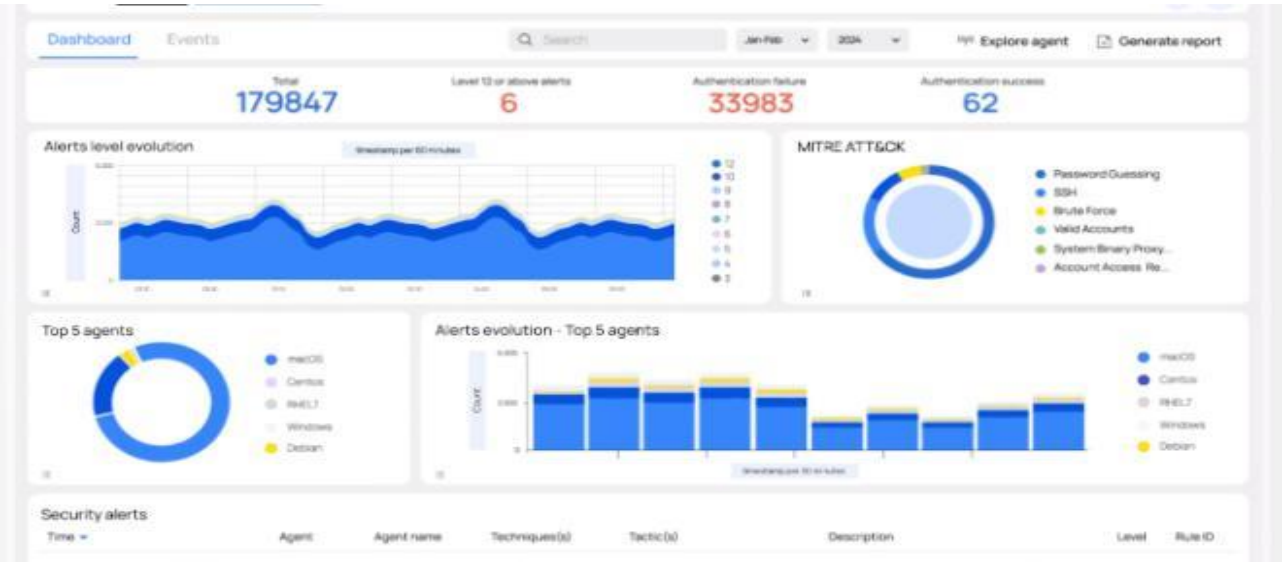

■ Cómo Crear un Centro de Operaciones de Seguridad (SOC, por sus siglas en inglés) con herramientas Open Source.

1. Planificación del SOC.

Antes de implementar, define:

- Objetivo del SOC:

¿Detección de amenazas? ¿Respuesta a incidentes?

- Infraestructura existente: Servidores, endpoints, redes.

- Equipo:

¿Quién gestionará el SOC? ¿Cuántas personas necesitas?

- Alcance:

¿Cuáles son los activos críticos que quieres proteger?

2. Herramientas Open Source clave para un SOC.

A. Monitoreo de registros (SIEM).

- Herramienta recomendada: [“Wazuh”]

(https://wazuh.com/)

- Wazuh es una plataforma gratuita que actúa como un SIEM, HIDS y gestor de cumplimiento normativo.

- Centraliza logs de diferentes fuentes, detecta amenazas y genera alertas.

- Complemento: Integra Wazuh con “Elastic Stack (ELK)” para análisis avanzado de datos.

B. Gestión de amenazas e inteligencia.

- Herramienta recomendada: [“MISP” (Malware Information Sharing Platform)]

(https://www.misp-project.org/)

- Comparte información sobre amenazas e indicadores de compromiso (IoCs).

- Útil para trabajar con equipos de respuesta a incidentes.

C. Detección de intrusiones (IDS/IPS).

- Herramienta recomendada:

[“Suricata”](https://suricata.io/) o [“Snort”]

(https://www.snort.org/)

- Sistemas de detección y prevención de intrusiones que analizan tráfico de red en tiempo real.

- Suricata es más flexible y puede integrarse con Elastic Stack para visualización.

D. Monitoreo de red.

- Herramienta recomendada: [“Zeek”]

(antes Bro)](https://zeek.org/)

- Proporciona análisis detallado del tráfico de red y captura eventos relevantes para investigación.

E. Análisis de datos y visualización.

- Herramienta recomendada: [“Kibana”]

(https://www.elastic.co/kibana)

- Parte de Elastic Stack; permite crear dashboards para visualización de alertas y registros.

F. Gestión de tickets y automatización.

- Herramienta recomendada: [“TheHive”]

(https://thehive-project.org/)

  - Plataforma para gestionar casos de incidentes.

  - Complemento: “Cortex” para la automatización del análisis de IoCs.

G. Recolección y análisis forense.

- Herramienta recomendada: [“Volatility”]

(https://www.volatilityfoundation.org/)

  - Para análisis forense de memoria.

- Complemento: [“Autopsy”]

(https://www.sleuthkit.org/autopsy/) para análisis de disco.

3. Diseño del SOC.

Arquitectura básica:

1. Recolección de datos:

   - Logs de dispositivos (firewalls, endpoints, servidores) usando agentes (como los de Wazuh).

2. Procesamiento y análisis:

   - Elastic Stack para procesar los logs.

   - Suricata y Zeek para monitoreo de red.

3. Visualización:

   - Dashboards personalizados en Kibana o Grafana.

4. Respuesta a incidentes:

   - TheHive para gestionar incidentes.

   - Playbooks para automatización de respuesta.

4. Implementación práctica.

1. Configura un servidor central:

Puede ser en un entorno on-premises o en la nube (ej., AWS, Azure).

2. Instala y configura las herramientas:

Empieza con Wazuh y Elastic Stack.

3. Crea reglas de correlación:

Define qué alertas son críticas para tu entorno.

4. Establece un flujo de trabajo:

Define cómo se manejarán las alertas (detección, escalamiento, resolución).

5. Prueba el SOC:

Realiza simulaciones de ataques para validar el sistema.

5. Capacitación y mejoras continuas.

Un SOC efectivo necesita:

- Capacitación regular del personal.

- Actualización constante de herramientas para protegerse de nuevas amenazas.

- Colaboración con comunidades Open Source para compartir información y mejoras.

Voy a empezar configurando un entorno básico de SOC utilizando “Wazuh” como la herramienta central para recolección y análisis de logs.

Este sistema es escalable y se puede integrar con Elastic Stack para un análisis más avanzado.

1. Requisitos previos.

Antes de empezar, asegúrate de tener:

1. Servidor con Linux (Ubuntu 20.04 o superior, CentOS 7/8, o Debian 10/11).

2. Acceso de administrador (root) o un usuario con privilegios sudo.

3. Recursos mínimos recomendados:

- CPU: 4 cores.
- RAM: 8 GB (para Elastic Stack).
- Disco: 50 GB o más (depende de los logs que recolectarás).

2. Instalación de Wazuh.

Paso 1: Instalar Wazuh Manager.

1. Actualiza el sistema:

```bash
sudo apt update && sudo apt upgrade -y
```

2. Añade el repositorio oficial de Wazuh:

```bash
curl -s https://packages.wazuh.com/key/GPG-KEY-WAZUH | sudo apt-key add -
echo "deb https://packages.wazuh.com/4.x/apt/ stable main" | sudo tee /etc/apt/sources.list.d/wazuh.list
sudo apt update
```

3. Instala el Wazuh Manager:

```bash
sudo apt install wazuh-manager -y
```

4. Inicia y verifica el servicio:

```bash
sudo systemctl enable wazuh-manager
sudo systemctl start wazuh-manager
sudo systemctl status wazuh-manager
```

Paso 2: Configurar Wazuh API.

1. Instala Wazuh API:

```bash
sudo apt install wazuh-api -y
```

2. Configura el API:

- Edita el archivo `/var/ossec/api/configuration/auth/auth.js` para definir usuarios y contraseñas.

- Reinicia el servicio:

```bash
sudo systemctl restart wazuh-api
```

Paso 3: Instalar Elastic Stack.

Si quieres visualizar los datos en dashboards más avanzados, debes integrar Wazuh con Elastic Stack.

1. Instala Elasticsearch:

```bash
curl -fsSL https://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo gpg --dearmor -o /usr/share/keyrings/elasticsearch-keyring.gpg
echo "deb [signed-by=/usr/share/keyrings/elasticsearch-keyring.gpg] https://artifacts.elastic.co/packages/8.x/apt stable main" | sudo tee /etc/apt/sources.list.d/elastic-8.x.list
sudo apt update && sudo apt install elasticsearch -y
```

2. Configura Elasticsearch:

- Edita el archivo `/etc/elasticsearch/elasticsearch.yml` para ajustar el `cluster.name` y `network.host` (puedes usar `0.0.0.0` para aceptar conexiones externas).

- Inicia el servicio:

```bash
sudo systemctl enable elasticsearch
sudo systemctl start elasticsearch
```

3. Instala Kibana:

```bash
sudo apt install kibana -y
sudo systemctl enable kibana
sudo systemctl start kibana
```

4. Integra Wazuh con Elastic Stack siguiendo la guía oficial de Wazuh [aquí]

(https://documentation.wazuh.com/current/installation-guide/integrations/elastic-stack.html).

Paso 4: Configurar agentes (opcional para monitorear endpoints).

1. Instala el agente de Wazuh en los dispositivos que quieras monitorear:

- Linux:

```bash
curl -sO https://packages.wazuh.com/4.x/apt/wazuh-agent_4.4.0-1_amd64.deb
sudo dpkg -i wazuh-agent_4.4.0-1_amd64.deb
```

- Windows: Descarga e instala el agente desde [el sitio oficial]

(https://documentation.wazuh.com/current/installation-guide/wazuh-agent/windows/index.html).

2. Configura el agente para que apunte a tu Wazuh Manager editando el archivo `/var/ossec/etc/ossec.conf` (Linux) o `ossec.conf` (Windows).

3. Inicia el agente:

```bash
sudo systemctl enable wazuh-agent
sudo systemctl start wazuh-agent
```

3. Validación y pruebas.

1. Accede a la interfaz de Kibana en

`http://<IP-del-servidor>:5601` y verifica que los logs de Wazuh estén siendo recolectados.

2. Prueba la generación de alertas creando un archivo sospechoso en algún endpoint:

```bash
touch /etc/test-malware
```

4. Extensiones adicionales.

- Automatización:

Integra “TheHive” para la gestión de incidentes.

- Inteligencia de amenazas:

Configura “MISP” para importar IoCs relevantes.

MANUEL NAVARRO HIDALGO.