any.whl (53 kB)
  Downloading pefile-2023.2.7-py3-none-any.whl (71 kB)
  Downloading pyinstaller_hooks_contrib-2024.6-py2.py3-none-any.whl (339 kB)
  Downloading pywin32-ctypes-0.2.2-py3-none-any.whl (30 kB)
  Downloading setuptools-69.5.1-py3-none-any.whl (894 kB)
  Downloading altgraph-0.17.4-py2.py3-none-any.whl (21 kB)
Installing collected packages: altgraph, setuptools, pywin32-ctypes, pefile, packaging, pyinstaller-hooks-contrib, pyinstaller
```

Debemos descargarnos este script de la página oficial de Wazuh. He tenido que modificar algunos parámetros, ya que me daba error. En mi GitHub lo he subido ya modificado el archivo.

```

Python 3.12.3 (tags/v3.12.3:f6650f9, Apr 9 2024, 14:05:25) [MSC v.1938 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> #!/usr/bin/python3
... # Copyright (C) 2015-2022, Wazuh Inc.
... # All rights reserved.
...
... import os
... import sys
... import json
... import datetime
...
... if os.name == 'nt':
...     LOG_FILE = "C:\\Program Files (x86)\\ossec-agent\\active-response\\active-responses.log"
... else:
...     LOG_FILE = "/var/ossec/logs/active-responses.log"
...
... ADD_COMMAND = 0
... DELETE_COMMAND = 1
... CONTINUE_COMMAND = 2
... ABORT_COMMAND = 3
...
... OS_SUCCESS = 0
... OS_INVALID = -1
...
... class message:
...     def __init__(self):
...         self.alert = ""
...         self.command = 0
...
...     def write_debug_file(ar_name, msg):
...         with open(LOG_FILE, mode="a") as log_file:
...             log_file.write(str(datetime.datetime.now().strftime('%Y/%m/%d %H:%M:%S')) + " " + ar_name + ": " + msg + "\n")
...
...     def setup_and_check_message(argv):
...
...         # get alert from stdin
...         input_str = ""
...         for line in sys.stdin:
...             input_str = line
...             break
...
...         try:
...             data = json.loads(input_str)
...         except ValueError:
...             extra_debug_file(argv[0], "Decoding JSON has failed (invalid input format)

```

Convertimos el script de python en un ejecutable de Windows.

```

PS C:\Users\usuario\Desktop> pyinstaller -F remove-threat.py
327 INFO: PyInstaller: 6.6.0, contrib hooks: 2024.6
327 INFO: Pythons: 3.12.3
359 INFO: Platform: Windows-10-10.0.19045-sp0
359 INFO: wrote C:\Users\usuario\Desktop\remove-threat.spec
375 INFO: Extending PYTHONPATH with paths
['C:\\Users\\usuario\\Desktop']
610 INFO: checking Analysis
610 INFO: Building Analysis because Analysis-00.toc is non existent
610 INFO: Running Analysis Analysis-00.toc
610 INFO: Target bytecode optimization level: 0
610 INFO: Initializing module dependency graph...
647 INFO: Caching module graph hooks...
655 INFO: Analyzing base library zip...
1908 INFO: Loading module hook 'hook-hheapq.py' from 'C:\\Users\\usuario\\AppData\\Local\\Programs\\Python\\Python312\\Lib\\site-packages\\PyInstaller\\hooks\\...'
2269 INFO: Loading module hook 'hook-encodings.py' from 'C:\\Users\\usuario\\AppData\\Local\\Programs\\Python\\Python312\\Lib\\site-packages\\PyInstaller\\hooks\\...'
4517 INFO: Loading module hook 'hook-pickle.py' from 'C:\\Users\\usuario\\AppData\\Local\\Programs\\Python\\Python312\\Lib\\site-packages\\PyInstaller\\hooks\\...'
6533 INFO: Caching module dependency graph...
6597 INFO: Looking for Python shared library...
6696 INFO: Using Python shared library: C:\Users\usuario\AppData\Local\Programs\Python\Python312\python312.dll
6696 INFO: Analyzing C:\Users\usuario\Desktop\remove-threat.py
6736 INFO: Processing module hooks...
6736 INFO: Performing binary vs. data reclassification (2 entries)
6783 INFO: Looking for ctypes DLLs
6799 INFO: Analyzing run-time hooks ...

```

Nos ha creado el siguiente archivo, así que lo movemos a la siguiente carpeta.

A screenshot of a Windows File Explorer window. The address bar shows the path: 'Este equipo > Disco local (C:) > Archivos de programa (x86) > ossec-agent > active-response > bin'. The main pane displays a list of files with columns for 'Nombre', 'Fecha de modificación', 'Tipo', and 'Tamaño'. The files listed are 'netsh.exe' (188 KB), 'remove-threat.exe' (7.090 KB), 'restart-wazuh.exe' (183 KB), and 'route-null.exe' (185 KB). The file 'remove-threat.exe' is highlighted with a blue selection bar. On the left side of the window, there are icons for 'Inicio', 'Este equipo', 'Disco local (C:)', and 'Archivos de programa (x86)'. The top of the window has tabs for 'Inicio', 'Compartir', 'Vista', and 'Herramientas de aplicación'.

Reiniciamos el servicio Wazuh.

```
PS C:\Windows\system32> Restart-Service -Name wazuh
PS C:\Windows\system32>
```

El cliente ya estaría completamente configurado. Ahora vamos a configurar la parte del servidor.

En el archivo de configuración ossec.conf añadimos lo siguiente, para que cuando salte una alerta con ese ID nos ejecute el exe.

```
71 <command>
72   <name>remove-threat</name>
73   <executable>remove-threat.exe</executable>
74   <timeout_allowed>no</timeout_allowed>
75 </command>
76
77 <active-response>
78   <disabled>no</disabled>
79   <command>remove-threat</command>
80   <location>local</location>
81   <rules_id>87105</rules_id>
82 </active-response>
```

Y en el archivo rules.locales.xml escribimos lo siguiente.

```
20 <group name="virustotal,">
21   <rule id="100092" level="12">
22     <if_sid>657</if_sid>
23     <match>Successfully removed threat</match>
24     <description>$(parameters.program) removed threat located at $(parameters.alert.data.virustotal.source.file)
25   </rule>
26
27   <rule id="100093" level="12">
28     <if_sid>657</if_sid>
29     <match>Error removing threat</match>
30     <description>Error removing threat located at $(parameters.alert.data.virustotal.source.file)</description>
31   </rule>
32 </group>
```

Ubuntu:

Bajamos el siguiente script de la página oficial de Wazuh que será el encargado de borrar los archivos maliciosos en Ubuntu.

```
usuariodesk@usuariodesk-VirtualBox: ~  
GNU nano 4.8 remove-threat.sh Modificado  
#!/bin/bash  
  
LOCAL=`dirname $0`;   
cd $LOCAL  
cd ../  
  
PWD=`pwd`  
  
read INPUT_JSON  
FILENAME=$(echo $INPUT_JSON | jq -r .parameters.alert.data.virustotal.source.file)   
COMMAND=$(echo $INPUT_JSON | jq -r .command)   
LOG_FILE="${PWD}/../logs/active-responses.log"  
  
#----- Analyze command -----#  
if [ ${COMMAND} = "add" ]  
then  
# Send control message to execd  
printf '{"version":1,"origin":{"name":"remove-threat","module":"active-respons  
  
read RESPONSE  
  
^G Ver ayuda ^O Guardar ^W Buscar ^K Cortar Text ^J Justificar ^C Posición  
^X Salir ^R Leer fich. ^\ Reemplazar ^U Pegar ^T Ortografía ^_ Ir a línea
```

Deberemos tener instalado el paquete jq.

```
usuariodesk@usuariodesk-VirtualBox:~$ sudo apt -y install jq  
Leyendo lista de paquetes... Hecho  
Creando árbol de dependencias  
Leyendo la información de estado... Hecho  
Se instalarán los siguientes paquetes adicionales:  
  libjq1 libonig5  
Se instalarán los siguientes paquetes NUEVOS:  
  jq libjq1 libonig5  
0 actualizados, 3 nuevos se instalarán, 0 para eliminar y 542 no
```

Lo copiamos en la carpeta adecuada y le damos los permisos y reiniciamos.

```
usuariodesk@usuariodesk-VirtualBox:~$ sudo mv remove-threat.sh /var/ossec/active  
-response/bin/  
usuariodesk@usuariodesk-VirtualBox:~$ sudo chmod 750 /var/ossec/active-response/  
bin/remove-threat.sh  
usuariodesk@usuariodesk-VirtualBox:~$ sudo chown root:wazuh /var/ossec/active-re  
sponse/bin/remove-threat.sh  
usuariodesk@usuariodesk-VirtualBox:~$
```

Ahora en el servidor en local_rules añadimos las dos siguientes reglas.

```
34 <group name="syscheck,pci_dss_11.5,nist_800_53_S1.7,">
35   <!-- Rules for Linux systems -->
36   <rule id="100200" level="7">
37     <if_sid>550</if_sid>
38     <field name="file">/root</field>
39     <description>File modified in /root directory.</description>
40   </rule>
41   <rule id="100201" level="7">
42     <if_sid>554</if_sid>
43     <field name="file">/root</field>
44     <description>File added to /root directory.</description>
45   </rule>
46 </group>
47 <group name="virustotal,">
48   <rule id="100092" level="12">
49     <if_sid>657</if_sid>
50     <match>Successfully removed threat</match>
51     <description>$(parameters.program) removed threat located at $(parameters.alert.data.virustotal.source.file)
52   </rule>
53   <rule id="100093" level="12">
54     <if_sid>657</if_sid>
55     <match>Error removing threat</match>
56     <description>Error removing threat located at $(parameters.alert.data.virustotal.source.file)</description>
57   </rule>
58 </group>
```

Y para acabar, añadimos esto a ossec.conf y reiniciamos.

```
<command>
  <name>remove-threat</name>
  <executable>remove-threat.sh</executable>
  <timeout_allowed>no</timeout_allowed>
</command>

<active-response>
  <disabled>no</disabled>
  <command>remove-threat</command>
  <location>local</location>
  <rules_id>87105</rules_id>
</active-response>
```

Y en el apartado syscheck agrego la siguiente línea para que me escanee solo la carpeta /root

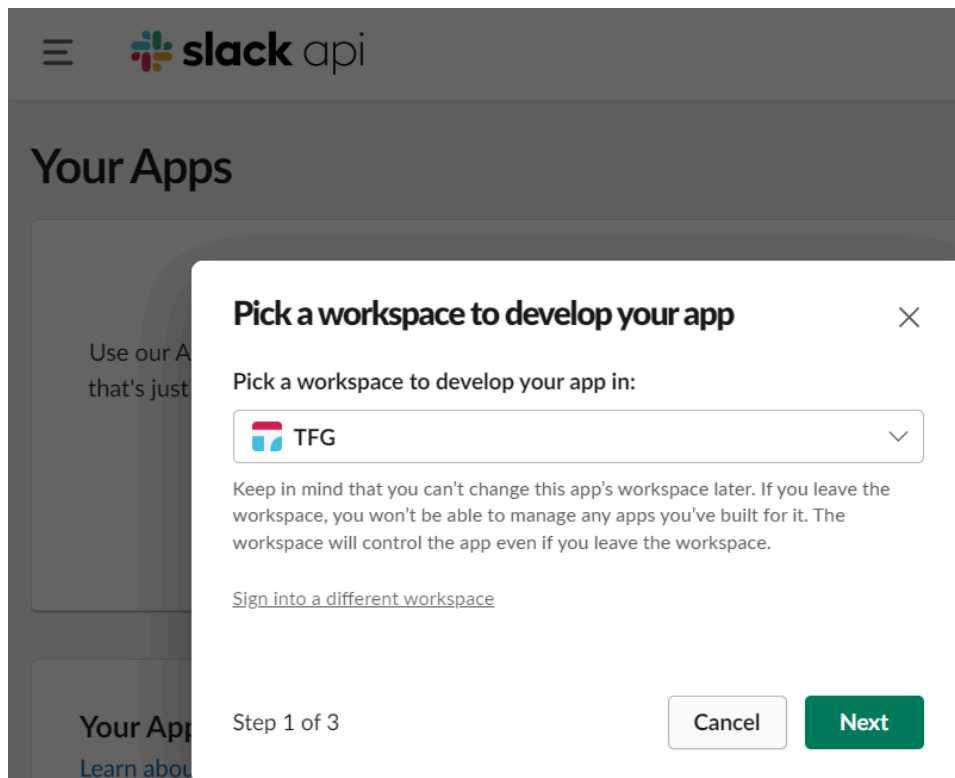
```
<syscheck>
  <disabled>no</disabled>

  <!-- Frequency that syscheck is executed default every 12 hours -->
  <frequency>43200</frequency>
  <directories realtime="yes">C:\Users\usuario\Downloads</directories>
  <directories realtime="yes">/root</directories>
  <scan on start>yes</scan on start>
```

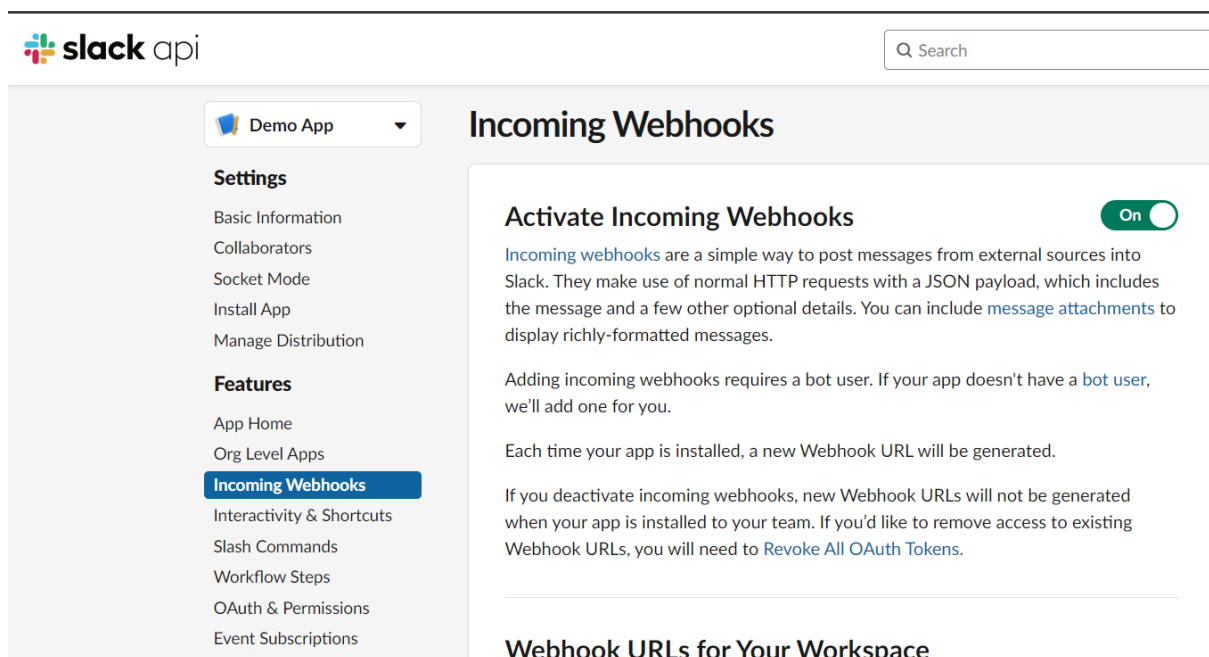
Slack

Ya que Wazuh nos genera muchas alertas, usaré slack para que me notifique los eventos más importantes. Podré estar informado en todo momento de los eventos. Ya que lo único que necesitamos es que el servidor tenga conexión a internet y accediendo a la página de slack podremos ver todo.

Nos registramos en Slack. Ahora entramos en API slack y vamos a empezar a configurarlo. Creamos una nueva APP, le damos un nombre al espacio de trabajo.



Una vez creado el espacio de trabajo, vamos al apartado de Incoming Webhooks y lo cambiamos a ON.



Debajo nos saldrá un botón Add New Webhook to workspace. Pulsamos en él.

Webhook URLs for Your Workspace

To dispatch messages with your webhook URL, send your [message](#) in JSON as the body of an `application/json` POST request.

Add this webhook to your workspace below to activate this curl example.

Sample curl request to post to a channel:

```
curl -X POST -H 'Content-type: application/json' --data '{"text":"Hello, World!"}' YOUR_WEBHOOK_URL_HERE
```

Webhook URL	Channel	Added By
No webhooks have been added yet.		

Add New Webhook to Workspace

Ponemos el nombre que será la bandeja de entrada donde nos entrarán los mensajes que vamos a configurar.



Esta aplicación fue creada por un miembro de tu espacio de trabajo, TFG.



Demo App solicita permiso para acceder al espacio de trabajo TFG

¿Dónde debe publicar Demo App?

Demo App requiere un canal en el que publicar como aplicación

eventos-wazuh

Cancelar

Permitir

Vemos que se nos ha creado, copiamos el link.

Webhook URLs for Your Workspace

To dispatch messages with your webhook URL, send your [message](#) in JSON as the body of an `application/json` POST request.

Add this webhook to your workspace below to activate this curl example.

Sample curl request to post to a channel:

```
curl -X POST -H 'Content-type: application/json' --data '{"text":"Hello, World!"}'  
https://hooks.slack.com/services/T076RED8K8C/B076J4DUV1C/UsbEcvWMn1EFIZgRbQIF9Jy  
q
```

Copy

Webhook URL	Channel	Added By
https://hooks.slack.com/services/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 Copy	#eventos-wazuh	jontfg2024 Jun 5, 2024 🗑️

Add New Webhook to Workspace

Ponemos el siguiente código en el archivo de configuración de Wazuh. He configurado para tener diferentes bandejas. Una será la general donde aparecerán todas las alertas. Y otras serán solo para alertas en específico. En el apartado de `hook_url` ponemos la URL que nos ha generado slack.

< Manager configuration

Refresh

Save

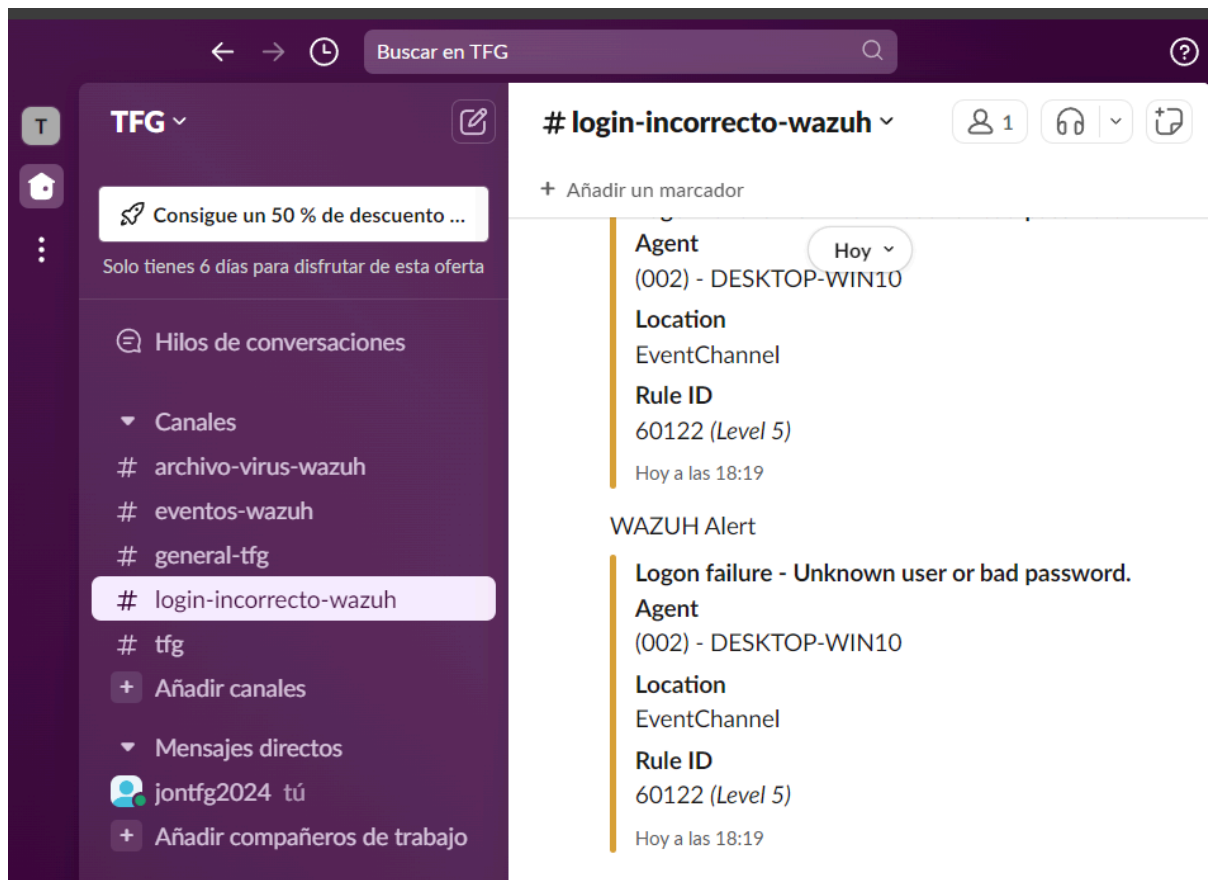
Restart Manager

Edit `ossec.conf` of Manager

① Changes will not take effect until a restart is performed.

```
23 <integration>  
24 <name>slack</name>  
25 <hook_url>https://hooks.slack.com/services/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</hook_url>  
26 <alert_format>json</alert_format>  
27 </integration>  
28  
29 <integration>  
30 <name>slack</name>  
31 <hook_url>https://hooks.slack.com/services/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</hook_url>  
32 <alert_format>json</alert_format>  
33 <rule_id>60122</rule_id>  
34 </integration>  
35  
36 <integration>  
37 <name>slack</name>  
38 <hook_url>https://hooks.slack.com/services/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</hook_url>  
39 <alert_format>json</alert_format>  
40 <rule_id>87105</rule_id>  
41 </integration>  
42
```

Vamos a slack y vemos que nos llegan las alertas que hemos configurado en las diferentes bandejas.



Optimización

He automatizado la forma de añadir nuevos clientes a nuestro servidor de Wazuh. Por si tenemos que añadir muchos clientes nuevos, no tengamos que ir de máquina en máquina instalando y configurando cada uno de ellos.

En las nuevas máquinas Linux, el único requisito que tienen que tener es que esté en la misma red que nuestro servidor y que tenga el SSH activado. Con el usuario, la contraseña y la IP del equipo ya podremos configurarlo todo remotamente sin acceder a la máquina.

En Windows, Ansible funciona de forma diferente ya que habría que hacer mucha configuración previa en la máquina para poder hacer todo remotamente. Ya que no sale rentable tener que hacer toda esa configuración previa he creado un script que con ejecutarlo en el nuevo equipo Windows se nos instalara y configura Wazuh.

También he automatizado la forma de configurar a cada cliente el script de autoborrado de archivos maliciosos. En las máquinas Linux se podrá hacer todo remoto con Ansible y en

Windows tendremos que ejecutar otro script en la propia máquina donde se quiere configurar.

[En mi github tengo subidos todos los archivos.](#)

Primero vamos a configurar Ansible en el servidor.

Añadimos el repositorio de Ansible en su última versión.

```
usuario@server:~$ sudo apt-add-repository ppa:ansible/ansible
[sudo] password for usuario:
Repository: 'deb https://ppa.launchpadcontent.net/ansible/ansible/ubuntu/ jammy main'
Description:
Ansible is a radically simple IT automation platform that makes your applications and systems easier
to deploy. Avoid writing scripts or custom code to deploy and update your applications- automate in
a language that approaches plain English, using SSH, with no agents to install on remote systems.

http://ansible.com/

If you face any issues while installing Ansible PPA, file an issue here:
https://github.com/ansible-community/ppa/issues
More info: https://launchpad.net/~ansible/+archive/ubuntu/ansible
Adding repository.
Press [ENTER] to continue or Ctrl-c to cancel.
```

Instalamos Ansible.

```
usuario@server:~$ sudo apt install ansible
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes adicionales:
  ansible-core python3-jmespath python3-kerberos python3-
python3-paramiko python3-requests-kerberos python3-requ
python3-winrm python3-xmldict sshpass
Paquetes sugeridos:
  python-nacl-doc python3-gssapi python3-invoke
Se instalarán los siguientes paquetes NUEVOS:
  ansible ansible-core python3-jmespath python3-kerberos
```

Instalamos y activamos el paquete para configurar el soporte de finalización de Ansible.

```
usuario@server:~$ sudo apt install python3-argcomplete
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
Se instalarán los siguientes paquetes NUEVOS:
  python3-argcomplete
0 actualizados, 1 nuevos se instalarán, 0 para eliminar y 14 no
Se necesita descargar 27,2 kB de archivos.
Se utilizarán 126 kB de espacio de disco adicional después de es
Des:1 http://es.archive.ubuntu.com/ubuntu jammy/universe amd64 p
..2 kB]
```

```
usuario@server:~$ sudo activate-global-python-argcomplete3
Installing bash completion script /etc/bash_completion.d/python-argcomplete.sh
usuario@server:~$ _
```

Configuramos las claves SSH en el servidor.

```
usuario@server:~$ ssh-keygen -t rsa -b 4096 -C "Ansible key"
Generating public/private rsa key pair.
Enter file in which to save the key (/home/usuario/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/usuario/.ssh/id_rsa
Your public key has been saved in /home/usuario/.ssh/id_rsa.pub
```

Creamos el usuario Ansible que será el que usemos.

```
usuario@server:~$ sudo adduser ansible
[sudo] password for usuario:
Adding user `ansible' ...
Adding new group `ansible' (1001) ...
Adding new user `ansible' (1001) with
```

Ahora configuramos el acceso sudo sin contraseña con el usuario Ansible.

```
usuario@server:~$ echo "ansible ALL=(ALL) NOPASSWD:ALL" | sudo tee /etc/sudoers.d/ansible
ansible ALL=(ALL) NOPASSWD:ALL
usuario@server:~$
```

Desactivo el inicio de sesión con contraseña del usuario Ansible recién creado.

```
usuario@server:~$ sudo usermod -L ansible
usuario@server:~$ _
```

Ya que todos los scripts los tengo en mi github usaré git para descargarlos. Instalo git.

```
usuario@server:~$ sudo apt install git
[sudo] password for usuario:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
git ya está en su versión más reciente (1:2.
```

Descargo los archivos para la implementación de los nuevos clientes.

```
usuario@server:~$ git clone https://github.com/joonw2/InstalacionClientesWazuh.git
Cloning into 'InstalacionClientesWazuh'...
remote: Enumerating objects: 37, done.
remote: Counting objects: 100% (37/37), done.
remote: Compressing objects: 100% (36/36), done.
remote: Total 37 (delta 17), reused 0 (delta 0), pack-reused 0
```

Descargo los archivos para la configuración del apartado de virustotal en los clientes.

```
usuario@server:~$ git clone https://github.com/joonw2/integracionvirustotalwazuh.git
Cloning into 'integracionvirustotalwazuh'...
remote: Enumerating objects: 55, done.
remote: Counting objects: 100% (55/55), done.
remote: Compressing objects: 100% (51/51), done.
Receiving objects: 100% (55/55), 21.85 KiB | 486.00 KiB/s, done.
remote: Total 55 (delta 18), reused 0 (delta 0), pack-reused 0
```

Los archivos playbook los meto en la ruta /etc/ansible/, ya que los scripts están configurados para que los ejecute desde ahí. Si se quiere modificar, solo habría que cambiar las rutas en los scripts.

```
usuario@server:~$ sudo mv InstalacionClientesWazuh/instalar.yml /etc/ansible/
usuario@server:~$ _

usuario@server:~$ sudo mv integracionvirustotalwazuh/virustotalplaybook.yml /etc/ansible/
usuario@server:~$ _
```

Ubuntu:

Una vez que ya tenemos configurado todo en el servidor es muy fácil añadir nuevos equipos. Ejecutamos en el servidor el script InsAgentLx.sh y nos pedirá usuario, contraseña e IP del nuevo equipo. Se añadirá y configurará todo automáticamente.

```
usuario@server:~$ ./InsAgentLx.sh
Nombre de usuario: prueba1
IP: 192.168.50.17
Contraseña:
spawn ssh-copy-id prueba1@192.168.50.17
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/usuario/.ssh/id_rsa.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed

/usr/bin/ssh-copy-id: WARNING: All keys were skipped because they already exist on the remote system
(if you think this is a mistake, you may want to use -f option)
BECOME password:
usuario@server:~$
```

Alguna vez salta un problema al ejecutarlo que se está bloqueando la instalación por el dpkg. Para solucionar esto deberemos ejecutar alguno de estos comandos en el equipo cliente:

- sudo rm /var/lib/dpkg/lock
- sudo rm /var/lib/dpkg/lock-frontent
- sudo rm /var/lib/apt/lists/lock
- sudo rm /var/cache/apt/archives/lock
- sudo dpkg --configure -a

He creado el siguiente playbook en Ansible por si me vuelve a pasar, que con ejecutarlo ya se nos arreglaría.

```
GNU nano 6.2 /etc/ansible/arreglardpkg.yml
---
- name: Limpiar dpkg
  hosts: all
  become: yes
  tasks:
    - name: Eliminar dpkg lock
      command: rm /var/lib/dpkg/lock

    - name: Eliminar dpkg lock-frontend
      command: rm /var/lib/dpkg/lock-frontend

    - name: Eliminar apt lists lock
      command: rm /var/lib/apt/lists/lock

    - name: Eliminar apt archives lock
      command: rm /var/cache/apt/archives/lock

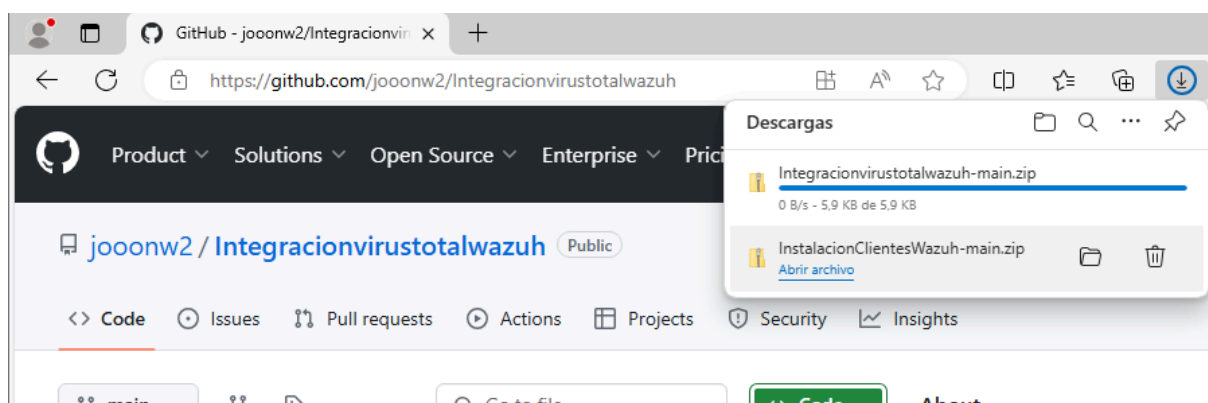
    - name: Configurar dpkg
      command: dpkg --configure -a
```

Ahora ejecutaremos el script de la instalación y configuración del script de autoborrado. Nos pedirá también el usuario, IP y contraseña y se nos configura todo automáticamente.

```
usuario@server:~$ ./virustotalplaybook.sh
Nombre de usuario: prueba1
IP: 192.168.50.17
BECOME password:
```

Windows:

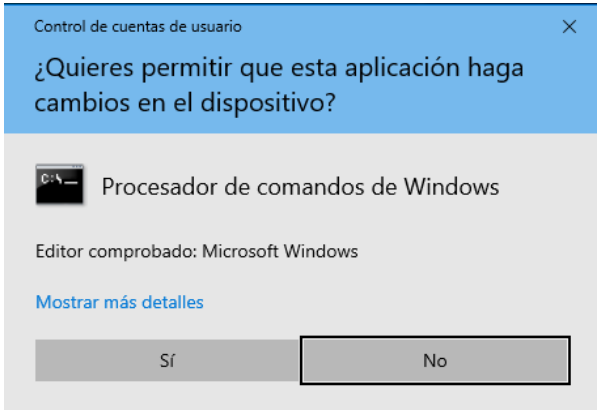
Como ya he contado antes, en Windows es muy tedioso configurar Ansible y no merece la pena, así que he creado los siguientes scripts que con ejecutarlos en la máquina donde queremos instalar Wazuh cliente nos vale.



Una vez descomprimido, ejecutamos el siguiente archivo.

InstalacionClientesWazuh-main				
	Nombre	Fecha de modificación	Tipo	Tamaño
InstAgentLx.sh	12/06/2024 17:36	Archivo SH	3 KB	
InstAgentWin.bat	12/06/2024 17:36	Archivo por lotes ...	1 KB	
instalar.yml	12/06/2024 17:36	Archivo YML	1 KB	
README.md	12/06/2024 17:36	Archivo MD	4 KB	

Nos pedirá permisos de administrador para ejecutarlo.



Si tenemos muy buen internet lo hace tan rápido que no da tiempo a ver lo que nos muestra el CMD. Para verificar la instalación, podemos ir a los servicios y ver que se nos ha añadido Wazuh y que está en ejecución.

Ubicador de llamada a proc...	en windows...	Manual	Servicio de rea
Uso de datos	Uso de dato...	En ejecu...	Automático
VirtualBox Guest Additions ...	Manages V...	En ejecu...	Automático
WalletService	Almacena o...	Manual	Sistema local
WarpJITSvc	Provides a JI...	Manual (dese...	Servicio local
Wazuh	Wazuh Win...	En ejecu...	Automático
Windows Installer	Agrega, mo...	En ejecu...	Manual

Y vemos en el servidor que ya nos salta el log de nuevo equipo.

>	Jun 12, 2024 @ 17:39:02.365	DESKTOP-69HMGSE	Wazuh agent started.	3	503
>	Jun 12, 2024 @ 17:38:51.452	DESKTOP-69HMGSE	Wazuh agent stopped.	3	506
>	Jun 12, 2024 @ 17:38:49.755	DESKTOP-69HMGSE	New wazuh agent connected.	3	501

Y ahora ejecutaremos el siguiente archivo para la implementación de virustotal.

<< Integracionvirustotalwazu... > Integracionvirustotalwazu-main					Buscar en Integracionvirust...	
	Nombre	Fecha de modificación	Tipo	Tamaño		
ido	integracionvirustotalwazu.bat	12/06/2024 17:36	Archivo por lotes ...	2 KB		
	integracionvirustotalwazu.sh	12/06/2024 17:36	Archivo SH	1 KB		
5	README.md	12/06/2024 17:36	Archivo MD	3 KB		
itos	remove-threat.py	12/06/2024 17:36	Archivo PY	4 KB		
	remove-threat.sh	12/06/2024 17:36	Archivo SH	1 KB		
	virustotalplaybook.sh	12/06/2024 17:36	Archivo SH	1 KB		
	virustotalplaybook.yml	12/06/2024 17:36	Archivo YML	1 KB		

Nos pide también permisos de administrador y empieza la instalación.

```
C:\> Seleccinar Administrador: C:\Windows\system32\cmd.exe
Descargando el instalador de Python...

Escribiendo solicitud web
Escribiendo secuencia de solicitud... (Número de bytes escritos: 4104266)
```

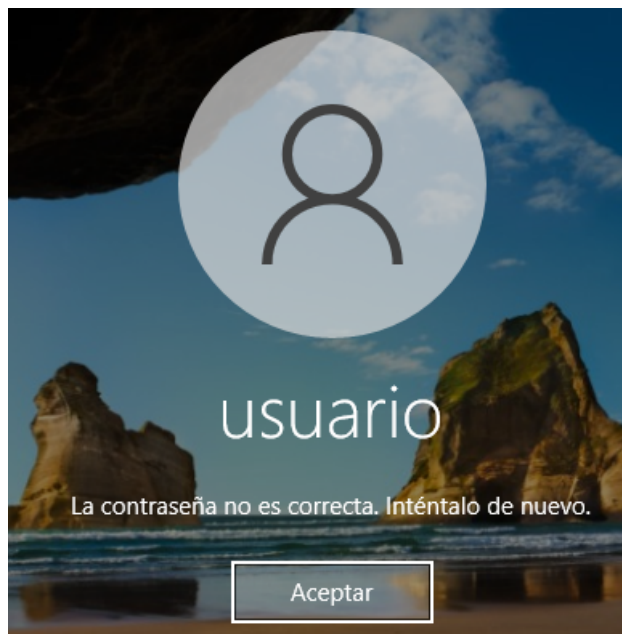
Aunque nos salen los siguientes mensajes de información nos lo hace correctamente. No he conseguido arreglar para que no salgan esos mensajes.

```
C:\> Administrador: C:\Windows\system32\cmd.exe
Descargando el instalador de Python...
Instalando Python...
Verificando la instalacion de Python...
Verificando la ubicaci|n de pip...
INFORMACIÓN: no se pudo encontrar ningún archivo para los patrones dados.
Verificando la ubicaci|n de pyinstaller...
INFORMACIÓN: no se pudo encontrar ningún archivo para los patrones dados.
Instalando PyInstaller...
"pip" no se reconoce como un comando interno o externo,
programa o archivo por lotes ejecutable.
Ejecutando PyInstaller...
```

Fase de pruebas

Login incorrecto

Si hacemos un login incorrecto en uno de nuestros equipos clientes vemos el siguiente evento. Si tuviéramos muchos intentos de sesión en poco tiempo del mismo equipo podríamos estar recibiendo un ataque de fuerza bruta.

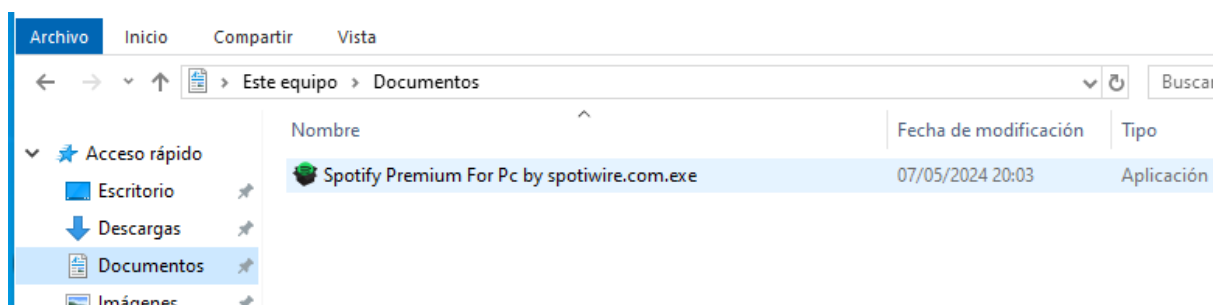


Podemos ver los eventos donde nos indica que la contraseña ha sido incorrecta.

>	May 31, 2024 @ 08:42:16.050	DESKTOP-WIN10	T1078, T1531	Defense Evasion, Persistence, Privilege Escalation, Initial Access, Impact	Logon failure - Unknown user or bad password.	5	60122
>	May 31, 2024 @ 08:42:12.257	DESKTOP-WIN10	T1078, T1531	Defense Evasion, Persistence, Privilege Escalation, Initial Access, Impact	Logon failure - Unknown user or bad password.	5	60122

Detección archivo malicioso y auto borrado con virustotal

Me he descargado este archivo que contiene virus.



Lo pongo en la carpeta descargas de Windows ya que es la única que tengo configurada para que la este vigilando virustotal. Vemos que Virustotal lo ha detectado de que es un archivo malicioso.

>	May 17, 2024 @ 13:40:14.271	-	-	-	-	-
>	May 17, 2024 @ 13:40:13.860	c:\users\usuario\downloads\spotify premium for pc by spotiwire.com.exe	https://www.virustotal.com/gui/file/7a2fe2a78359d2c3d16492187841049cbfb4185a2fe9ce6882c72ba0b9d9f6c/detection/f-7a2fe2a78359d2c3d16492187841049cbfb4185a2fe9ce6882c72ba0b9d9f6c-1715539014	1	1	73
>	May 17, 2024 @ 13:40:11.691	-	-	-	-	-
>	May 17, 2024 @ 13:40:10.400	c:\users\usuario\downloads\spotify premium for pc by spotiwire.com.exe	https://www.virustotal.com/gui/file/7a2fe2a78359d2c3d16492187841049cbfb4185a2fe9ce6882c72ba0b9d9f6c/detection/f-7a2fe2a78359d2c3d16492187841049cbfb4185a2fe9ce6882c72ba0b9d9f6c-1715539014	1	1	73

Como nos ha detectado que es un archivo malicioso nos lo borra automáticamente.

>	May 17, 2024 @ 13:40:14.271	active-response/bin/remove-threat.exe removed threat located at c:\users\usuario\downloads\spotify premium for pc by spotiwire.com.exe	12	100092
>	May 17, 2024 @ 13:40:13.860	VirusTotal: Alert - c:\users\usuario\downloads\spotify premium for pc by spotiwire.com.exe - 1 engines detected this file	12	87105
>	May 17, 2024 @ 13:40:11.691	active-response/bin/remove-threat.exe removed threat located at c:\users\usuario\downloads\spotify premium for pc by spotiwire.com.exe	12	100092
>	May 17, 2024 @ 13:40:10.681	File deleted.	7	553
>	May 17, 2024 @ 13:40:10.400	VirusTotal: Alert - c:\users\usuario\downloads\spotify premium for pc by spotiwire.com.exe - 1 engines detected this file	12	87105
>	May 17, 2024 @ 13:40:07.565	File added to the system.	5	554

Detección de fuerza bruta

Deberemos crear un diccionario con las contraseñas que vamos a usar para intentar iniciar sesión. Lanzo el ataque al protocolo SSH al equipo 192.168.50.15 con la herramienta hydra.

```
usuariodesk@usuariodesk-VirtualBox:~$ sudo hydra -l badguy -P passwords.txt 192.168.50.15 ssh
Hydra v9.0 (c) 2019 by van Hauser/THC - Please do not use in military or secret service organizations, or for illegal purposes.
```

Vemos que detecta todas las sesiones erróneas y, al recibir muchas en poco tiempo, nos notifica que es un ataque de fuerza bruta al protocolo SSH.

>	Jun 7, 2024 @ 12:06:45.216	usuariodesk-VirtualBo x PAM: User login failed.	5	5583
>	Jun 7, 2024 @ 12:06:45.216	usuariodesk-VirtualBo x sshd: brute force trying to get access to the system. Non existent user.	10	5712
>	Jun 7, 2024 @ 12:06:45.216	usuariodesk-VirtualBo x PAM: User login failed.	5	5583

Vulnerabilidades

Cada día se hace un escaneo a cada cliente para encontrar si tiene vulnerabilidades. Como vemos nos muestra diferentes vulnerabilidades con severidad alta, que con el cve si existiera el exploit podría ser explotada.

>	May 17, 2024 @ 14:11:22.750	libasn1-8-heimdal	CVE-2022-41916	High
>	May 17, 2024 @ 14:11:22.654	xxd	CVE-2022-1927	High
>	May 17, 2024 @ 14:11:22.542	libgststreamer-plugins-good1.0-0	CVE-2022-1925	High
>	May 17, 2024 @ 14:11:22.444	gststreamer1.0-pulseaudio	CVE-2022-1925	High

Conclusiones finales

Grado de cumplimiento de los objetivos fijados.

Mi idea inicial era desarrollar un proyecto de monitorización de red utilizando la herramienta Velociraptor. Después de investigar cómo funciona y explorar todas sus posibilidades. Me di cuenta que había poca documentación disponible más allá de la oficial y poca gente lo usaba. Además, sus posibilidades se quedaban cortas para lo que quería.

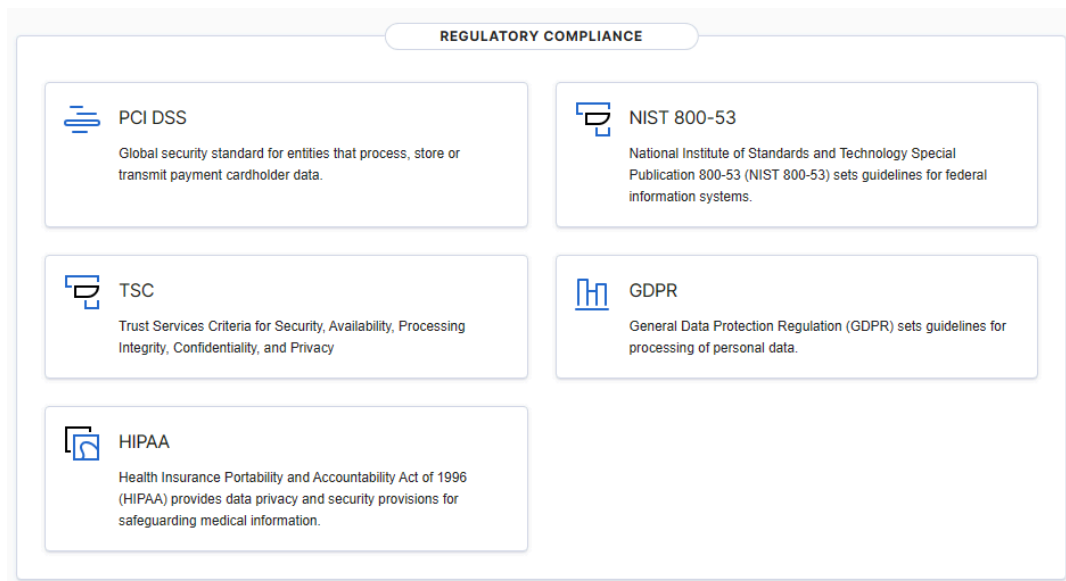
Tras investigar y consultar otras herramientas que pudieran hacer la misma función, me decidí por Wazuh ya que es de código abierto, tiene muchas funcionalidades y últimamente se está haciendo un hueco entre las empresas más grandes de ciberseguridad.

Propuesta de modificaciones o ampliaciones futuras del sistema implementado.

En futuras mejoras de la herramienta, se podrá implementar lo siguiente:

Modulo Regulatory Compliance

El apartado "Regulatory Compliance" en Wazuh es una sección para ayudar a las organizaciones a cumplir regulaciones y estándares de seguridad establecidas por entidades gubernamentales, industrias específicas, etc.



PCI DSS (Payment Industry Data Security Standard): Es un conjunto de estándares diseñados para garantizar la seguridad de las transacciones con tarjetas de pago. Establece requisitos para la protección de datos de los titulares, seguridad de las redes y el control de acceso.

NIST 800-53: Proporciona pautas y controles de seguridad para sistemas de información federales en agencias gubernamentales. Incluye controles para la gestión de riesgos, seguridad de datos, acceso y autenticación.

TSC (Security Technical Implementation Guide - STIG): Guía técnica para asegurar la conformidad con los estándares de seguridad. Proporciona directrices detalladas para configurar sistemas y aplicaciones de manera segura.

GDPR (General Data Protection Regulation): Regulación de la Unión Europea que establece normas para la protección de datos personales de los ciudadanos.

HIPAA (Health Insurance Portability and Accountability Act): Es una ley estadounidense que establece estándares para la protección y seguridad de la información médica y de salud personal identificable.

Modulo Cloud security monitoring

El apartado Cloud security monitoring en Wazuh está diseñado para proporcionar una visibilidad y control sobre entornos en la nube, detectar actividades sospechosas, responder a incidentes y asegurar el cumplimiento normativo. Para poder usar este módulo todas las cuentas de las plataformas deberán ser de pago. Lo podemos implementar en las siguientes plataformas:

- Amazon S3
- Azure
- Google Cloud

- GitHub
- Office 365

Cloud security monitoring

Name	Description
Amazon S3	Security events related to Amazon AWS services, collected directly via AWS API
Azure Logs	Configuration options of the Azure Logs wodle
Google Cloud Pub/Sub	Configuration options of the Google Cloud Pub/Sub module
GitHub	Detect threats targeting GitHub organizations
Office 365	Configuration options of the Office 365 module

Documentación final

Docker

Contenedor

Usare el siguiente contenedor: <https://github.com/wazuh/wazuh-docker> en la versión 4.7

Estructura

La estructura del contenedor docker que vamos a usar es la siguiente:

```

wazuh-docker/single-node/
├── config
│   ├── certs.yml
│   ├── wazuh_cluster
│   │   ├── wazuh_manager.conf
│   │   └── wazuh_dashboard
│   │       ├── opensearch_dashboards.yml
│   │       └── wazuh.yml
│   ├── wazuh_indexer
│   │   ├── internal_users.yml
│   │   ├── wazuh_indexer.yml
│   │   └── wazuh_indexer_ssl_certs
│   │       ├── admin-key.pem
│   │       ├── admin.pem
│   │       ├── root-ca.key
│   │       ├── root-ca-manager.key
│   │       ├── root-ca-manager.pem
│   │       ├── root-ca.pem
│   │       ├── wazuh_dashboard-key.pem
│   │       ├── wazuh_dashboard.pem
│   │       ├── wazuh_indexer-key.pem
│   │       ├── wazuh_indexer.pem
│   │       ├── wazuh_manager-key.pem
│   │       └── wazuh_manager.pem
├── docker-compose.yml
├── generate-indexer-certs.yml
└── README.md

```

config/certs.yml: Archivo para la configuración de certificados SSL.

config/wazuh_cluster: Configuración del cluster. El cluster se encarga de gestionar y analizar los datos

config/wazuh_dashboard: Configuración del dashboard. El dashboard es la interfaz gráfica con la que podemos interactuar con Wazuh

config/wazuh_indexer: Configuración del indexer. El indexer es el responsable de almacenar y gestionar los datos que provienen de los agentes

config/wazuh_indexer_ssl_certs: Archivo que contiene certificados SSL para el indexador Elasticsearch

docker-compose.yml: Archivo donde tenemos toda la configuración del docker compose

generate-indexer-certs.yml: Archivo para configurar los certificados de Elasticsearch

readme.md: Archivo donde nos especifica los pasos que debemos configurar antes de levantar el docker y como levantarlo.

Manual de Instalación

Lo instalaré sobre un Ubuntu Server 22.04 LTS en docker. Primero instalamos algunos paquetes de requisitos previos.

```
usuario@velociraptorsrv:~$ sudo apt install apt-transport-https ca-certificates curl software-properties-common
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
ca-certificates ya está en su versión más reciente (20230311ubuntu0.20.04.1).
fijado ca-certificates como instalado manualmente.
curl ya está en su versión más reciente (7.68.0-1ubuntu2.21).
fijado curl como instalado manualmente.
```

Añadimos la clave GPG para el repositorio docker.

```
usuario@velociraptorsrv:~$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
OK
usuario@velociraptorsrv:~$
```

Agregamos el repositorio docker a las fuentes de APT.

```
usuario@velociraptorsrv:~$ sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu focal stable"
Des:1 https://download.docker.com/linux/ubuntu focal InRelease [57,7 kB]
Des:2 https://download.docker.com/linux/ubuntu focal/stable amd64 Packages [37,8 kB]
Obj:3 http://es.archive.ubuntu.com/ubuntu focal InRelease
Obj:4 http://es.archive.ubuntu.com/ubuntu focal-updates InRelease
Obj:5 http://es.archive.ubuntu.com/ubuntu focal-backports InRelease
```

Actualizamos los paquetes recién agregados.

```
usuario@velociraptorsrv:~$ sudo apt update
Obj:1 http://es.archive.ubuntu.com/ubuntu focal InRelease
Obj:2 https://download.docker.com/linux/ubuntu focal InRelease
Obj:3 http://es.archive.ubuntu.com/ubuntu focal-updates InRelease
Obj:4 http://es.archive.ubuntu.com/ubuntu focal-backports InRelease
Obj:5 http://es.archive.ubuntu.com/ubuntu focal-security InRelease
Leyendo lista de paquetes... Hecho
```

Instalamos el repositorio docker.

```
usuario@velociraptor:~$ sudo apt install docker-ce
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
Los paquetes indicados a continuación se instalaron de forma automática y ya no son necesarios.
  libfupdpplugin1 libxmlb1
Utilice «sudo apt autoremove» para eliminarlos.
Se instalarán los siguientes paquetes adicionales:
  containerd.io docker-buildx-plugin docker-ce-cli docker-ce-rootless-extras docker-compose-plugin
  pigz slirp4netns
Paquetes sugeridos:
  aufs-tools cgroupfs-mount | cgroup-lite
Se instalarán los siguientes paquetes NUEVOS:
  containerd.io docker-buildx-plugin docker-ce docker-ce-cli docker-ce-rootless-extras
```

Clonamos el repositorio de Wazuh del Github oficial.

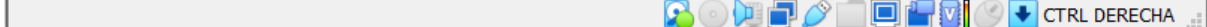
```
usuario@server:~$ git clone https://github.com/wazuh/wazuh-docker.git -b v4.7.3_
```

Ahora generamos los certificados que vamos a usar en wazuh.

```
usuario@server:~/wazuh-docker/single-node$ sudo docker-compose -f generate-indexer-certs.yml run --rm generator
WARN[0000] Found orphan containers ([single-node-wazuh.dashboard-1 single-node-wazuh.indexer-1 single-node-wazuh.manager-1]) for this project. If you removed or renamed this service in your compose file, you can run this command with the --remove-orphans flag to clean it up.
The tool to create the certificates exists in the in Packages bucket
29/04/2024 16:55:34 INFO: Admin certificates created.
29/04/2024 16:55:34 INFO: Wazuh indexer certificates created.
29/04/2024 16:55:34 INFO: Wazuh server certificates created.
29/04/2024 16:55:34 INFO: Wazuh dashboard certificates created.
Moving created certificates to the destination directory
Changing certificate permissions
Setting UID indexer and dashboard
Setting UID for wazuh manager and worker
usuario@server:~/wazuh-docker/single-node$ _
```

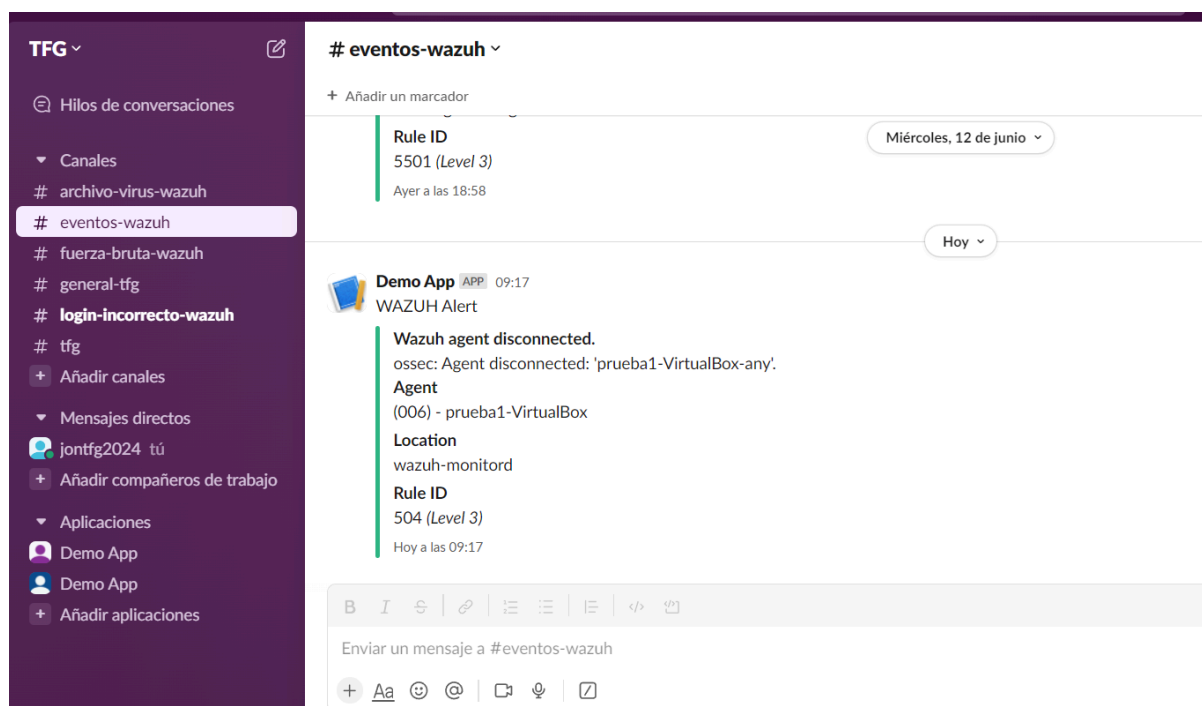
Ahora ya podemos levantar el docker. Lo levanto en segundo plano.

```
usuario@server:~/wazuh-docker/single-node$ sudo docker-compose up -d
[+] Running 3/3
  Container single-node-wazuh.indexer-1   Running      0.0s
  Container single-node-wazuh.manager-1   Running      0.0s
  Container single-node-wazuh.dashboard-1 Start...     0.0s
usuario@server:~/wazuh-docker/single-node$ _
```

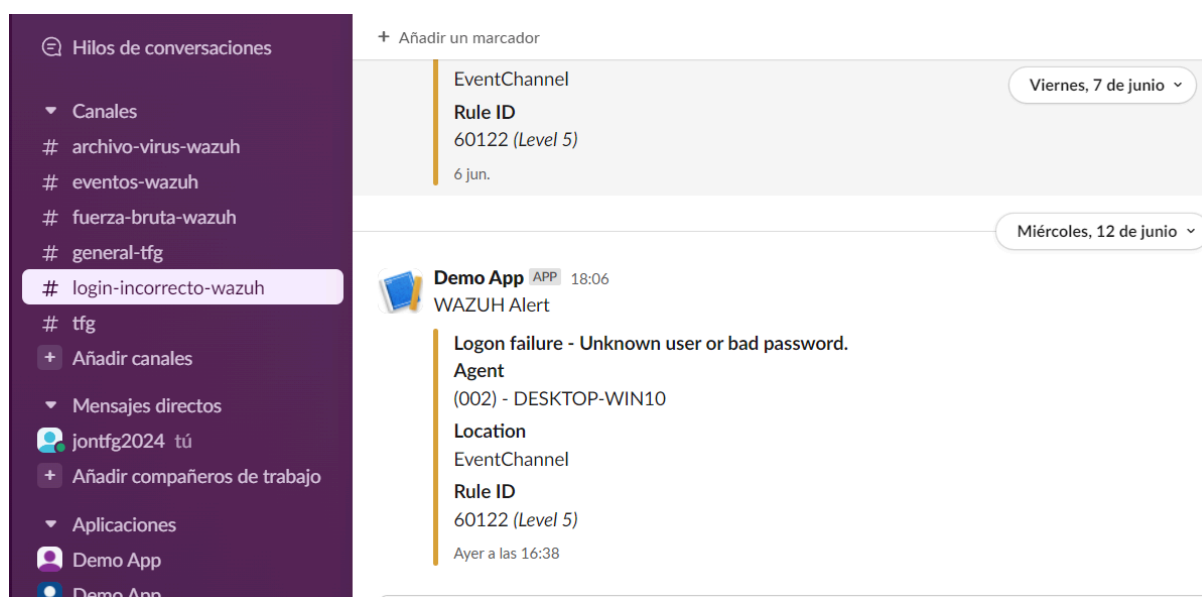


Manual de uso

Una vez tengamos ya implementado y configurado todo será muy fácil su uso, deberemos ir a Slack y estar al tanto de las nuevas notificaciones que tenemos. Como hemos configurado diferentes bandejas de entrada, tendremos una principal donde se nos notificará de todas las alertas. En esta bandeja nos saldrán todas las alertas. De está no deberemos estar muy atentos ya que llegarán muchos eventos que no resultan ningún peligro.



De las que debemos estar atentos son de las siguientes. Login-incorrecto, archivo-virus y fuerza-bruta ya que si nos llega una notificación deberíamos analizarla.



Tanto si hemos recibido un ataque o queremos ver las especificaciones de un cliente para ver posibles vulnerabilidades y brechas de seguridad podremos generar un reporte de cada equipo donde nos mostrará versiones de sistemas operativos, puertos abiertos, servicios corriendo, interfaces de red...

Para verlo deberemos ir al apartado de Agentes. Nos saldrán todos los clientes que tenemos añadidos y si están en este momento encendidos o no.

Agents (3)

🔍 id!=000 and

Search

WQL

Refresh

Deploy new agent

Refresh

Export formatted

⚙️

ID ↑	Name	IP address	Group(s)	Operating system	Version	Status	Actions
002	DESKTOP-WIN10	192.168.50.6	Windows	Microsoft Windows 10 Pro 10.0.19045.4412	v4.7.3	<div><div></div><div>?</div><div>👁</div><div>🔗</div></div>	
003	usuariodesk-VirtualBox	192.168.50.15	Linux	Ubuntu 20.04 LTS	v4.7.3	<div><div></div><div>?</div><div>👁</div><div>🔗</div></div>	
005	EquipoWIN10deprueba	192.168.50.16	Windows	Microsoft Windows 10 Pro 10.0.19045.2965	v4.7.3	<div><div></div><div>?</div><div>👁</div><div>🔗</div></div>	

Rows per page: 10

< 1 >

Si vamos a un cliente en concreto, lo primero que podemos ver son las especificaciones del equipo, interfaces de red y puertos abiertos.

☰

🏠

wazuh.

▼

Agents

DESKTOP-WIN10

Inventory data

a

?

DESKTOP-WIN10

Generate report

Cores: 2

Memory: 4095.55 MB

Arch: x86_64

Operating system: Microsoft Windows 10 Pro 10.0.19045.4412

CPU: AMD Ryzen 7 5700U with Radeon Graphics

Host name: DESKTOP-WIN10

Board serial: 0

Last scan: Jun 6, 2024 @ 09:01:28.000

Network interfaces (3)

Refresh

Export formatted

Search

WQL

Name ↑	MAC	State	MTU	Type
Ethernet	08:00:27:2e:96:bf	up	1500	ethernet
Ethernet 2	08:00:27:71:9b:f8	up	1500	ethernet
Loopback Pseudo-Interface 1	00:00:00:00:00:00	up	2147483647	

Rows per page: 10

< 1 >

Network ports (33)

Refresh

Export formatted

Search

WQL

Local port	Local IP address	Process	State	Protocol
68	0.0.0.0	svchost.exe		udp
135	::	svchost.exe	listening	tcp6
135	0.0.0.0	svchost.exe	listening	tcp
137	192.168.50.6	System		udp
138	192.168.50.6	System		udp
139	192.168.50.6	System	listening	tcp
445	0.0.0.0	System	listening	tcp
445	::	System	listening	tcp6
500	::	svchost.exe		udp6
500	0.0.0.0	svchost.exe		udp

Rows per page: 10

< 1 2 3 4 >

Podremos ver mucha más información, también podemos generar un reporte que nos extraerá toda la información y nos la exportará en un PDF. Le damos a generar un reporte y nos crea el PDF. Para descargarlo, debemos ir a Management>Reporting

File	Size	Created ↓	Actions
wazuh-agent-inventory-002-1717658442.pdf	36.08KB	Jun 6, 2024 @ 09:20:43.825	Download Copy

Rows per page: 10 ▾

Desde aquí ya nos lo podremos descargar

The screenshot shows a web browser window with the Wazuh interface. The URL is <https://192.168.50.11/reports/wazuh-agent-inventory-002-1717658442.pdf>. The page displays the Wazuh logo and contact information (info@wazuh.com, https://wazuh.com). The main content is an 'Inventory data report' for agent 002. It includes a table with agent details and sections for hardware and operating system information.

ID	Name	IP address	Version	Manager	Operating system	Registration date	Last keep alive
002	DESKTOP-WIN10	192.168.50.6	Wazuh v4.7.3	wazuh.manager	Microsoft Windows 10 Pro 10.0.19045.4412	May 3, 2024 @ 15:25:35.000	Jun 6, 2024 @ 07:20:39.000

Hardware information

- 2 cores
- AMD Ryzen 7 5700U with Radeon Graphics
- 4.00GB RAM

Operating system information

- x86_64
- Microsoft Windows 10 Pro 10.0.19045.4412

Bibliografía

Toda la documentación consultada: libros, apuntes, páginas webs, foros, etc.

<https://docs.docker.com/engine/install/debian/>

[Configuration file - Wazuh dashboard · Wazuh documentation](#)

[Detecting and removing malware using VirusTotal integration \(wazuh.com\)](#)

[Wazuh - Protege tus Activos Digitales: Instalación de Agentes Wazuh \(youtube.com\)](#)

[Setting Up Virus Total With Wazuh For Windows Endpoint \(youtube.com\)](#)

[Desplegando #WAZUH - Herramienta Esencial para la detección de intrusos \(youtube.com\)](#)

[Installing & Configuring Wazuh \(youtube.com\)](#)

<https://documentation.wazuh.com/current/user-manual/manager/manual-email-report/smtp-authentication.html>

<https://www.redhat.com/es/topics/automation/learning-ansible-tutorial>

<https://github.com/wazuh/wazuh/issues/15825>

<https://stackoverflow.com/questions/76071086/im-trying-to-configure-email-out-in-wazuh-but-failed-to-do-so-i-followed-the-of>

https://www.reddit.com/r/Wazuh/comments/136vdq8/email_alerts/

https://www.youtube.com/watch?v=kH8tfraVzFk&ab_channel=Pentester77

https://www.youtube.com/watch?v=g3aXZbHfLk&ab_channel=WorldofInnovations

https://www.youtube.com/watch?v=dSHJ_u02qGc&ab_channel=UnrealLabs

<https://www.elmundo.es/uestudio/2024/04/03/660d7930e9cf4a94058b4585.html>

<https://blog.invgate.com/es/ansible>

https://docs.rockylinux.org/es/books/learning_ansible/02-advanced/