

Universidad de Sevilla

Escuela Técnica Superior de Ingeniería Informática



**PAI 1 - Sistema de detección de intrusos para
almacenamiento masivo basado en verificadores de
seguridad**

Grado en Ingeniería Informática – Ingeniería del Software

Seguridad en Sistemas Informáticos e Internet

Curso 2023 – 2024

Participantes:

Juan Luis Ruano Muriedas

José Joaquín Rojas Romero

Antonio José Suárez García

Índice

Índice.....	2
Resumen ejecutivo.....	3
Esquema de la arquitectura de la aplicación Wazuh y decisiones de diseño.....	3
1. Cálculo, almacenamiento y comprobación de hash.....	5
2. Estructura para almacenar los Hash de forma eficiente:.....	6
3. Debilidades de los HIDS que Wazuh puede prevenir:.....	6
4. Fichero de configuración.....	7
5. Logs de actualización diaria.....	8
6. Informe mensual.....	9
7. Pruebas de archivos.....	10
Bibliografía.....	10

Resumen ejecutivo

Para el desarrollo de la práctica hemos optado por el uso de **Wazuh**, una plataforma de seguridad informática de código abierto utilizada para la detección de intrusos, monitorización de logs, análisis de seguridad y respuesta ante incidentes. Proporciona capacidades integrales de seguridad, incluyendo detección de amenazas en tiempo real, correlación de eventos, análisis forense, y gestión de cumplimiento normativo.

Otra opción que discutimos al comienzo de la práctica fue el uso de **Ossec**, pero cambiamos a **Wazuh**, ya que **Ossec** se encuentra **obsoleto**.

En cuanto a la configuración, hemos optado por el tercer método de instalación según la documentación oficial de **Wazuh**, mediante la cual se importa una **máquina virtual** (OVA) que hemos preparado para que esté lista para el despliegue del **HIDS**.

Esquema de la arquitectura de la aplicación **Wazuh** y decisiones de diseño

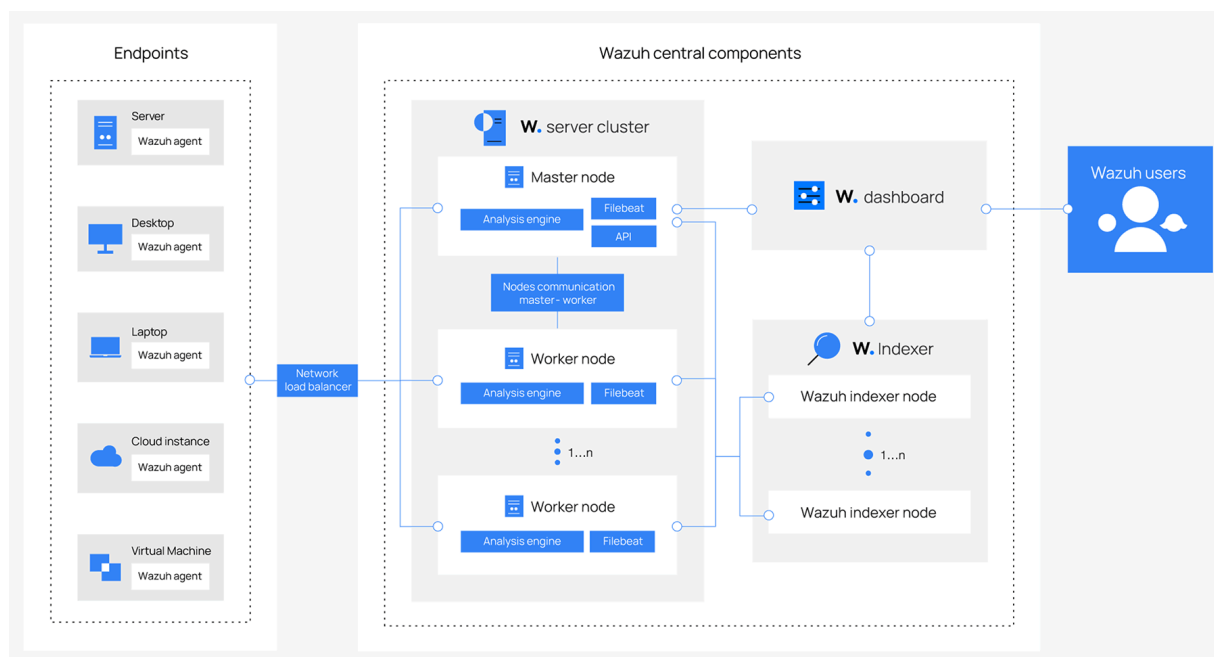


Figura 1. Esquema de la arquitectura de **Wazuh**, donde se relacionan los agentes clientes en la sección izquierda llamada *endpoints*, junto con los componentes centrales de la aplicación en el centro de la imagen, los cuales son un panel o *dashboard* donde visualizar lo que está ocurriendo con la integración, un *cluster* que realiza las tareas FIM que necesitamos en cada caso y un sistema de indexado en la parte lateral derecha que nos permite visualizar a través del dashboard el nodo en concreto del *cluster*.

Si comparamos la **Figura 1** con el esquema de la documentación del **PAI 1** (ver **Figura 2**) podemos encontrar un símil, el cual nos ha servido para decidirnos por *Wazuh* como sistema HIDS, de manera que hay una correlación directa entre los siguientes elementos:

Wazuh	ESQUEMA DEL PAI 1
Endpoints	Customer
Server cluster + Worker node	File Server + HIDS
Dashboard	Logs

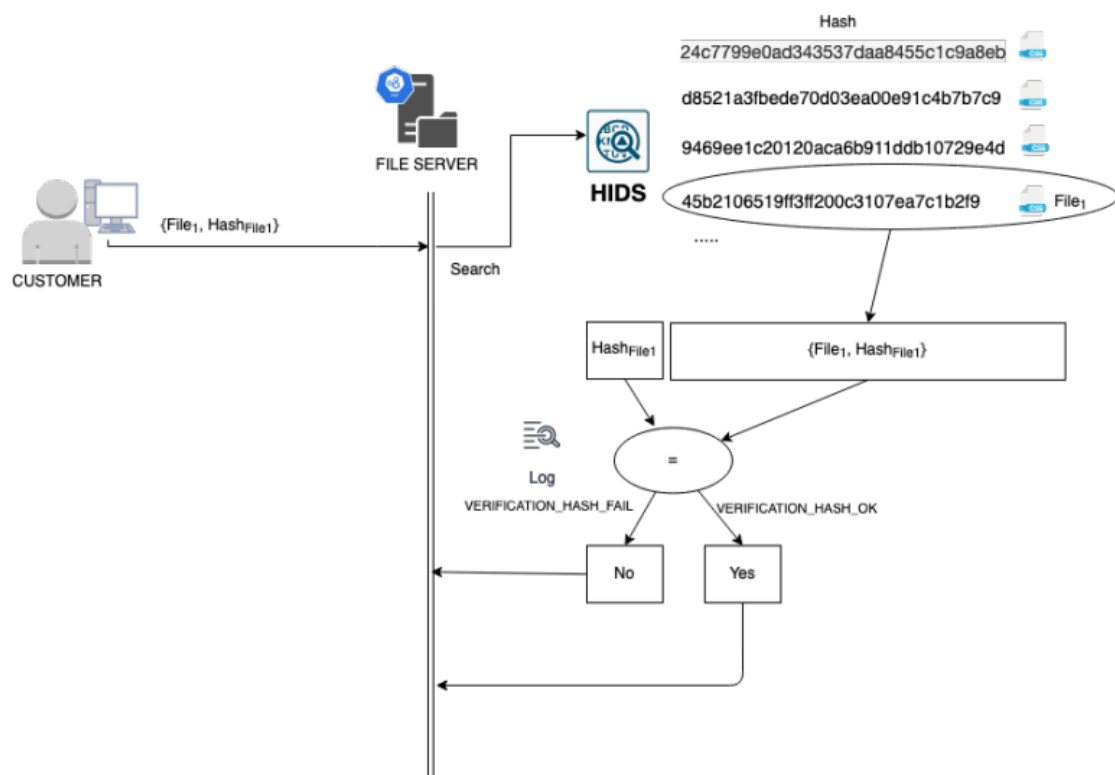


Figura 2. Esquema de la arquitectura del PAI 1, donde se relacionan los clientes en la sección izquierda llamados *CUSTOMER*, junto con los componentes centrales del esquema en el centro de la imagen, los cuales son los archivos a monitorear junto con el propio sistema de monitoreo (HIDS), el cual contempla una base de datos con los *hashings*. Abajo encontramos el sistema de logs el cual notifica cambios en la integridad de los ficheros y actúa en consecuencia.

Una vez comparadas ambas arquitecturas y viendo la semejanza entre la **Figura 1** y **Figura 2** hemos creado un servidor SMTP con Postfix en la máquina virtual para poder notificar a un correo determinado sobre los cambios que puedan ocurrir en el directorio especificado en el archivo de configuración anteriormente comentado.

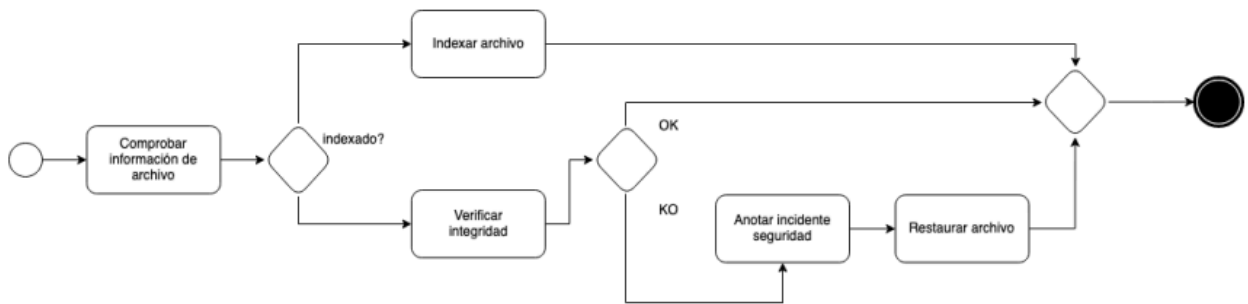


Figura 3. Modelo de procesos de monitoreo de la integridad de los archivos descrito en el documento del PAI 1.

Según el modelado de procesos que se ofrece en el PAI 1 (ver **Figura 3**), nosotros hemos cumplido con todos ellos.

La configuración de *Wazuh* se encuentra *hardcodeada* en su totalidad a través del archivo principal de configuración del agente *endpoint* o cliente, llamado ***ossec.conf*** el cual se encuentra en la ruta ***/var/ossec/etc/ossec.conf***. Esto nos permite siguiendo la documentación de la API oficial el poder gestionar los apartados más importantes del monitoreo y gestión de integridad que se piden, como por ejemplo: tipo/s de hash/es, frecuencia de monitoreo, directorio/s que se desean monitorear, gestión de avisos y alertas según los tipos de cambios de integridad en los ficheros, alertas por correo y gestión de logs, entre muchas otras opciones. De manera paralela, a través de este archivo se pueden realizar tareas relacionadas con la integridad del sistema operativo, como el control de rootkits y modificación de registros entre otras opciones.

1. Cálculo, almacenamiento y comprobación de hash

Método de hashing seguro:

Wazuh calcula los hashes de los archivos utilizando diferentes algoritmos criptográficos como *MD5*, *SHA1* y *SHA256*. El algoritmo se selecciona en la configuración del agente de *Wazuh*. Los hashes se calculan de la siguiente manera según el tamaño del archivo:

- Archivos completos: Se calcula el hash del archivo completo.
- Secciones de archivos: Se divide el archivo en secciones y se calcula el hash de cada sección.
- Archivos grandes: Se utiliza un algoritmo de hash incremental para calcular el hash sin necesidad de leer el archivo completo en la memoria.

Almacenamiento de Hash:

Los *hash* calculados se almacenan en el servidor de *Wazuh* que hace la función de unidad centralizada para organizar los distintos sistemas finales que haya conectados al servicio HIDS, adicionalmente, se puede configurar para que además se almacenen en una base de datos externa como *MySQL* o *SQLite*. La base de datos se puede configurar para que se actualice cada vez que se modifica un archivo o se instala un nuevo software o se puede actualizar cada cierto tiempo.

Comprobación de Hash:

Wazuh puede comprobar periódicamente la integridad de los archivos comparando sus hashes con los almacenados en la base de datos. Nuestro sistema HIDS está configurado para que haga comprobaciones **cada 12 horas** a través de las etiquetas **<frequency>** y además tenemos activado el parámetro **realtime** de las etiquetas **<directories>** para que haga una comprobación cada vez que el sistema modifique un archivo que se está monitoreando, esto último es útil para las pruebas pero puede suponer un gasto de procesamiento extra si se está planeando en trabajar sobre esos archivos.

Time	syscheck.path	syscheck.event	rule.description	rule.level	syscheck.sha256_after	syscheck.sha256_before
> Feb 28, 2024 @ 19:48:05.573	c:\users\juanl\one drive\escritorio\prueba_wazuh\nueva carpeta\apuntes.txt	modified	Integrity checksum changed.	7	55ca24497a8319767a5ea041e1fee89da14ee63d9dadceefd7e366ed685979ea	6ac17240db748763fee996dfe6c9f9b87f34ce3ccde84984ed1b7d4b6587893b
> Feb 28, 2024 @ 19:23:53.211	c:\users\juanl\one drive\escritorio\prueba_wazuh\nueva carpeta\apuntes.txt	modified	Integrity checksum changed.	7	6ac17240db748763fee996dfe6c9f9b87f34ce3ccde84984ed1b7d4b6587893b	49a59fc091770b2c57ade2d2fed2a961f6e50b39f4a6d253dd33f19244262fc
> Feb 28, 2024 @ 19:23:46.232	c:\users\juanl\one drive\escritorio\prueba_wazuh\nueva carpeta\apuntes.txt	added	File added to the system.	5	49a59fc091770b2c57ade2d2fed2a961f6e50b39f4a6d253dd33f19244262fc	-
> Feb 28, 2024 @ 19:23:46.188	c:\users\juanl\one drive\escritorio\prueba_wazuh\nueva carpeta\nuevo documento de texto.txt	deleted	File deleted.	7	49a59fc091770b2c57ade2d2fed2a961f6e50b39f4a6d253dd33f19244262fc	-

Figura 4. Lista con la información sobre los últimos archivos modificados.

2. Estructura para almacenar los Hash de forma eficiente:

La base de datos está optimizada para el almacenamiento y la recuperación eficiente de datos. La estructura de la base de datos es la siguiente:

- Tabla de archivos: Contiene el nombre del archivo, la ruta, el tamaño, el algoritmo de hash y el hash del archivo.
- Tabla de metadatos: Contiene información adicional sobre los archivos, como la fecha de creación, la fecha de modificación y el propietario del archivo.

3. Debilidades de los HIDS que *Wazuh* puede prevenir:

Las debilidades típicas del HIDS *Wazuh* incluyen:

- Desactivación de los HIDS de manera externa.
- La falta de Hash seguros, como sha1.
- Alarmas tardías, restauración del fichero y correos de aviso.

- Protección de la base de datos o información del HIDS.

Wazuh mitiga algunas de estas debilidades de la siguiente manera:

- El sistema HIDS de *Wazuh* permite elegir entre distintos tipos de Hash para el seguimiento de la integridad, con el más seguro siendo SHA256.
- *Wazuh* posee un subsistema de alertas basado en niveles de aviso que el sistema asigna un nivel a distintas acciones externas, por ejemplo, la modificación o eliminación de un archivo da una alerta de nivel 7, mientras que la adición de uno da una alerta de nivel 5. Este subsistema va de rango 1 a 16, se almacenan en el archivo `alerts.log` y se puede configurar a través de las etiquetas `<global>` o `<alerts>` para que dependiendo del nivel de la alerta el sistema envíe una alerta a una dirección de correo establecido.

```
Wazuh Notification.
2024 Feb 28 18:23:53

Received From: (mipc) 192.168.0.141->syscheck
Rule: 550 fired (level 7) -> "Integrity checksum changed."
Portion of the log(s):

File 'c:\users\juanl\onedrive\escritorio\prueba_wazuh\nueva carpeta\apuntes.txt' modified
Mode: realtime
Changed attributes: size,mtime,md5,sha1,sha256
Size changed from '21' to '18'
Old modification time was: '1709144572', now it is '1709144633'
Old md5sum was: 'e9f006e72a693a389776eefa3347ee59'
New md5sum is : '77fb12b14cce4338f6c6873946d22da6'
Old sha1sum was: 'fe3f288fdbdba90114dc3434c523ca51a85e3d63'
New sha1sum is : 'a609295021744033e92fd3fb5c758fc7f19ed697'
Old sha256sum was: '49a59fc091770b2c57ade2d2feed2a961f6e50b39f4a6d253dd33f19244262fc'
New sha256sum is : '6ac17240db748763fee996dfe6c9f9b87f34ce3ccde84984ed1b7d4b6587893b'

Attributes:
- Size: 18
- Permissions: SYSTEM (allowed):
DELETE|READ_CONTROL|WRITE_DAC|WRITE_OWNER|SYNCHRONIZE|READ_DATA|WRITE_DATA|APPEND_DATA|READ_EA|WRITE_EA|EXECUTE|READ_ATTRIBUTES|WRITE_ATTRIBUTES,
Administradores (allowed):
DELETE|READ_CONTROL|WRITE_DAC|WRITE_OWNER|SYNCHRONIZE|READ_DATA|WRITE_DATA|APPEND_DATA|READ_EA|WRITE_EA|EXECUTE|READ_ATTRIBUTES|WRITE_ATTRIBUTES, juanl
(allowed): DELETE|READ_CONTROL|WRITE_DAC|WRITE_OWNER|SYNCHRONIZE|READ_DATA|WRITE_DATA|APPEND_DATA|READ_EA|WRITE_EA|EXECUTE|READ_ATTRIBUTES|WRITE_ATTRIBUTES
- Date: Wed Feb 28 18:23:53 2024
- Inode: 0
- User: juanl (S-1-5-21-3930233709-1439467953-3995363694-1001)
- MD5: 77fb12b14cce4338f6c6873946d22da6
- SHA1: a609295021744033e92fd3fb5c758fc7f19ed697
- SHA256: 6ac17240db748763fee996dfe6c9f9b87f34ce3ccde84984ed1b7d4b6587893b
- File attributes: ARCHIVE

What changed:
< deFGETGRHJFGHKTRGSDKH
---
> Esto es una prueba
```

Figura 5. Email enviado por el sistema después de que un archivo haya sido modificado.

- En cuanto al sistema HIDS en sí, *Wazuh* posee la opción de crear *backups* o copias de seguridad de información y configuración del HIDS tanto del servidor como de los agentes activos de manera que se pueden restaurar en caso de emergencia.

4. Fichero de configuración

Gracias al entorno virtual con la OVA que obtenemos de *Wazuh*, obtenemos la última versión de *Wazuh* y el fichero de configuración preparado para el uso inmediato del servidor. La configuración que hemos tenido que aportar al agente es la siguiente:

- Selección de los directorios a monitorizar, esto está indicado en la etiqueta **<directories>** donde además añadimos configuración extra sobre el seguimiento del directorio: *check_sha256sum* le comunicamos que nos haga los hash en sha256, *report_changes* para que se registren los cambios en los logs, *realtime* para que se monitoree la actividad a tiempo real y *recursion_level* le comunica a Wazuh a cuantos niveles del directorio tiene permitido acceder para monitorizar archivos.

```
<!-- File integrity monitoring -->
<syscheck>

  <disabled>no</disabled>

  <!-- Frequency that syscheck is executed default every 12 hours -->
  <frequency>43200</frequency>

  <directories check_sha256sum="yes" report_changes="yes" realtime="yes" recursion_level="5">C:\Users\
```

Figura 6. Configuración del agente.

- Para el servidor añadimos las líneas necesarias con la etiqueta **<localfile>** para que nos registrara el directorio a monitorizar.
- Adicionalmente, se ha creado un servidor SMTP con postfix para que se puedan enviar los correos de alerta al correo seleccionado en la etiqueta **<email_to>** de **<global>**. El servidor SMTP está configurado y se activa automáticamente al iniciar el servidor.
- Por último se ha comentado el análisis de archivos del sistema de Windows en el agente para mayor facilidad de revisión.

5. Logs de actualización diaria

Wazuh tiene la función predefinida de enviar al correo seleccionado los logs con el contenido que se desee. Es a través de la etiqueta **<reports>**.

```
<reports>
  <category>syscheck</category>
  <title>Reporte diario: cambios en los archivos</title>
  <email_to>juaruamur@alum.us.es</email_to>
</reports>
```

Figura 7. Configuración de los logs diarios.

La etiqueta **<category>** te permite seleccionar qué tipos de log queremos que nos muestre, en este ejemplo se ha definido *syscheck* que son las modificaciones de los archivos monitoreados y el correo seleccionado es de un integrante del grupo, deberá cambiar el correo al suyo.

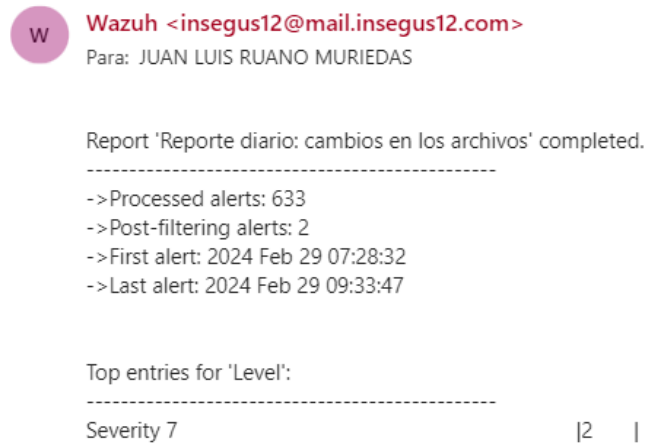


Figura 8. Captura de un apartado del reporte diario de Wazuh.

6. Informe mensual

El dashboard de *Wazuh* permite crear informes a voluntad sobre la integridad de los archivos monitoreados por el HIDS, para ello solo tiene que ir al dashboard e ir al agente que desea hacer el reporte, elegir el intervalo de tiempo, en este caso 30 días y automáticamente se creará un reporte de los últimos 30 días.

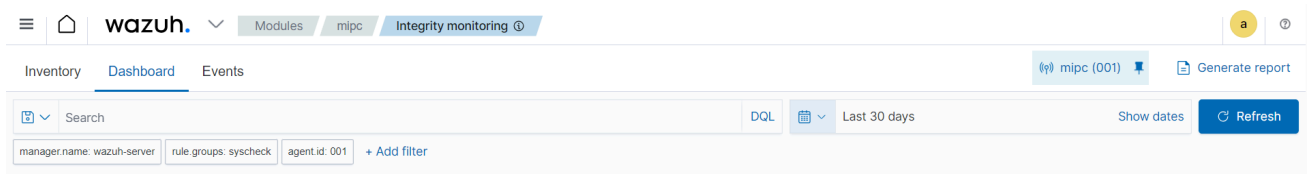


Figura 9. Navegador para generar el reporte.

Ahora en la pestaña que hay al lado del icono *Wazuh*, abra el desplegable y vaya al apartado *Management > Reporting* para descargar el reporte.

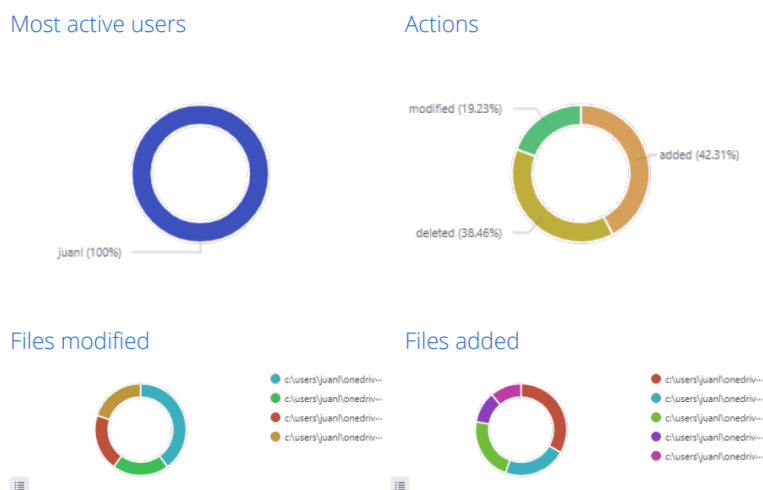


Figura 10. Reporte de los archivos que han sufrido cambios en los últimos 30 días.

7. Pruebas de archivos



File	Last Modified 	User 	User ID	Group	Group ID	Size
c:\users\juan\onedrive\escritorio\prueba_wazuh\burndown.xlsx	Dec 13, 2023 @ 19:52:11.000	juanl	S-1-5-21-393023...			16764
c:\users\juan\onedrive\escritorio\prueba_wazuh\conversión de datos.ipynb	Jan 5, 2022 @ 13:48:31.000	juanl	S-1-5-21-393023...			10900
c:\users\juan\onedrive\escritorio\prueba_wazuh\descarga.jfif	Oct 14, 2021 @ 12:09:16.000	juanl	S-1-5-21-393023...			613857
c:\users\juan\onedrive\escritorio\prueba_wazuh\etsimarkt1.mkv	Dec 13, 2023 @ 21:45:48.000	juanl	S-1-5-21-393023...			213514679
c:\users\juan\onedrive\escritorio\prueba_wazuh\fraudes por internet.csv	Dec 22, 2023 @ 17:49:30.000	juanl	S-1-5-21-393023...			5751
c:\users\juan\onedrive\escritorio\prueba_wazuh\index.html	Feb 25, 2024 @ 13:17:46.000	juanl	S-1-5-21-393023...			3297
c:\users\juan\onedrive\escritorio\prueba_wazuh\jdk-15.0.1_windows-x64_bin.exe	Oct 21, 2020 @ 15:47:49.000	juanl	S-1-5-21-393023...			167452312

Figura 11. Captura de los archivos monitoreados localmente en Wazuh.

En las pruebas con el HIDS de *Wazuh*, se han monitoreado con un conjunto de archivos heterogéneos como son .mp3, .html, .exe, .pdf, .csv, .txt, etc. Y con tamaños de máximo 2,56 GB y una cantidad de 50 archivos. La única excepción que el HIDS no puede registrar son archivos .bin, los archivos binarios no se pueden hacer seguimiento por seguridad. Se han hecho pruebas de creación, modificación y eliminación de ficheros y cada uno hace saltar una alarma de diferentes niveles según la configuración de *Wazuh*.

Bibliografía

- Comunidad Hacking Ético. (2023, Julio 21). *Desplegando #Wazuh - Herramienta Esencial para la detección de intrusos*. YouTube. Retrieved February 25, 2024, from https://www.youtube.com/watch?v=kFopnSGSduE&ab_channel=ComunidadHacking%C3%89tico
- Oracle. (n.d.). Oracle VM VirtualBox. Retrieved February 20, 2024, from <https://www.virtualbox.org/>
- Wazuh. (n.d.). *Enrollment via agent configuration - Wazuh agent enrollment*. Wazuh documentation. Retrieved February 28, 2024, from <https://documentation.wazuh.com/current/user-manual/agent-enrollment/via-agent-configuration/index.html>
- Wazuh. (n.d.). *Virtual Machine (OVA) - Installation alternatives*. Wazuh documentation. Retrieved February 28, 2024, from <https://documentation.wazuh.com/current/deployment-options/virtual-machine/virtual-machine.html>