

# **Proyecto de Laboratorio: Monitoreo y Control con Wazuh + pfSense**

**wazuh.**



# Contenido

- Introducción.....3**
  - ¿Por qué es importante?.....3
  - ¿Qué se abordará en este laboratorio?..... 3
- Desarrollo.....4**
  - Infraestructura..... 4
  - Configuración de Herramientas.....4
    - Wazuh – File Integrity Monitoring (FIM).....4
    - Configuración de reglas en el firewall pfSense..... 7
- Conclusión.....8**

# Introducción

En un escenario donde las amenazas cibernéticas evolucionan constantemente, incluso las organizaciones más pequeñas necesitan implementar medidas de protección efectivas. Este laboratorio técnico simulado propone una solución accesible y escalable para comenzar a construir una infraestructura segura desde cero, haciendo uso de tecnologías **open source** como **Wazuh** y **pfSense**.

A lo largo de este entorno práctico se trabajan los pilares esenciales de la seguridad informática, tales como:

- la **segmentación de redes** para reducir la superficie de ataque,
- la **protección de archivos críticos** mediante monitoreo de integridad,
- el **seguimiento continuo de eventos**,
- la **gestión y configuración de firewalls**,
- el uso de **agentes de SIEM** para detección proactiva,
- y la **virtualización de entornos cliente-servidor** para simular escenarios reales.

Pero esto va más allá de la práctica técnica: demuestra cómo, con recursos mínimos, es posible sentar las bases de un sistema alineado a buenas prácticas y estándares internacionales.

## ¿Por qué es importante?

Porque estas herramientas no solo fortalecen la postura defensiva frente a incidentes, sino que permiten avanzar hacia el **cumplimiento normativo** (como la Ley 25.326 o la ISO 27001), incluso en entornos con limitaciones presupuestarias como una PyME.

## ¿Qué se abordará en este laboratorio?

- **Monitoreo en tiempo real** de archivos sensibles con Wazuh (FIM)
- **Alertas automáticas** por creación, modificación o eliminación de archivos
- **Firewall segmentado y personalizado** para restringir protocolos no deseados
- **Topología virtualizada** escalable para próximos desafíos

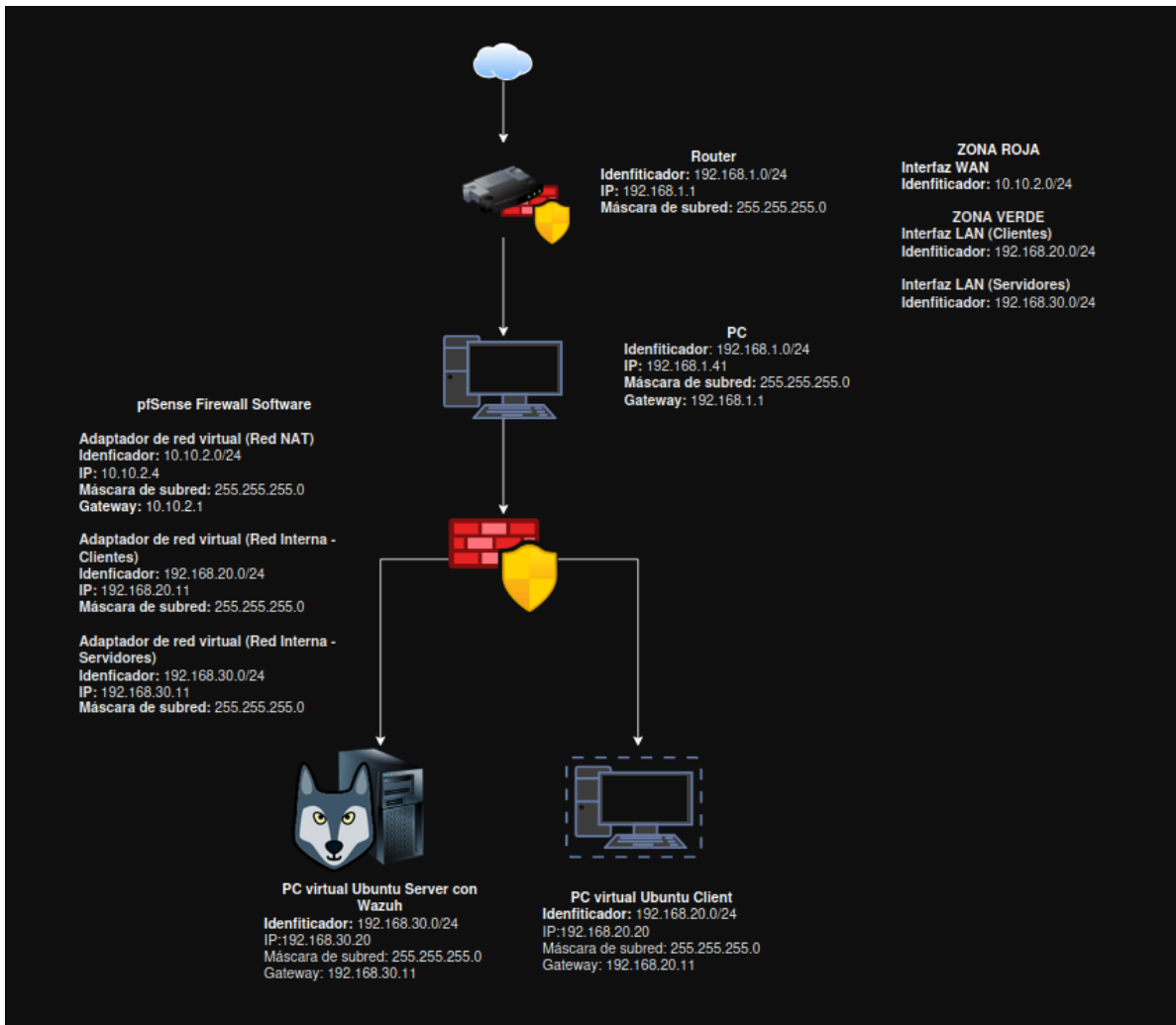
Este es solo el primer paso. En futuras entregas se incorporarán nuevas técnicas defensivas y ofensivas, simulaciones basadas en **MITRE ATT&CK**, y análisis de eventos complejos con múltiples vectores.

# Desarrollo

## Infraestructura

La infraestructura del laboratorio fue creada en un entorno virtualizado con **VMware** y está compuesta por:

- **Ubuntu Server 22.04** – Servidor con Wazuh
- **pfSense** – Firewall central con reglas configuradas
- **Ubuntu Client 22.04** – Cliente con el agente de Wazuh instalado



## Configuración de Herramientas

### Wazuh – File Integrity Monitoring (FIM)

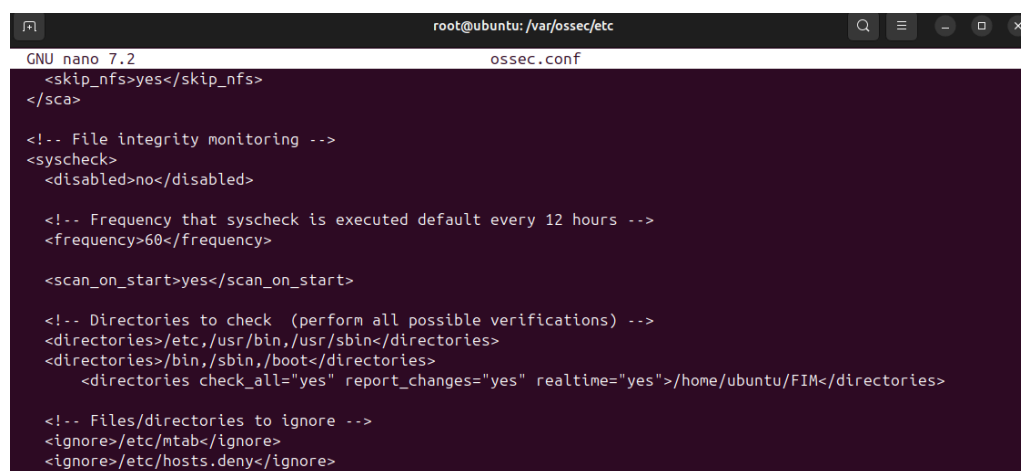
Se implementó **Wazuh** como solución de monitoreo de integridad de archivos (FIM), instalando el servidor en una máquina con **Ubuntu Server 22.04** y configurando un **agente** en un cliente con **Ubuntu Desktop 22.04**.

## Pasos principales:

1. Instalación de Wazuh desde su **repositorio oficial** en el servidor.
2. Creación del directorio **/home/ubuntu/FIM/** en el cliente, como carpeta de prueba para la detección de cambios.
3. Instalación del agente de Wazuh en el cliente.
4. Configuración del archivo **ossec.conf** (ubicado en **/var/ossec/etc/**) para que el agente monitoree la carpeta mencionada.

## ¿Qué es ossec.conf?

Es el archivo principal de configuración del agente de Wazuh. Define qué módulos están activos, qué directorios observar, cómo enviar logs y a qué servidor conectarse. En este laboratorio, el foco estuvo puesto en el módulo **syscheck**, encargado de detectar alteraciones en archivos críticos.



```
root@ubuntu: /var/ossec/etc
GNU nano 7.2 ossec.conf
<skip_nfs>yes</skip_nfs>
</sca>

<!-- File integrity monitoring -->
<syscheck>
  <disabled>no</disabled>

  <!-- Frequency that syscheck is executed default every 12 hours -->
  <frequency>60</frequency>

  <scan_on_start>yes</scan_on_start>

  <!-- Directories to check (perform all possible verifications) -->
  <directories>/etc,/usr/bin,/usr/sbin</directories>
  <directories>/bin,/sbin,/boot</directories>
  <directories check_all="yes" report_changes="yes" realtime="yes">/home/ubuntu/FIM</directories>

  <!-- Files/directories to ignore -->
  <ignore>/etc/mtab</ignore>
  <ignore>/etc/hosts.deny</ignore>
```

## Cambios realizados:

### 1. **<frequency>60</frequency>**

Indica que Wazuh ejecutará un análisis completo del FIM cada 60 segundos (antes eran 12 horas = 43200 segundos por defecto).

Esto es útil en ambientes de prueba donde queremos ver cambios rápidamente, pero no recomendado en producción porque puede generar mucha carga y eventos.

### 2. **<directories check\_all="yes" report\_changes="yes" realtime="yes">/home/ubuntu/FIM</directories>**

Indica a Wazuh que debe monitorear el directorio **/home/ubuntu/FIM** con los siguientes parámetros:

***check\_all="yes"***

Hace todas las verificaciones posibles sobre los archivos, como hashes (MD5/SHA1), permisos, propietario, tamaño, fecha de modificación, etc.

***report\_changes="yes"***

Si un archivo cambia, Wazuh intenta generar un diff (diferencia) entre el contenido anterior y el nuevo (si es texto). Es útil para ver exactamente qué cambió.

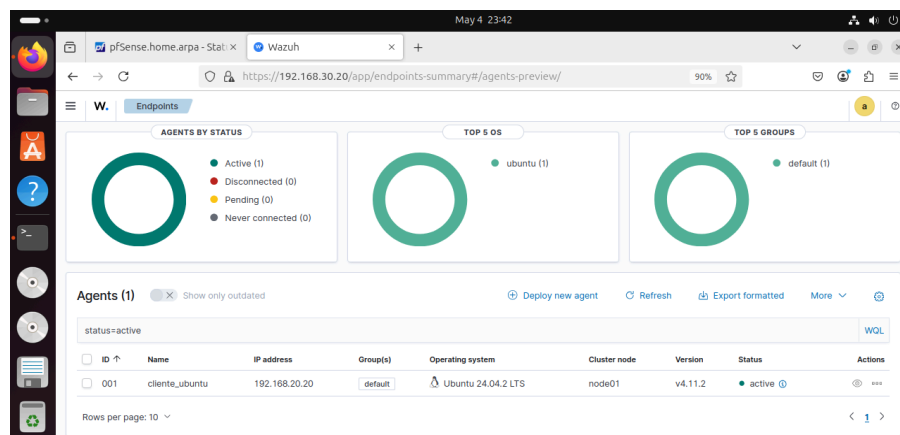
***realtime="yes"***

En lugar de esperar al análisis programado, Wazuh utiliza inotify (en Linux) para detectar cambios en tiempo real. Apenas un archivo cambia, se reporta el evento.

Tras aplicar estos ajustes, se reinició el agente con:

***systemctl restart wazuh-agent***

Volvemos al dashboard de Wazuh y comprobamos que detecte el activo que acabamos de agregar.



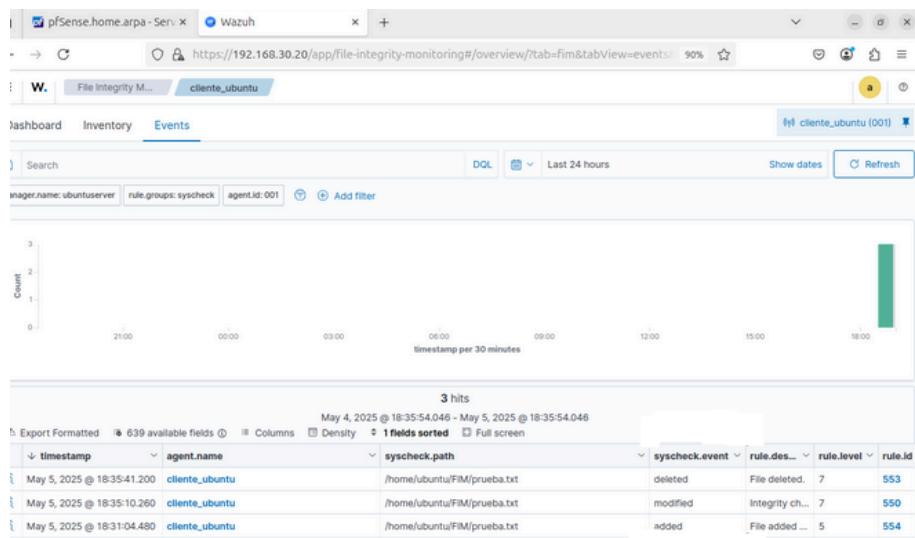
Con estos pasos realizados el File Integrity Monitoring (FIM) se encuentra activo para monitorear cambios en el directorio /home/Ubuntu/FIM/

### Pruebas realizadas en el directorio /home/Ubuntu/FIM/

- Se creó un archivo (prueba.txt) → alerta “added”.
- Se modificó el contenido → alerta “modified”.
- Se eliminó el archivo prueba.txt → alerta “deleted”.

```
ubuntu@ubuntu: ~/FIM
ubuntu@ubuntu:~/Desktop$ cd ..
ubuntu@ubuntu:~$ ls
Desktop  Downloads  Music      Public     Templates
Documents FIM        Pictures   snap       Videos
ubuntu@ubuntu:~$ cd FIM
ubuntu@ubuntu:~/FIM$ touch prueba.txt
ubuntu@ubuntu:~/FIM$
```

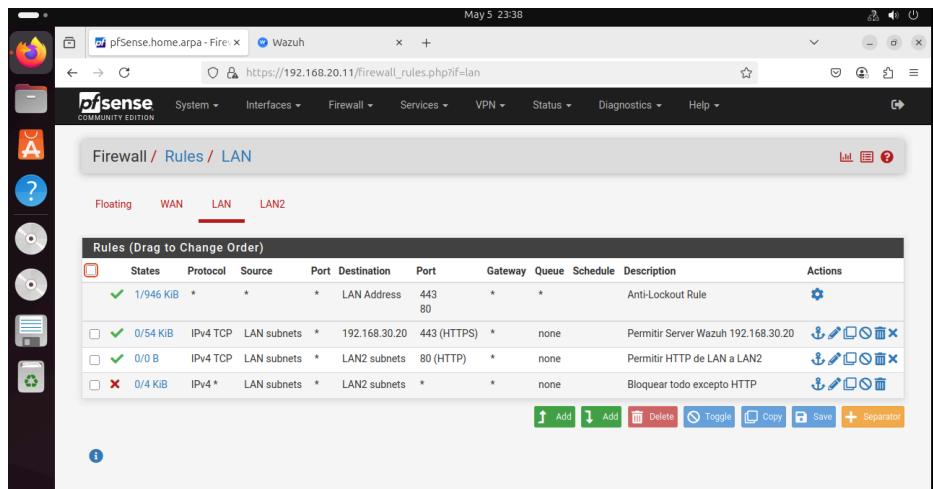
Verificamos que las alertas lleguen con el resultado esperado



### Configuración de reglas en el firewall pfSense

Se configuraron tres reglas en el firewall pfSense ubicado entre el cliente y el servidor, con el objetivo de permitir únicamente el tráfico HTTP y bloquear todo el resto del tráfico:

- **Regla 1 – Acceso al Dashboard de Wazuh (HTTPS):** Se permite el tráfico saliente desde la interfaz de clientes hacia el servidor Wazuh mediante el puerto 443 (HTTPS), con el fin de garantizar el acceso al panel de monitoreo y gestión.
- **Regla 2 – Tráfico HTTP:** Se habilita el tráfico HTTP (puerto 80) desde la red LAN hacia la interfaz LAN2, permitiendo únicamente este tipo de comunicación entre ambas redes.
- **Regla 3 – Denegación por defecto:** Se establece una regla de denegación para bloquear todo el tráfico no especificado anteriormente, asegurando que únicamente las conexiones HTTP (y HTTPS hacia Wazuh) estén permitidas.



## Conclusión

El desarrollo de este informe permitió integrar conocimientos teóricos y prácticos en torno a la seguridad de la información, demostrando cómo una PyME puede aplicar marcos legales y técnicos para proteger sus activos digitales. La implementación de controles como FIM con Wazuh y reglas específicas en pfSense no solo fortalece la seguridad operativa, sino que también facilita el cumplimiento con normativas vigentes como la Ley 25.326 e ISO/IEC 27001. Esto refuerza la importancia de adoptar un enfoque sistemático y normativo para gestionar los riesgos cibernéticos de manera efectiva.