LOGS CRÍTICOS PARA MONITOREAR: UNA GUÍA PARA SOC ANALISTAS

Tabla de contenidos

1. *Introducción*

Los logs son las huellas de cada actividad digital, sirviendo como un registro cronológico de eventos dentro de sistemas, redes y aplicaciones.

En un Centro de Operaciones de Seguridad (SOC) moderno, los analistas confían en estos logs para detectar amenazas, investigar incidentes de seguridad y mantener la postura de seguridad general de una organización. Sin logs estructurados y bien supervisados, incluso las soluciones de seguridad más avanzadas pueden pasar por alto indicadores críticos de compromiso (IoC) o no correlacionar comportamientos sospechosos en varios sistemas. El objetivo de esta guía es destacar qué logs son más importantes, por qué son esenciales y cómo abordar su supervisión de una manera que beneficie tanto a los analistas de SOC junior como a los de nivel medio.

1. *IMPORTANCIA DE LA SUPERVISIÓN DE LOGS EN SOC*

En cualquier infraestructura de TI de gran tamaño, los volúmenes de datos sin procesar pueden ser enormes: los firewalls por sí solos pueden generar miles de entradas de registro por segundo. Si bien estos logs a veces pueden parecer líneas de texto sin importancia, contienen información valiosa que ayuda a detectar y contrarrestar las amenazas de seguridad. La supervisión adecuada de los logs es crucial por varias razones:

1. ***Visibilidad y contexto***
   Los logs proporcionan contexto al mostrar lo que sucedió, cuándo sucedió y cómo se ejecutó. Esta visibilidad es esencial para distinguir los comportamientos normales de las anomalías. Por ejemplo, una escalada de privilegios inesperada en un registro del sistema de Windows puede apuntar a un movimiento lateral por parte de un atacante. Del mismo modo, los errores de autenticación repetidos en un entorno Linux pueden indicar un ataque de fuerza bruta.

2. ***Detección y respuesta a incidentes***
   Las alertas automatizadas de una plataforma SIEM (Security Information and Event Management) a menudo se originan a partir de patrones sospechosos en los logs. Estas alertas permiten a los analistas de SOC identificar y responder rápidamente a posibles incidentes. Por ejemplo, las reglas de correlación pueden marcar a un usuario que inicia sesión desde dos ubicaciones geográficamente distantes en un corto período de tiempo, lo que sugiere una contraseña robada.

3. ***Auditoría y Cumplimiento***
   Muchos marcos regulatorios, como PCI DSS, HIPAA o ISO 27001, exigen la retención de logs y la revisión periódica. Al monitorear los logs, las organizaciones se aseguran de cumplir con los requisitos de cumplimiento y pueden producir una pista de auditoría clara durante las investigaciones o auditorías. Los logs suelen ser el primer lugar que los auditores comprueban

para confirmar que los controles de seguridad están en su lugar y funcionan según lo previsto.

4. ***Caza de amenazas***
   Más allá de la detección, los logs forman la base para la búsqueda proactiva de amenazas. Los analistas buscan patrones inusuales, como la ejecución de un script de PowerShell en un entorno donde el uso de PowerShell es raro, para descubrir ataques sigilosos. Al analizar los logs a lo largo del tiempo, los cazadores de amenazas pueden identificar tendencias y tácticas de adversarios que podrían pasar desapercibidas solo por los sistemas automatizados.

5. ***Investigaciones Forenses***
   Cuando se produce un incidente, los logs bien estructurados son la clave de las investigaciones forenses. Ayudan a recrear la línea de tiempo de un ataque, muestran a qué sistemas se accedió y resaltan los datos que se filtraron. Logs detallados de las acciones del usuario, red conexiones y las llamadas al sistema pueden ser la diferencia entre atribuir con precisión un incidente y permitir que los atacantes permanezcan sin ser detectados.

Ejemplo del mundo real

*Considere un escenario en el que un analista de SOC detecta tráfico saliente inusual desde un servidor crítico. Al revisar **los logs de firewall** correlacionados con los **logs de eventos de Windows**, el analista descubre un proceso malicioso que se comunica con una dirección IP externa. El análisis de Quick muestra que la comunicación comenzó justo después de un evento sospechoso de escalada de privilegios. Esta correlación puede guiar a los equipos de respuesta a incidentes para aislar el servidor, contener la amenaza y remediar la vulnerabilidad antes de que los datos se vean comprometidos.*

Consejo práctico

*En los sistemas Linux, comandos como **journalctl -p warning -r** pueden ayudarlo a localizar rápidamente eventos de mayor prioridad en orden cronológico inverso, lo que permite una clasificación más rápida de posibles problemas de seguridad.*

*En Windows, herramientas como **wevtutil qe Security /rd:true /f:text /q:"*" |findstr /i "4624 4625 4634 4672"** pueden filtrar el registro de eventos de seguridad para identificadores de eventos específicos relacionados con inicios de sesión.*

1.2. Alcance y finalidad de la Guía

Esta guía está dirigida tanto a los analistas de SOC nuevos como a los de nivel medio que desean mejorar sus habilidades en la supervisión de logs. Cubre las fuentes comunes de logs, como los sistemas operativos, las redes, las aplicaciones y las herramientas de seguridad, y destaca lo que se debe buscar en

cada una. Al centrarse en los logs más críticos y describir cómo encajan en la estrategia de seguridad general, esta guía tiene como objetivo agilizar el trabajo diario de los profesionales de SOC. En concreto, se pretende:

- ***Identificación de orígenes de logs clave***
  Analizaremos ***los logs del sistema*** *(Windows, Linux, macOS),* ***los logs de red*** *(firewall, enrutador, IDS / IPS),* ***los logs de aplicaciones y bases de datos****,* ***los logs de seguridad*** *(AV, EDR, XDR),* ***los logs en la nube*** *(AWS, Azure, GCP), los logs de contenedores (Docker, Kubernetes) y* ***los logs de IoT / SCADA / OT.*** *El enfoque principal es qué hace que cada categoría sea crítica, cómo recopilarlas y qué eventos son más indicativos de un problema de seguridad.*

- ***Presentar Técnicas Prácticas de Monitoreo***
  Desde las reglas de correlación hasta la detección basada en anomalías, analizaremos las prácticas que traducen los logs sin procesar en información procesable. También trataremos temas como ***la retención de*** *logs,* ***la seguridad de los logs*** *y las mejores prácticas en torno* ***a la clasificación de datos*** *para garantizar que los logs confidenciales permanezcan protegidos.*

- ***Mostrar casos de uso de la vida real***
  Cada tipo de registro viene con su conjunto único de desafíos y vectores de ataque. Analizaremos escenarios realistas, como la detección de movimiento lateral, escalada de privilegios o cargas maliciosas, y demostraremos cómo los logs sirven como evidencia vital.

- ***Guía sobre herramientas de apoyo***
  Las herramientas SIEM (Security Information and Event Management) y SOAR (Security Orchestration, Automation, and Response) son el núcleo de los SOC modernos. La guía explica cómo estas plataformas se integran con diferentes fuentes de registro, automatizan las alertas y ayudan a orquestar las acciones de respuesta.

- ***Fomentar el aprendizaje continuo***
  Las amenazas cibernéticas evolucionan rápidamente, al igual que las mejores prácticas de registro y monitoreo. Con referencias a recursos como NIST SP 800-92 (Guía para la administración de logs de seguridad informática) y documentación oficial de proveedores (por ejemplo, la documentación del registro de eventos de Windows de Microsoft), esta guía dirige a los lectores a fuentes confiables para la educación continua.

Al centrarse en estas áreas, la guía tiene como objetivo equipar a los analistas con el conocimiento y las habilidades para priorizar los logs de manera efectiva y detectar posibles infracciones antes de que se intensifiquen. A través de una combinación de explicaciones teóricas y ejemplos prácticos, los lectores ganarán confianza para configurar estrategias de registro, ajustar alertas y realizar investigaciones exhaustivas.

2. *TIPOS CLAVE DE LOGS*

1. *Logs del sistema (Windows, Linux, macOS)*

Los logs del sistema forman la columna vertebral de los esfuerzos de detección y respuesta a incidentes, proporcionando a los analistas los datos de referencia esenciales necesarios para investigar eventos anormales, realizar un seguimiento de las actividades de los usuarios y diagnosticar amenazas de seguridad.
En Windows, Linux y macOS, estos logs comparten el objetivo común de registrar eventos clave del sistema operativo (SO), aunque cada plataforma organiza y estructura los logs a su manera. Comprender cómo funcionan, qué registran y cómo interpretarlos es crucial para los analistas de SOC.

Logs de Windows

Orígenes de registro comunes

- ***Registro del sistema**: Captura los eventos generados por el sistema operativo Windows y sus servicios integrados. Registra los problemas de los controladores, los inicios y apagados del servicio y los mensajes a nivel del kernel.*
- ***Registro de aplicaciones**: almacena eventos específicos de la aplicación, como errores, advertencias o mensajes informativos del software instalado en el sistema (por ejemplo, clientes de bases de datos, herramientas de productividad).*
- ***Registro de seguridad**: se centra en eventos relacionados con la seguridad: intentos de inicio de sesión, bloqueos de cuentas, y cesiones de derechos de usuario. A menudo se utiliza para auditorías e investigaciones forenses.*
- ***Otros logs**: Windows también crea logs dedicados para servicios especializados, como **la replicación DFS** y **PowerShell**, que se pueden ver en los **logs de aplicaciones y servicios** en el Visor de eventos.*

Consejos prácticos de seguimiento

1. ***Visor de eventos**: Integrado en Windows, el Visor de eventos ofrece una forma rápida de ver y filtrar eventos. Los analistas pueden agrupar eventos por gravedad (Crítico, Error, Advertencia, Información) o por ID de evento.*

2. ***Filtrado y búsqueda**: use **el filtrado XML** en el Visor de eventos o en los comandos de PowerShell para buscar identificadores de eventos específicos (por ejemplo, 4624 para inicios de sesión correctos, 4625 para inicios de sesión con errores).*

3. ***Líneas base de seguridad**: supervise los identificadores de eventos de alto valor. Por ejemplo:*

- ***4624*** *(Inicio de sesión exitoso en la cuenta)*
- ***4625*** *(Inicio de sesión fallido)*
- ***4672*** *(Privilegios especiales asignados a un usuario)*
- ***4688*** *(Se ha creado un nuevo proceso)*
- ***4648*** *(se intentó iniciar sesión con contraseñaes explícitas)*

4. ***Registro de PowerShell****: al habilitar el* ***registro de módulos*** *y el* ***registro de bloques de scripts****, los analistas pueden realizar un seguimiento de comandos sospechosos u ofuscados. Consulte Microsoft Docs (registro de PowerShell) para obtener instrucciones.*

Ejemplo: Filtrado de eventos de seguridad a través de PowerShell

```
Get-WinEvent -LogName Security | Where-Object {$_.Id -in 4624, 4625
}
```

Este comando extrae eventos de registro de seguridad para inicios de sesión exitosos y fallidos, lo que permite la detección rápida de actividad anormal.

Logs de Linux

Syslog y Journald

La mayoría de las distribuciones de Linux se basan en ***syslog*** *o* ***systemd-journald*** *para recopilar y administrar mensajes de registro:*

- ***/var/log/syslog*** *o* ***/var/log/messages****: Contiene eventos informativos y no críticos del sistema.*
- ***/var/log/auth.log*** *o* ***/var/log/secure****: se centra en los mensajes relacionados con la autenticación. Esencial para detectar intentos de inicio de sesión por fuerza bruta, actividad sudo o inicios de sesión SSH.*
- ***/var/log/kern.log****: Almacena mensajes a nivel de kernel, útiles para diagnosticar problemas de controladores o eventos inusuales del kernel.*
- ***Journal logs*** *(distribuciones basadas en systemd): consolida los logs en formato binario, accesible vía journalctl.*

Áreas clave a monitorear

1. ***Autenticación****: esté atento a los repetidos intentos fallidos de inicio de sesión, las adiciones de nuevos usuarios en /etc/passwd, o cambios repentinos en el uso de sudo.*
2. ***Cron jobs****: Compruebe* ***/var/log/cron*** *o los logs asociados para ver si hay tareas programadas no autorizadas. Los trabajos cron pueden ser utilizados por los adversarios para la persistencia.*

3. ***Kernel messages****: investigue las advertencias o errores repetidos del kernel que podrían indicar problemas de hardware o una posible actividad de*

rootkit.

4. ***Logs de servicio**: Para servicios como Apache, Nginx o SSH, supervise los logs dedicados (p. ej., /var/log/apache2/access.log, /var/log/nginx/access.log, /var/log/secure) para tráfico o errores de autenticación repetidos.*

Ejemplo: Uso de journalctl

```
# View all logs related to SSH

journalctl -u sshd

# Filter logs for a specific time range

journalctl --since "2023-01-01" --until "2023-01-31"
```

Este enfoque ayuda a los analistas a buscar rápidamente anomalías dentro de un servicio o período de tiempo en particular.

Logs de macOS

Sistema de registro unificado

Desde macOS Sierra (10.12), Apple introdujo un sistema de registro unificado que almacena los mensajes de registro en un formato estructurado:

- ***Aplicación de consola**: La consola incorporada permite ver logs del sistema, informes de diagnóstico y logs de fallas.*
- ***Comandos de registro**: La utilidad de registro en el terminal ofrece un amplio filtrado, transmisión y capacidades de búsqueda. Por ejemplo:*

  #View live log messages (system-

  wide) `log stream --`

  `level=info` #Search for specific

  processes or errors

  ```
  log show --predicate 'process == "sshd" AND eventMessage
  CONTAINS "Failed password"'
  ```

- ***Subsistemas y categorías**: Los logs de macOS clasifican los mensajes por subsistema (p. ej., com.apple.networking) y categoría (p. ej., conexión). Esto ayuda a los analistas a acotar los eventos.*

Logs específicos de seguridad

- ***/var/log/system.log**: Conserva muchos mensajes del sistema central y suele ser*

la primera parada Al solucionar problemas.

- ***Apple System Log (ASL):*** *registro heredado que coexiste con el sistema de registro unificado, accesible a través de herramientas de línea de comandos para versiones anteriores de macOS.*
- ***Logs de autenticación****: Los intentos de iniciar sesión a través de SSH o cuentas locales pueden aparecer en /var/log/asl/ o a través de la interfaz de registro unificada.*

Monitoreo y detección

1. ***Concéntrese en los errores repetidos****: Al igual que en Linux, los repetidos intentos fallidos de SSH o los lanzamientos inesperados de procesos merecen atención.*

2. ***Comprobar los informes*** *de fallos: Los atacantes a veces provocan fallos en las herramientas de seguridad. Los logs de fallos en macOS pueden proporcionar indicadores tempranos de manipulación.*

3. ***Aproveche las herramientas integradas****: utilice la* ***consola*** *para filtrar los logs por proceso o tipo de mensaje. La documentación para desarrolladores de Apple sobre el registro unificado proporciona detalles sobre el uso avanzado.*

Consideraciones multiplataforma

Aspecto	***Windows***	***Linux***	***macOS***
Archivos de registro	*Visor de eventos (sistema, seguridad, etc.)*	*/var/log/syslog, /var/log/auth.log, etc.*	*Sistema de registro unificado (log show, log stream)*

Aspecto	Windows	Linux	macOS
Herramientas comunes	PowerShell, Visor de eventos, WMI	cola, grep, awk, journalctl	Aplicación de consola, CLI de registro
Foco de alerta	Identificadores de evento (4624, 4625, etc.), cambios en las políticas	Fallos de SSH, escaladas de privilegios, errores de servicio de systemd	Errores de SSH, bloqueos del sistema, mensajes inesperados del subsistema
Centralización	Reenvío de eventos de Windows (WEF), logs de Sysmon a SIEM	Rsyslog, Syslog-ng, systemd-journald a SIEM	Exporte logs a través de la función de recopilación de logs o la transmisión a un SIEM

Agentes de registro y centralización

Muchas organizaciones optan por reenviar los logs de Windows, Linux y macOS a un SIEM central o a una plataforma de administración de logs:

- ***Windows**: Windows Event Forwarding (WEF), Sysmon para el registro detallado a nivel de proceso, o agentes de terceros como NXLog o Splunk Universal Forwarder.*
- ***Linux**: Rsyslog, Syslog-ng o systemd-journald pueden reenviar logs a servidores remotos.*
  Beats (Filebeat, Metricbeat) de Elastic también puede recopilar y enviar logs.
- ***macOS:** use agentes de terceros (por ejemplo, Osquery para el registro basado en consultas, o Splunk, agentes de Datadog) para unificar los logs en un solo panel.*

*2.2. Logs de red (Firewa**l**, router, IDS/IPS)*

Los logs de los dispositivos de red y los sistemas de seguridad se encuentran entre las fuentes de datos más críticas de un Centro de Operaciones de Seguridad (SOC). Al analizar los logs de firewall, enrutador e IDS/IPS, los analistas de SOC obtienen visibilidad de los patrones de tráfico, los eventos de seguridad y las posibles anomalías. Esta visibilidad es crucial para identificar el comportamiento malintencionado de forma temprana y para responder a los incidentes antes de que puedan propagarse dentro del entorno. A continuación se muestran los conceptos clave, los procedimientos recomendados y los escenarios del mundo real que ilustran cómo trabajar eficazmente con los logs de red.

1. *Logs de firewall*

Los firewalls suelen ser la primera línea de defensa, ya que filtran el tráfico en función de reglas predefinidas. La supervisión de los logs de firewall proporciona información

sobre las conexiones de red permitidas y denegadas.

1. Campos comunes en los logs del firewall

Los logs típicos de firewall incluirán campos como:

- ***Marca de tiempo**: la fecha y la hora en que se registró el evento.*
- ***IP de origen/IP de destino**: direcciones IP del cliente y del servidor.*
- ***Puerto de origen/puerto de destino**: puertos utilizados por los servicios o aplicaciones que se comunican.*
- ***Protocolo**: Protocolo de red en uso (por ejemplo, TCP, UDP, ICMP).*
- ***Acción**: indica si el tráfico se permitió, se denegó, se quitó o se rechazó.*
- ***Nombre de regla o política**: identifica qué regla de firewall desencadenó la entrada de registro.*

Los firewalls también pueden registrar detalles adicionales como los nombres de las interfaces (por ejemplo, eth0, WAN, LAN), el tamaño del paquete o el motivo de una acción de denegación o rechazo. Los firewalls modernos, especialmente los firewalls de próxima generación (NGFW), pueden registrar datos a nivel de aplicación e información del usuario si se integran con sistemas de gestión de identidades.

Campo	***Descripción***	***Valor de ejemplo***
Timestamp	*Fecha y hora del evento*	*2025-01-25 10:15:32*
IP de origen	*Dirección IP de origen*	*192.168.10.5*
IP de destino	*Dirección IP de destino*	*10.0.5.20*
Puerto de origen	*Puerto TCP/UDP de origen*	*53452*
Puerto de destino	*Puerto TCP/UDP de destino*	*443*
Protocolo	*Protocolo de red (TCP, UDP, ICMP)*	*TCP*
Acción	*Permitidos, denegados, abandonados, etc.*	*Permitido*
Nombre de la regla	*Nombre de directiva o regla de firewall*	*Block_Telnet*

2. Casos de uso práctico

- ***Intentos de conexión bloqueados**: La supervisión de los intentos de conexión repetidos en puertos sensibles (por ejemplo, 22 para SSH o 3389 para RDP) puede revelar intentos de fuerza bruta o escaneos de puertos.*
- ***Volúmenes de tráfico inusuales**: Un aumento repentino en el tráfico de una sola IP o subred podría indicar un intento de DoS o DDoS.*

- ***Supervisión de entrada frente a salida**: Las conexiones de salida a direcciones IP sospechosas o países en los que la organización no realiza negocios pueden ser indicadores tempranos de hosts comprometidos (por ejemplo, malware que llama a casa).*

3. Ejemplo de análisis de logs de firewall en SIEM

A continuación se muestra un ejemplo de consulta de Splunk que filtra las conexiones denegadas con un enfoque en el puerto TCP 3389 (RDP):

```
index=firewall_logs action=DENY dest_port=3389
| stats count by src_ip, dest_ip, action, rule_name
```

Esta consulta ayuda a resaltar las direcciones IP de origen que intentan repetidamente acceder a los servicios RDP pero se deniegan, lo que podría indicar un intento de intrusión.

2. *Logs del router*

Los enrutadores reenvían principalmente paquetes entre redes y mantienen tablas de enrutamiento. Los logs generados por los enrutadores a menudo se centran en los mensajes del sistema, las actualizaciones de enrutamiento y los errores de interfaz en lugar de los datos específicos de la aplicación. Sin embargo, siguen siendo fundamentales para la visibilidad general, especialmente en entornos con arquitecturas distribuidas.

1. Tipos de logs de enrutador

- ***Logs del sistema o de eventos**: incluye mensajes sobre reinicios de dispositivos, bloqueos de software o Cambios en la configuración.*
- ***Logs de protocolo de** enrutamiento: información relacionada con BGP, OSPF, EIGRP u otros protocolos de enrutamiento.*
- ***Logs de interfaz**: cambios de estado en las interfaces (arriba/abajo), errores de paquetes (errores CRC, colisiones) y uso de ancho de banda.*
- ***Logs de autenticación**: inicios de sesión exitosos o fallidos a través de SSH, Telnet o acceso de consola al enrutador.*

Los logs del router a menudo siguen el formato estándar de Syslog (por ejemplo, routers Cisco con niveles de gravedad 0-7). La integración de estos logs en un SIEM permite a los analistas correlacionar los cambios en la topología de la red con los eventos de seguridad (por ejemplo, si una interfaz de enrutador se cae justo antes de un incidente de seguridad en ese segmento).

2. Ejemplo: Mensajes de Syslog del router Cisco

Los routers Cisco envían mensajes Syslog con varios niveles de gravedad. Un mensaje de ejemplo podría ser el siguiente:

```
<189>Jan 25 10:25:10 MY-ROUTER: %LINK-3-UPDOWN: Interface
GigabitEthernet0/1, changed state to up
```

- *189 corresponde a la prioridad de Syslog.*
- *%LINK-3-UPDOWN indica un cambio de estado de enlace con nivel de gravedad 3 (error).*
- *El mensaje indica qué interfase cambió de estado.*

En un SIEM, puede filtrar los eventos %LINK-3-UPDOWN para realizar un seguimiento de los cambios de estado inesperados de la interfaz. Si una interfaz se cae repentinamente, puede indicar un problema físico, una configuración incorrecta o una actividad maliciosa con el objetivo de interrumpir segmentos de la red.

3. *Logs IDS/IPS*

Los sistemas de detección de intrusiones (IDS) y los sistemas de prevención de intrusiones (IPS) supervisan el tráfico de red en busca de signos de comportamiento malintencionado, infracciones de políticas o firmas de ataque conocidas. Mientras que los firewalls suelen operar en la capa de transporte o de red, las soluciones IDS/IPS pueden inspeccionar los paquetes con más profundidad (Capa 7), lo que proporciona un contexto más rico sobre las amenazas a nivel de aplicación.

1. IDS vs. IPS

- ***IDS (Sistema de Detección de Intrusos): Detecta** amenazas potenciales y genera alertas. Eso No bloquea automáticamente el tráfico.*
- ***IPS (Sistema de Prevención de Intrusiones): Detecta** amenazas y puede tomar medidas preventivas, como descartar paquetes maliciosos o bloquear direcciones IP en tiempo real.*

2. Campos comunes en logs IDS/IPS

- ***ID de firma**: un identificador único para la regla o firma activada (por ejemplo, las reglas de Snort tienen valores SID).*

- ***Evento o mensaje de alerta**: El nombre o la descripción de la actividad sospechosa (por ejemplo, "ET TROYANO Zeus Tracker").*
- ***Gravedad o prioridad**: Indica la criticidad de la alerta.*
- ***IP de origen/IP de destino/Puertos**: Información sobre el flujo de tráfico.*
- ***Acción**: si el tráfico se ha detectado, eliminado o permitido.*

3.3. Ejemplo práctico con Suricata

Suricata es un popular motor IDS/IPS de código abierto. Suricata genera logs JSON

que pueden ser ingeridos por herramientas SIEM como Elasticsearch o Splunk. Una entrada de alerta típica de Suricata en formato JSON podría incluir:

```
{
  "timestamp": "2025-01-25T10:30:45.123456+0000",
  "flow_id": 1234567890,
  "event_type": "alert",
  "src_ip":
  "192.168.1.100",
  "src_port": 53452,
  "dest_ip": "10.0.5.20",
  "dest_port": 80,
  "proto": "TCP",
  "alert": {
    "action": "blocked",
    "gid": 1,
    "signature_id": 2010935,
    "rev": 3,
    "signature": "ET TROJAN Known Malicious Domain",
    "category": "Trojan Activity",
    "severity": 2

  }
}
```

el ejemplo anterior:

- ***signature_id:** 2010935 corresponde a un ID de regla de Suricata que hace referencia a una firma de troyano específica.*
- ***acción:** El tráfico fue bloqueado por Suricata (modo IPS).*
- ***categoría:** "Actividad troyana" indica el tipo de amenaza.*

3.4. Escenarios de ataque de la vida real

- ***Intentos de inyección SQL**: Las soluciones IDS/IPS buscan patrones en las solicitudes HTTP que coinciden con las técnicas de inyección de SQL conocidas.*
- ***Kits de exploits**: si un host intenta descargar o conectarse a un dominio de kit de exploits, los logs de IDS/IPS pueden revelar el nombre de dominio sospechoso y la coincidencia de firma.*
- ***Movimiento lateral**: Los atacantes pueden intentar moverse horizontalmente dentro de una red. IDS/IPS puede detectar patrones de tráfico SMB o RDP inusuales.*

4. *Análisis y análisis de logs de red en la práctica*

1. Gestión de logs e integración de SIEM

Los analistas de SOC suelen centralizar los logs de firewall, router e IDS/IPS en un

SIEM para la correlación y el análisis. Esto permite referencias cruzadas de eventos de múltiples fuentes. Por ejemplo, si una alerta IDS indica una firma troyana y los logs del firewall muestran tráfico saliente a una IP sospechosa, el SIEM puede generar una alerta de mayor prioridad.

Ejemplo de fragmento de configuración de Logstash para analizar logs JSON de Suricata:

```
input {
  file {
    path => "/var/log/suricata/eve.json"
    type => "suricata"
    codec => "json"

  }
}

filter {
  if [event_type] == "alert" {
    mutate {
      add_tag => ["suricata_alert"]

    }
  }
}

output {
  elasticsearch {
    hosts => ["localhost:9200"]
    index => "suricata-alerts-%{+YYYY.MM.dd}"

  }
}
```

Esta configuración lee el archivo de eve.json de Suricata, filtra los eventos de alerta y, a continuación, los etiqueta como suricata_alert antes de enviarlos a Elasticsearch.

4.2. Reglas de correlación

En un SIEM, las reglas de correlación pueden buscar condiciones como:

1. ***Alto volumen de conexiones denegadas**: si se producen más de 100 denegaciones de firewall desde la misma IP de origen en 5 minutos, genere una alerta.*

2. ***Alertas de varios IDS para el mismo host**: si un host activa más de 3 IDS diferentes*
   firmas en un corto período de tiempo, elevan la prioridad del incidente.

3. ***Alerta de +IDS + Caída de la interfaz del router:** Si una interfaz crítica se cae y se detectan varias alertas IDS en segmentos de red adyacentes, investigue un*

posible sabotaje o un compromiso generalizado.

Al crear reglas de correlación que combinen diferentes tipos de logs, los analistas de SOC pueden detectar ataques coordinados y reducir el volumen de falsos positivos.

5. *Desafíos y mejores prácticas*

- ***Volumen de registro**: Los dispositivos de red pueden generar una gran cantidad de datos. Puede ser necesario usar filtros o muestreos, pero tenga cuidado de no descartar información importante.*
- ***Normalización**: Diferentes proveedores (Cisco, Palo Alto, Fortinet, etc.) a menudo tienen formatos de registro únicos. La normalización de los campos (por ejemplo, garantizar la nomenclatura coherente de src_ip, dest_ip) es crucial para una correlación eficaz.*
- ***Cifrado y transporte seguro**: Asegúrese de que los datos de registro se transmitan de forma segura, por ejemplo, utilizando TLS para Syslog (Syslog over TLS). Los logs no cifrados pueden ser interceptados y manipulados por los adversarios.*
- ***Ajuste regular**: los conjuntos de reglas de IDS/IPS necesitan actualizaciones periódicas para reflejar nuevas amenazas. Del mismo modo, las políticas de firewall deben revisarse para asegurarse de que se alinean con el entorno de red en evolución.*
- ***Sincronización de tiempo**: NTP (Network Time Protocol) debe estar habilitado y configurado correctamente en todos los dispositivos para mantener marcas de tiempo consistentes. Las marcas de tiempo precisas son fundamentales para la correlación de eventos.*

6. *Referencias y lecturas adicionales*

- ***Documentación oficial de Suricata**: https://suricata-ids.org/docs/*
- ***Snort (IDS/IPS)Documentación**: https:// www.snort.org/*
- ***Guía de Cisco Syslog**: https:// www.cisco.com/c/en/us/support/docs/security-vpn/syslog/*
- ***Mejores prácticas de firewall (NIST):** https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-41r1.pdf*

Estas fuentes proporcionan información fidedigna sobre la configuración, los estándares de registro y los patrones de detección de amenazas para dispositivos de red y soluciones de seguridad.

2.3. *Logs de aplicaciones y bases de datos*

Los logs de aplicaciones capturan eventos y comportamientos vinculados directamente a la funcionalidad de una aplicación. Pueden incluir interacciones de usuario, operaciones del sistema, excepciones, detalles de depuración y eventos

personalizados definidos por los desarrolladores. Los logs de base de datos, por otro lado, registran todas las acciones relacionadas con las transacciones de datos, los cambios de esquema, la autenticación y los posibles errores o cuellos de botella de rendimiento en un sistema de base de datos. Juntos, estos logs proporcionan una visión holística de cómo funciona el software y cómo se accede a los datos o cómo se manipulan. Esto es crucial para detectar actividades no autorizadas, problemas de rendimiento y otras anomalías. A continuación se muestran las consideraciones clave, las prácticas recomendadas y los ejemplos del mundo real para ayudarle a supervisar y analizar eficazmente los logs de aplicaciones y bases de datos.

Descripción de los logs de aplicaciones

Tipos comunes de logs de aplicaciones

1. ***Logs de errores y excepciones***
   Estos capturan comportamientos inesperados y seguimientos de pila. Por lo general, se generan mediante marcos como Log4j o Logback de Java, la biblioteca de registro de Python o . Funciones de registro integradas de NET.

2. ***Logs de depuración y diagnóstico***
   Estos logs detallan las operaciones internas, que a menudo contienen información detallada que se usa durante el desarrollo o la solución de problemas. Los logs de depuración pueden ser extremadamente detallados, por lo que Por lo general, solo se habilitan en entornos de prueba o desarrollo, a menos que un El problema requiere una comprensión más profunda.

3. ***Logs de transacciones o eventos***
   Las aplicaciones que controlan las transacciones de los usuarios, como los pagos de comercio electrónico, a menudo generan logs de transacciones. Estos logs detallan cada paso en el flujo del usuario (por ejemplo, agregar artículos a un carrito, pagar, procesar pagos).

4. ***Logs de auditoría***
   Algunas aplicaciones producen logs de auditoría por motivos de cumplimiento o seguridad. Estos logs realizan un seguimiento del acceso de los usuarios, los cambios de roles o las actualizaciones críticas de la configuración, lo que le ayuda a ver quién hizo qué y cuándo.

Marcos y formatos de registro

Las aplicaciones modernas a menudo se basan en marcos de registro estandarizados:

- ***Java (Log4j, Logback)***
- ***.NET (Serilog, NLog)***
- ***Python (registro)***
- ***Node.js (Winston, Pino)***

Estos marcos permiten a los desarrolladores configurar niveles de registro, estructurar mensajes de registro (por ejemplo, en JSON) y especificar destinos de salida como consola, archivos o servicios de agregación externos. La coherencia en el formato es importante para el análisis y la correlación dentro de un SIEM.

Ejemplo de configuración (Log4j2 en Java)

```
<Configuration status="warn">
  <Appenders>
    <File name="FileLogger" fileName="logs/application.log">
      <PatternLayout pattern="%d{ISO8601} [%t] %-5level %logger{36}
- %msg%n" />
    </File>
  </Appenders>
  <Loggers>
    <Logger name="com.example.app" level="info" additivity="false">
      <AppenderRef ref="FileLogger"/>
    </Logger>
    <Root level="error">
      <AppenderRef ref="FileLogger"/>
    </Root>
  </Loggers>
</Configuration>
```

Este fragmento de código especifica un archivo application.log, utilizando un patrón para registrar las marcas de tiempo, los nombres de los subprocesos, los niveles de registro y el mensaje real. Al establecer el nivel de raíz en error y el registrador de aplicaciones en info, puede evitar ruido innecesario.

Supervisión de los logs de aplicaciones en la práctica

1. ***Establecer el comportamiento de referencia***
   Conocer el comportamiento normal de la aplicación (por ejemplo, tiempos de respuesta promedio, tasas de error típicas) ayuda a detectar anomalías, como una afluencia repentina de excepciones específicas que podrían indicar un ataque o una configuración incorrecta.

2. ***Busque patrones de ataque comunes***
   Los intentos de acceso no autorizados a menudo se muestran como errores de inicio de sesión repetidos, parámetros sospechosos en URL (por ejemplo, sondas de inyección SQL) o comportamientos inusuales en la administración de sesiones. Las aplicaciones web pueden registrar errores 404 o métodos HTTP sospechosos con mayor frecuencia en caso de ataque.

3. ***Integración con SIEM***
   *Para correlacionar los logs de aplicaciones con eventos del sistema o de la red, utilice una herramienta SIEM como **Splunk**, **IBMQRadar** o **Elastic Security**. Por ejemplo, al correlacionar el evento de "múltiples inicios de sesión fallidos" de una aplicación con un registro de firewall que muestra un*

análisis de IP sospechoso, puede confirmar o descartar rápidamente un intento de intrusión.

4. ***Alertas y umbrales***
   *Alertas basadas en umbrales sobre tasas de error, caídas del volumen de transacciones o picos de excepciones puede detectar incidentes a tiempo. La detección de anomalías basada en el aprendizaje automático en herramientas como **Azure Sentinel** o **AWS Security Hub** puede refinar aún más las alertas mediante la identificación de patrones no capturados por las reglas estáticas.*

5. ***Políticas de retención***
   Debido al volumen, los logs de aplicaciones pueden crecer rápidamente. Debe definir políticas de retención que equilibren los requisitos de seguridad y los costos de almacenamiento. Los marcos de cumplimiento (por ejemplo, PCI, DSS, HIPAA) a veces dictan períodos mínimos de retención para tipos de logs específicos.

Descripción de los logs de la base de datos

Tipos clave de logs de base de datos

1. ***Logs de transacciones***
   Capture todos los cambios en los datos de la base de datos. Son fundamentales para la recuperación y el análisis forense. Por ejemplo, en Microsoft SQL Server, el registro de transacciones realiza un seguimiento de cada modificación en el orden en que se producen, lo que permite la recuperación a un momento dado.

2. ***Logs de errores***
   Estos resaltan eventos críticos, como problemas de inicio del servidor o errores graves que afectan la disponibilidad de la base de datos. Algunos ejemplos son los error.log de MySQL o los logs de alertas de Oracle.

3. ***Logs generales de query (MySQL) /Logs de auditoría (varios proveedores)***
   Registre todas las consultas recibidas por el servidor o realice un seguimiento de la actividad de la cuenta. Son muy valiosos para detectar ataques de inyección SQL, extracción de datos sospechosos o intentos de escalada de privilegios.

4. ***Logs lentos de query***
   Encontradas en MySQL o PostgreSQL, estas capturan consultas que superan un determinado umbral de tiempo de ejecución. Las consultas lentas pueden indicar problemas de rendimiento o posibles intentos de denegación de servicio si un atacante está manipulando las consultas.

Estrategias prácticas de monitoreo

1. ***Revisión periódica de queries sospechosas***
   Supervise las consultas que eliminan o modifican tablas críticas sin los

tickets de control de cambios esperados. Busque también búsquedas con comodines o extractos de datos de gran tamaño que se produzcan a horas inusuales.

2. ***Detección de abuso de privilegios***
   Si un usuario con privilegios mínimos comienza a ejecutar consultas típicas de un administrador, es un fuerte indicador de contraseñaes comprometidas o escalada de privilegios. Aplicar el privilegio mínimo y, a continuación, revisar los logs en busca de anomalías es una estrategia potente.

3. ***Análisis de patrones de error***
   Los mensajes de error repetidos de la base de datos, como "nombre de columna no válido" o "error de sintaxis", pueden indicar intentos de inyección de SQL. Los analistas de SOC pueden configurar reglas de correlación de SIEM para marcar errores repetitivos desde una única IP de origen.

4. ***Datos de rendimiento***
   Los logs que apuntan a un uso elevado de recursos o tiempos de espera pueden ser una advertencia temprana de un ataque de fuerza bruta o de denegación de servicio en la capa de la base de datos.

Ejemplos de configuraciones y queries

MySQL

Para habilitar el registro de consultas general:

```
SET GLOBAL general_log = 'ON';
SET GLOBAL general_log_file = '/var/log/mysql/general.log';
To enable the slow query log:
SET GLOBAL slow_query_log = 'ON';
SET GLOBAL long_query_time = 2; -- Queries taking longer than 2
seconds will be logged
```

***Nota**: El registro de todas las consultas puede afectar significativamente el rendimiento, por lo que solo hay que habilitarlo temporalmente para los diagnósticos o canalizar los logs a un sistema centralizado donde pueda analizarlos y analizarlos de forma eficaz.*

PostgreSQL

PostgreSQL tiene amplias configuraciones de registro en postgresql.conf. Por ejemplo:

```
logging_collector = on
log_directory = 'pg_log'
log_filename = 'postgresql-
%a.log' log_statement = 'all'
```

```
log_min_duration_statement = 2000  # logs queries over 2ms
```

Al establecer log_statement en all, puede ver todas las instrucciones, aunque esto suele ser demasiado detallado para la producción.

Escenarios del mundo real

- ***Detección de exfiltración de datos***
  Un analista de SOC observa un registro de aplicación que muestra valores de parámetros inusuales en una llamada a la API de REST. Al hacer referencias cruzadas a los logs de la base de datos, el analista confirma varias declaraciones SELECT grandes que recuperan datos confidenciales de los clientes. La correlación adicional con los logs de red muestra una gran transferencia de datos a una IP externa. Los logs apuntan colectivamente a un intento de exfiltración de datos en curso.
- ***Auditoría para el cumplimiento***
  En una aplicación de servicios financieros, los requisitos de cumplimiento exigen auditar cada transacción. Al revisar los logs de la aplicación (que capturan la capa lógica) y el
  Los auditores pueden confirmar que cada depósito o retiro está autorizado y ejecutado correctamente.
- ***Identificación de ataques de rendimiento***
  Una serie de consultas lentas puede parecer inicialmente un cuello de botella en el rendimiento. Sin embargo, una investigación más profunda revela que los atacantes están elaborando intencionadamente consultas que consumen muchos recursos para degradar la capacidad de respuesta de la aplicación. Las alertas en el SIEM correlacionan estas consultas lentas con errores 503 repetidos en el servidor web, lo que confirma un intento de denegación de servicio.

Recursos adicionales

- ***Serie de hojas de trucos de OWASP****: https://cheatsheetseries.owasp.org/*
  Ofrece directrices sobre prácticas de registro seguras, específicamente en torno a la saneación de logs y la prevención de la falsificación de logs.
- ***Documentos oficiales de MySQL****: https://dev.mysql.com/doc/*
  Instrucciones detalladas sobre cómo configurar logs de errores, logs de consultas generales y logs de consultas lentas.
- ***Documentación de PostgreSQL****: https:// www.postgresql.org/docs/*
  Contiene guías de configuración completas para las funciones de registro y auditoría.
- ***Microsoft SQL Server Docs****: https://docs.microsoft.com/en-us/sql/*

Explica cómo administrar e interpretar logs de transacciones, logs de errores y otros datos de diagnóstico.

- ***Documentos de Oracle Database**: https://docs.oracle.com/en/database/*
  Proporciona detalles sobre el registro de alertas, los archivos de seguimiento y las configuraciones de auditoría avanzadas.

Comparaciones y datos

Tipo de registro	***Ejemplos***	***Caso de uso típico***	***Posible información sobre seguridad***
Error/Excepción	*Seguimientos de pila, referencias de líneas de código*	*Depuración de bloqueos de aplicaciones, identificación de módulos defectuosos*	*Las excepciones frecuentes pueden insinuar entradas maliciosas o intentos de aprovechar vulnerabilidades*
Transacción	*Logs de comercio electrónico, transacciones bancarias*	*Auditar el éxito/fracaso de las acciones críticas*	*El monitoreo en tiempo real ayuda a detectar transacciones fraudulentas*
Auditoría (DB de App C)	*Acciones de usuario, cambios de rol, modificaciones de esquema*	*Cumplimiento de la normativa, rendición de cuentas*	*Identificación de acciones de administración no autorizadas o escalamientos de privilegios*
Query lento	*Queries que superan un umbral de tiempo*	*Ajuste del rendimiento o análisis de cuellos de botella*	*Identificación de posibles intentos de DoS o ataques de agotamiento de recursos*

*Al recopilar y analizar estos logs, considere la posibilidad de normalizar los campos (marcas de tiempo, ID de usuario, nombres de host) para que los diferentes orígenes de registro se puedan correlacionar de forma eficaz. Algunas plataformas SIEM o soluciones de registro centralizado (por ejemplo, ELK **Stack**) le permiten definir asignaciones de campos comunes y paneles que unifican la información de las aplicaciones y las bases de datos.*

2.4. *Logs de seguridad (AV, EDR, XDR)*

Los logs de seguridad generados por las soluciones Antivirus (AV), Endpoint Detection and Response (EDR) y Extended Detection and Response (XDR) son cruciales para las operaciones modernas de SOC. Ofrecen visibilidad granular de las posibles amenazas que afectan a los endpoints y al entorno en general. A continuación se muestra una exploración de los fundamentos, junto con ejemplos del mundo real y orientación para una supervisión eficaz de los logs.

Comprensión de los componentes

Logs de antivirus (AV)

Las soluciones antivirus se centran principalmente en la detección de firmas de malware conocidas y en el bloqueo de archivos sospechosos. Sus logs suelen incluir:

- ***Detecciones** de malware: Alertas que se activan cuando un archivo coincide con una firma conocida o muestra un comportamiento malicioso.*
- ***Quarantine y acciones de corrección**: logs que muestran qué archivos se pusieron en cuarentena, se eliminaron o se neutralizaron de otro modo.*
- ***Eventos de actualización y análisis**: logs de actualizaciones de firmas, análisis programados y*
  Resultados del análisis a demanda.

Ejemplo del mundo real

*Una herramienta antivirus tradicional como **Microsoft Defender Antivirus** (parte de Seguridad de Windows) genera logs en el Registro de eventos de Windows:*

- ***El identificador de evento 1116** indica la detección de malware.*
- ***El identificador de evento 5001** registra el inicio del motor de detección.*
  Al agregar estos ID de eventos en un SIEM, los analistas pueden ver rápidamente los patrones de intentos de infección y confirmar que se han aplicado las actualizaciones.

Logs de detección y respuesta de puntos de conexión (EDR)

Las soluciones EDR amplían las funciones básicas de antivirus al proporcionar telemetría de endpoints en profundidad, detección de amenazas en tiempo real y capacidades de respuesta. Los datos de registro comunes incluyen:

- ***Creación y terminación** de procesos: seguimiento detallado de los parámetros de la línea de comandos, el contexto del usuario y las rutas de los archivos.*
- ***Indicadores de comportamiento**: Observaciones relacionadas con actividades sospechosas como el código*
  inyección, escaladas de privilegios o modificaciones inusuales del registro.
- ***Acciones de aislamiento y respuesta**: logs que muestran cuándo y por qué se aisló un endpoint, se bloquearon las conexiones de red o se ejecutó un script automatizado para la contención.*

*Los logs EDR a menudo presentan una secuencia de eventos correlacionados, lo que facilita a los analistas de SOC la reconstrucción de la línea de tiempo de un ataque. Herramientas como **CrowdStrike Falcon**, **SentinelOne** o **Carbon Black** ofrecen paneles que muestran reglas de detección activadas (por ejemplo, técnicas MITRE ATTCCK) junto con acciones de corrección automatizadas.*

Ejemplo de análisis de logs EDR (Splunk)

A continuación se muestra un ejemplo de cómo podría analizar los logs de EDR en Splunk para identificar procesos secundarios sospechosos de PowerShell:

```
index=edr_logs parent_process=PowerShell.exe
| stats count by child_process, user, host
| where count > 3
```

Esta consulta busca cualquier proceso secundario generado por PowerShell y marca cualquier proceso repetido ocurrencias, que podrían indicar scripts maliciosos o técnicas de vivir de la tierra.

Logs de detección y respuesta extendidas (XDR)

Las soluciones XDR adoptan el enfoque centrado en los endpoints de EDR y lo amplían para incorporar datos de dispositivos de red, cargas de trabajo en la nube y aplicaciones. El objetivo es unificar la detección, la investigación y la respuesta en varias capas del entorno de TI.

- ***Correlación entre fuentes**: XDR agrega logs de endpoints, gateways de correo electrónico, proveedores de identidad y más, aplicando análisis para descubrir amenazas ocultas.*
- ***Integraciones híbridas y en la nube**: la telemetría de las plataformas en la nube y las cargas de trabajo en contenedores a menudo se fusiona con los datos de los endpoints, lo que ofrece una visión completa de los ataques complejos.*
- ***Respuesta adaptativa**: en función del aprendizaje automático y las reglas de correlación, XDR puede activar playbooks automatizados que respondan a las amenazas en tiempo real (por ejemplo, deshabilitar cuentas de usuario comprometidas, aislar hosts infectados o bloquear dominios sospechosos en el firewall).*

Arquitecturas de referencia

- ***Microsoft 365 Defender** integra datos de puntos de conexión (Defender para punto de conexión), correo electrónico (Defender para Office 365), identidades (Azure Active Directory) y aplicaciones en la nube (Defender for Cloud Apps).*
- ***Palo Alto Cortex XDR** procesa los datos de los endpoints y se integra con sensores de red o firewalls para proporcionar una correlación mejorada.*

Procedimientos recomendados para la recopilación de logs

1. ***Centralice los logs en un SIEM:** consolide todos los logs de AV, EDR y XDR en una plataforma SIEM como **Splunk**, **Elastic Stack** o **IBM QRadar**. Esto garantiza una vista única para la búsqueda de amenazas y la clasificación de*

alertas.

2. ***Utilice un formato de registro coherente**: siempre que sea posible, estandarice el formato (por ejemplo, JSON, Syslog) para optimizar el análisis, la correlación y el almacenamiento a largo plazo.*
3. ***Conserve el historial suficiente**: en función de los requisitos normativos y el modelado de amenazas, conserve los logs históricos el tiempo suficiente para investigar ataques de movimiento lento o amenazas persistentes avanzadas.*
4. ***Correlación entre múltiples fuentes**: las alertas de antivirus por sí solas pueden proporcionar un contexto mínimo. Cuando se cruzan con la telemetría de puntos de conexión y los patrones de inicio de sesión de usuario, revelan el panorama general, especialmente relevante para ataques avanzados o de varias etapas.*
5. ***Implemente reglas de detección automatizadas**: aproveche las capacidades de detección integradas de su solución EDR/XDR y compleméntelas con reglas personalizadas adaptadas a su entorno. Por ejemplo, cree una alerta cuando un proceso seguro conocido genere un proceso secundario inusual (por ejemplo, outlook.exe iniciar cmd.exe).*
6. ***Aproveche la inteligencia de amenazas**: enriquezca los eventos de detección con fuentes de inteligencia de amenazas (por ejemplo, VirusTotal, AlienVault OTX). Esto ayuda a validar la actividad sospechosa, especialmente cuando una alerta hace referencia a un dominio malintencionado conocido o a un hash de archivo.*

Eventos de seguridad comunes a los que hay que prestar atención

Tipo de evento	***Indicadores clave***	***Herramientas de ejemplo***
Detecciones de malware	*Hashes de archivos, firmas conocidas, comportamientos sospechosos de archivos*	*Microsoft Defender, McAfee, Symantec*
Conductual Anomalías	*Modificaciones inusuales en el registro, anormales Árboles de proceso*	*Halcón de CrowdStrike, CentinelaUno*
Escalada de privilegios	*Intenta cambiar el privilegio de usuario o ejecutar procesos como administrador*	*Correlación Sysmon + EDR*
Intentos de exfiltración	*Conexiones de red a dominios sospechosos o grandes transferencias de datos*	*Palo Alto Cortex XDR, Logs de Splunk*
Mecanismos de persistencia	*Nuevos servicios, elementos de inicio, tareas programadas*	*Reglas de detección de Sysmon +EDR*

El monitoreo de estos eventos casi en tiempo real permite a los analistas de SOC priorizar las alertas de mayor riesgo e iniciar acciones de contención rápidamente.

Consejos prácticos de seguimiento

- ***Seguimiento de intentos de corrección fallidos y exitosos**: si un antivirus intenta poner en cuarentena un archivo repetidamente pero falla, podría ser un signo de malware avanzado o manipulación del usuario.*
- ***Supervise el estado del agente EDR**: asegúrese periódicamente de que los agentes EDR se ejecuten en todos los terminales. El tiempo de inactividad inesperado del agente puede ser un indicador temprano del intento de un atacante de deshabilitar los controles de seguridad.*
- ***Revise los resultados del manual de estrategias automatizadas**: las plataformas XDR a menudo ejecutan respuestas automatizadas. Confirme que estas respuestas son efectivas y están alineadas con los procedimientos de respuesta a incidentes de su organización.*
- ***Interactúe con la documentación del proveedor**: cada proveedor de AV, EDR o XDR tiene mejores prácticas específicas para la recopilación e interpretación de logs. Por ejemplo, **Microsoft Defender para punto** de conexión publica directrices de registro detalladas en Microsoft Docs.*

Ejemplo de flujo de incidentes con AV, EDR y XDR

1. ***Alerta AV**: se activa en un ejecutable sospechoso con un hash malicioso conocido.*
2. ***Correlación EDR**: Asigna el ejecutable sospechoso a un árbol de procesos, mostrando que fue lanzado por un script inusual.*
3. ***Visibilidad XDR**: confirma que el script se descargó de un dominio no reconocido y vincula este dominio a un actor de amenazas conocido a través de fuentes de inteligencia de amenazas.*
4. ***Respuesta automatizada**: XDR o una plataforma SOAR pone en cuarentena el punto final, bloquea el*
   en el firewall y abre un ticket en el sistema de gestión de incidentes.
5. ***Acción** de analista de SOC: investiga toda la cadena de eventos, verifica la eliminación de amenazas y actualiza las reglas de detección para evitar ataques similares.*

Al combinar las fortalezas de los logs AV, EDR y XDR en una estrategia de monitoreo bien estructurada, los analistas de SOC pueden responder rápidamente a una amplia gama de amenazas, desde malware básico hasta ataques sofisticados y persistentes.

2.5. *Logs en la nube (AWS, Azure, GCP) y logs de contenedores (Docker, Kubernetes)*

Las plataformas en la nube y los sistemas de orquestación de contenedores se han convertido en una parte esencial de muchas organizaciones. En un entorno SOC, la supervisión de los logs de estas plataformas es fundamental para la detección de

amenazas, el cumplimiento y la resolución de problemas. A continuación, se muestra una descripción general de las fuentes de registro más importantes y consideraciones prácticas para AWS, Azure, GCP, Docker y Kubernetes.

Logs de AWS

Tipos de registro comunes

1. ***Logs de CloudTrail***
    - ***Propósito**: Realizar un seguimiento de las llamadas a la API y la actividad de la cuenta en los servicios de AWS.*
    - ***Campos clave**: eventName, eventSource, awsRegion, sourceIPAddress, userAgent,*
      requestParameters, responseElements.
    - ***Uso en seguridad**: identifica acciones sospechosas o no autorizadas, como cambios inesperados en las políticas de IAM, creación o eliminación de recursos críticos o inicios de sesión inusuales en la consola.*
2. ***Logs de CloudWatch***
    - ***Propósito**: Registro centralizado para los servicios de AWS (logs del sistema EC2, logs de funciones de Lambda, etc.).*
    - ***Campos clave**: varían en función de los eventos específicos del servicio; normalmente incluyen marcas de tiempo, nivel de registro (ERROR, WARNING, INFO) y mensajes de aplicaciones personalizadas.*
    - ***Uso en seguridad**: ayuda a correlacionar eventos de nivel de sistema con actividades de nivel superior. Ejemplo: correlacionar los logs de errores del sistema de una instancia EC2 con un intento de acceso no autorizado que se muestra en CloudTrail.*
3. ***Logs de flujo de VPC***
    - ***Propósito**: Capturar información de flujo de red (IP de origen/destino, puertos, aceptación/rechazo de tráfico).*
    - ***Campos clave**: versión, account-id, interface-id, srcaddr, dstaddr, srcport, dstport, protocolo, acción, estado del registro.*
    - ***Uso en seguridad**: Identifica patrones de tráfico inusuales o intentos de exfiltración de datos,*
      como grandes transferencias de datos salientes o tráfico desde rangos de IP desconocidos.

Ejemplo práctico

Un flujo de trabajo típico para la ingesta implica el reenvío de logs de flujo de

CloudTrail y VPC a un bucket de S3 y, a continuación, el uso de Amazon Kinesis o una herramienta de terceros (por ejemplo, Logstash) para analizar y enviar eventos a un SIEM. Por ejemplo, con AWS CLI puede habilitar el registro de CloudTrail:

```
aws cloudtrail create-trail \
  --name MySecurityTrail \
  --s3-bucket-name my-security-logs \
  --include-global-service-events
```

Para obtener más información, consulte la documentación de AWS CloudTrail.

Logs de Azure

Tipos de registro comunes

1. ***Logs de actividad de Azure***
    - ***Propósito**: Proporcionar información sobre las operaciones de administración (por ejemplo, creación, modificación o eliminación de recursos).*
    - ***Campos clave**: authorization, caller, category, operationName, resourceId, status.*
    - ***Uso en seguridad**: detecte la creación de recursos no autorizados, cambios en los grupos de seguridad o intentos de elevar los privilegios.*

2. ***Logs de Azure Monitor (LogAnalytics)***
    - ***Propósito**: recopilar logs de recursos, contenedores, máquinas virtuales y aplicaciones de Azure.*
    - ***Campos clave**: varían según el tipo de recurso; por lo general, incluyen marcas de tiempo, ID de operación, detalles del usuario y otros datos contextuales.*
    - ***Uso en seguridad**: Ofrece amplias capacidades de consulta y correlación. Los equipos de SOC pueden detectar anomalías combinando señales de múltiples fuentes (logs de actividad, logs de VM, etc.).*

3. ***Logs de diagnóstico***

- ***Propósito**: información detallada de servicios específicos de Azure, como Key Vault logs de acceso, logs de Azure App Service o logs de Azure Storage.*
    - ***Campos clave**: dependen del servicio, pero a menudo incluyen puntos de conexión de solicitud, detalles de autenticación y códigos de resultado.*
    - ***Uso en seguridad**: detecta un posible uso indebido de contraseñaes, actividad sospechosa en el almacenamiento de datos o comportamiento inusual de la aplicación.*

Ejemplo práctico

El envío de logs a Azure Monitor se puede realizar mediante la configuración de una configuración de diagnóstico para cada recurso. Por ejemplo, para enrutar logs de actividad a Azure Monitor y a una cuenta de almacenamiento:

```
Set-AzDiagnosticSetting -ResourceId
/subscriptions/<SUBSCRIPTION_ID>/resourceGroups/<RESOURCE_GROUP>/pr
o
viders/Microsoft.Web/sites/<APP_NAME> `

  -WorkspaceId <AZURE_MONITOR_WORKSPACE_ID> `

  -StorageAccountId
/subscriptions/<SUBSCRIPTION_ID>/resourceGroups/<RESOURCE_GROUP>/pr
o viders/Microsoft.Storage/storageAccounts/<STORAGE_ACCOUNT_NAME>
`

  -Enabled $true
```

Consulte la documentación de Azure Monitor para obtener más información.

Logs de GCP

Tipos de registro comunes

1. ***Logs de auditoría de Cloud***
    - ***Propósito**: registrar eventos de administración y acceso a datos para los servicios de GCP (similar a AWS CloudTrail).*
    - ***Campos clave**: protoPayload.serviceName, protoPayload.methodName, resourceName, authenticationInfo, requestMetadata.*
    - ***Úselo en seguridad**: Muestre intentos de escalada de privilegios o modificaciones sospechosas en los recursos de GCP (por ejemplo, habilitación o inhabilitación de servicios críticos).*
2. ***Logs de flujo de VPC***
    - ***Propósito**: recopilar información sobre el flujo de red para las VPC de Google Cloud.*
    - ***Campos clave**: srcIP, destIP, srcPort, destPort, protocol, connectionEstablished, bytesSent, bytesReceived.*
    - ***Uso en seguridad**: Detecte la actividad de reconocimiento o exfiltración mediante el análisis*
      Patrones de tráfico entrante y saliente.
3. ***Registro en la nube***

- ***Propósito**: Servicio de registro central para eventos de servicios de GCP, contenedores, aplicaciones personalizadas.*
- ***Campos clave**: datos específicos del servicio, marcas de tiempo, niveles de gravedad, etiquetas de recursos (p. ej., k8s_container, gce_instance).*
- ***Uso en seguridad**: permite la correlación de logs de nivel de aplicación con eventos de nivel de infraestructura.*

Ejemplo práctico

Para exportar logs de GCP a un SIEM, puedes crear un receptor que dirija los logs a un tema de Pub/Sub, A continuación, el recopilador personalizado o de terceros puede reenviar. Un ejemplo con la CLI de gcloud:

```
gcloud logging sinks create my-security-sink \

storage.googleapis.com/<BUCKET_NAME> \

--log-filter="resource.type=gce_instance AND severity>=WARNING"
```

Para obtener orientación detallada, consulta la documentación de Google Cloud Logging.

Logs de contenedor (Docker, Kubernetes)

Los contenedores empaquetan las aplicaciones y sus dependencias en una sola unidad ligera. Dado que los contenedores a menudo ejecutan cargas de trabajo efímeras, el registro continuo y estandarizado es clave para la supervisión de la seguridad.

Logs de Docker

1. ***Logs del motor de Docker***
   - ***Ubicación**: Normalmente se almacena en /var/log/docker.log en hosts Linux.*
   - ***Campos clave**: eventos de nivel de demonio, como inicios/paradas de contenedores, extracciones de imágenes, errores del tiempo de ejecución del contenedor.*
   - ***Uso en seguridad**: identifique la creación de contenedores no autorizados o las imágenes malintencionadas que se extraen de logs que no son de confianza.*
2. ***Logs de contenedor STDOUT/STDERR***

- ***Ubicación**: De forma predeterminada, almacenado en /var/lib/docker/containers/<container_id>/<container_id>-json.log.*
- ***Uso en seguridad**: Detecte anomalías dentro de las aplicaciones en ejecución (por ejemplo, mensajes de error repetidos que indican un intento de fuerza bruta o un uso indebido de la aplicación).*

1. ***Controladores de registro de Docker***

   - ***Tipos**: json-file, syslog, fluentd, gelf, awslogs y otros.*
   - ***Uso en seguridad**: Puede integrarse con soluciones de registro centralizadas, reduciendo el*
     Posibilidad de manipulación de logs si el contenedor se ve comprometido.

Ejemplo de Docker

Con el controlador de registro syslog, puede dirigir los logs de contenedor a un servidor syslog remoto:

```
docker run --log-driver=syslog --log-opt syslog-
address=tcp://192.168.1.10:514 \
          --log-opt tag="{{.ImageName}}/{{.Name}}/{{.ID}}" \
          my_secure_image
```

Consulte la documentación de registro de Docker para obtener detalles de configuración.

Logs de Kubernetes

1. ***Logs de contenedor***

   - ***Colección**: Por lo general, se recopila a través de logs de kubectl <pod_name> o a través de un registro*
     agente (Fluentd, Logstash o un patrón de sidecar).
   - ***Uso en seguridad**: detecte errores sospechosos de aplicaciones o desencadenantes específicos, como respuestas 401/403 repetidas que indican un intento de fuerza bruta de autenticación.*

2. ***Logs de Kubelet***

   - ***Ubicación**: las rutas difieren según la distribución del sistema operativo; puede incluir*
     /var/registro/kubelet.log.
   - ***Uso en seguridad**: realice un seguimiento de los problemas de programación de contenedores, los intentos no autorizados de*

programar pods con privilegios o las interacciones que podrían indicar un nodo comprometido.

3. ***Logs del plano de control (APIServer, Scheduler, Controller Manager)***
    - ***Ubicación**: A menudo en /var/log/ en el nodo del plano de control o agregado mediante una solución de registro centralizada.*
    - ***Uso en seguridad**: identifique las llamadas a la API no autorizadas, las creaciones sospechosas de pods o los intentos de escalar privilegios a través de enlaces de roles de Kubernetes.*
4. ***Logs de auditoría***
    - ***Propósito**: Registre cada solicitud al servidor de la API de Kubernetes.*
    - ***Configuración**: Habilite la auditoría modificando --audit-log-path y --audit-policy-file en el servidor de API.*
    - ***Uso en Seguridad**: Fundamental para la investigación de incidentes. Puede realizar un seguimiento de todo, desde los cambios de RBAC hasta los generadores de contenedores con privilegios.*

Ejemplo de Kubernetes

Un archivo de política de auditoría simple (audit-policy.yaml) podría tener el siguiente aspecto:

```
apiVersion: audit.k8s.io/v1
kind: Policy
rules:
- level:
  Metadata
  resources:
  - group: ""
    resources: ["secrets"]
- level: RequestResponse
  resources:
  - group: ""
    resources: ["pods/exec"]
```

Puede consultar la documentación de auditoría de Kubernetes para obtener configuraciones más avanzadas.

Consideraciones prácticas y consejos del mundo real

- ***Centralización**: Ya sea que utilice AWS CloudWatch, Azure Monitor, Google Cloud Logging o pilas ELK autoalojadas, la centralización de logs de múltiples proveedores de nube y plataformas de contenedores es esencial para la correlación.*
- ***Controles de acceso**: asegúrese de que los logs, especialmente los que contienen información confidencial (contraseñaes, datos personales), se almacenen en áreas restringidas. Configure roles de IAM o equivalentes para controlar quién puede leer o exportar logs.*
- ***Alertas y paneles**: cree alertas específicas. Por ejemplo, cree una alerta si un nuevo Se crea un enlace de roles de clúster de Kubernetes que concede privilegios de administrador del clúster.*
- ***Políticas de retención**: alinearse con los requisitos normativos. Algunas industrias requieren mantenimiento logs durante períodos prolongados, mientras que otros pueden priorizar la optimización de costos.*
- ***Volumen de registro frente a relevancia**: filtrar el "ruido" excesivo ayuda a evitar la sobrecarga de datos. Configure el registro granular solo cuando sea necesario o implemente el muestreo de logs para eventos de gran volumen, como los logs de depuración de contenedores.*
- ***Correlación entre plataformas**: al investigar un incidente, haga referencias cruzadas de los logs de contenedores con los logs de infraestructura en la nube subyacentes. Por ejemplo, si se identifica un contenedor malicioso, la revisión de los logs de auditoría de AWS CloudTrail o GCP Cloud puede mostrar quién lo implementó y desde dónde.*

Al cubrir estas áreas, los analistas de SOC obtienen una mejor visibilidad de los entornos basados en la nube y en contenedores. Cada plataforma ofrece diferentes tipos de logs y varias formas de configurarlos, pero el principio general sigue siendo el mismo: necesita un registro completo, centralizado y confiable para detectar y responder de manera efectiva a los incidentes de seguridad.

2.6. *Logs de IoT/SCADA/OT*

Los dispositivos IoT (Internet de las cosas), SCADA (Control de supervisión y adquisición de datos) y OT (tecnología operativa) desempeñan un papel fundamental en las industrias modernas, desde las plantas de fabricación hasta las redes de energía. Los logs generados por estos sistemas proporcionan información valiosa sobre el estado operativo, las métricas de rendimiento y las posibles amenazas de seguridad. La supervisión eficaz de estos logs puede ser un reto debido a la diversidad de protocolos, la variedad de sistemas operativos y firmware implicados, y los requisitos de alta disponibilidad que suelen caracterizar a los entornos industriales. A continuación se muestra una descripción general de las consideraciones principales, ejemplos de orígenes de registro y procedimientos recomendados para garantizar una supervisión completa.

Descripción de los entornos de IoT, SCADA y OT

Descripción general de IoT

Los dispositivos IoT suelen ser sistemas integrados que se utilizan en diversos contextos: hogares inteligentes, sensores industriales, dispositivos sanitarios, etc. A menudo tienen:

- ***Recursos limitados** (CPU, memoria) que dificultan el almacenamiento local de logs.*
- ***Firmware personalizado** que puede o no producir logs estandarizados.*
- ***Limitaciones de red** , como ancho de banda bajo o conectividad intermitente.*

SCADA y OTSystems

SCADA y otros sistemas OT controlan y monitorean procesos industriales en energía, fabricación, transporte e infraestructura crítica. Las distinciones clave incluyen:

- ***Procesamiento en tiempo real o casi en tiempo real** con estrictos requisitos de rendimiento y disponibilidad.*
- ***Uso de protocolos especializados** (por ejemplo, Modbus, DNP3, OPC-UA) donde las capacidades de registro pueden diferir de las que se encuentran en los entornos de TI.*
- ***Componentes heredados** que podrían no ser compatibles con los estándares de ciberseguridad actuales o modernos*
  marcos de registro.

Tipos de logs que se van a supervisar

1. ***Logs del sistema**: Los sistemas operativos integrados o las variaciones especializadas del sistema operativo utilizadas por los controladores industriales (PLC, RTU, HMI) pueden producir mensajes del kernel o entradas de syslog estándar cuando estén disponibles.*
2. ***Logs de tráfico de red**: Muchos protocolos industriales pueden ser capturados por sensores de red o pasarelas especializadas. Las anomalías en el tráfico, como los códigos de función Modbus inesperados, pueden indicar actividad maliciosa o una configuración incorrecta.*
3. ***Logs de aplicaciones**: El software SCADA registra eventos como acciones del operador, umbrales de proceso, estados de alarma y problemas de conectividad del dispositivo. Estos logs pueden revelar cambios no autorizados en puntos de ajuste críticos.*
4. ***Logs de firmware/dispositivos**: Los dispositivos IoT y OT suelen generar mensajes relacionados con las comprobaciones de integridad del firmware o las actualizaciones de parches. La supervisión de estos puede ayudar a detectar intentos sospechosos de instalar firmware no autorizado.*
5. ***Logs de dispositivos de seguridad**: Cuando la seguridad perimetral está presente en redes industriales, los firewalls, IDS/IPS y los dispositivos de seguridad OT dedicados producen logs sobre intentos de intrusión, tráfico bloqueado y amenazas detectadas.*

Desafíos prácticos en la recopilación de logs

1. ***Complejidad del protocolo***
   Muchos protocolos industriales y de IoT son propietarios o solo están parcialmente documentados. La interpretación de los logs requiere una comprensión de la versión específica del protocolo y la implementación del proveedor.

2. ***Potencia de almacenamiento y procesamiento limitada***
   Algunos dispositivos rotan los logs rápidamente debido al espacio limitado en el disco. Es posible que los analistas de SOC deban reenviar estos logs a un servidor central en tiempo real para evitar la pérdida de datos.

3. ***Requisitos de alta disponibilidad***
   Detener o reconfigurar un sistema de producción para habilitar ciertos logs podría ser inviable si interrumpe las operaciones críticas. Los analistas deben planificar cuidadosamente las configuraciones de registro, a menudo durante los tiempos de inactividad programados.

4. ***Segmentación y espacios de aire***
   Las redes industriales a veces están aisladas ("air-gapped") de las redes corporativas. Se necesitan mecanismos seguros (por ejemplo, diodos de datos, hosts de salto) para transmitir los logs sin introducir nuevas vulnerabilidades.

Prácticas recomendadas para la supervisión

1. Estandarizar y normalizar logs

Siempre que sea posible, configure los dispositivos para que generen logs en un formato estandarizado, como syslog o JSON. Este paso simplifica la ingesta en una solución de administración de logs o SIEM.

Ejemplo: Configuración de syslog en un controlador industrial basado en Linux

```
sudo apt-get install rsyslog
sudo systemctl enable rsyslog
# Configure /etc/rsyslog.conf to forward logs
*.* @192.168.100.10:514
```

En un entorno de OT, es posible que necesite documentación específica del proveedor para habilitar el reenvío de syslog. Si syslog no es una opción, utilice una puerta de enlace industrial o un convertidor de protocolos que pueda analizar los logs nativos y enviarlos en un formato común.

2. Correlacionar con datos de procesos físicos

Los sistemas SCADA a menudo rastrean las variables del proceso (por ejemplo, temperatura, presión, flujo). Las referencias cruzadas de estas métricas con intentos de inicio de sesión o cambios de configuración pueden revelar acciones maliciosas o

erróneas. Las reglas de correlación de SIEM pueden buscar:

- ***Cambios repentinos en el punto de ajuste seguidos** de reconocimientos de alarma.*
- ***Cuentas de usuario no autorizadas** creadas justo antes de que el equipo crítico se desconecte.*
- ***Escaneos repetidos de la red** que coinciden con lecturas de temperatura elevadas en sensores IoT.*

3. Implementación de privilegios mínimos y controles de acceso

Las soluciones industriales modernas a menudo incluyen controles de acceso basados en roles. Los logs de las herramientas de administración de identidades y accesos muestran quién accedió al sistema y qué cambios se realizaron:

- *Asegúrese de que se registren todos los inicios de sesión y las escalaciones de roles.*
- *Habilite la autenticación multifactor (MFA) para conexiones remotas a consolas SCADA y OT.*

4. Supervise las actualizaciones de firmware y la integridad

Las actualizaciones de firmware no programadas o no autorizadas pueden ser una señal temprana de compromiso. Supervise los logs para:

- ***Falta de coincidencia de la versión del firmware** o reinicios inesperados.*
- ***Eventos de recreación de imágenes del dispositivo** que se producen fuera de las ventanas de mantenimiento normales.*

Muchos proveedores de dispositivos industriales ofrecen funciones de comprobación de integridad. Aproveche estos y reenvíe los eventos relacionados al SOC para su revisión.

5. Aproveche la inteligencia especializada en amenazas

Las fuentes de inteligencia de amenazas centradas en las vulnerabilidades de ICS/SCADA pueden ayudar a enriquecer su análisis de logs. Por ejemplo, MITRE ATTCCK para ICS (https://collaborate.mitre.org/attackics/index.php/Main_Page) enumera las técnicas y tácticas utilizadas por los adversarios que apuntan a la tecnología operativa. La incorporación de estos indicadores en las reglas de supervisión puede mejorar las capacidades de detección.

Ejemplos del mundo real

Ejemplo 1: Intento de manipulación de la red eléctrica

Un atacante obtiene acceso a una estación de trabajo SCADA utilizada para administrar una red eléctrica regional. La revisión de los logs muestra:

1. ***Varios inicios de sesión RDP fallidos*** *desde una IP externa.*
2. ***Inicio de sesión exitoso*** *con una cuenta de administrador (posiblemente a través de contraseñaes robadas).*
3. ***Cambios repentinos*** *en los comandos de apertura/cierre del disyuntor emitidos en momentos inusuales.*

Las referencias cruzadas de los logs del software SCADA con los logs del firewall revelan que las conexiones entrantes se saltaron los canales VPN normales, lo que indica un compromiso en el perímetro. La correlación oportuna de estos logs evitó una interrupción a gran escala.

Ejemplo 2: Red de sensores de IoT comprometida

Una instalación de fabricación experimenta lecturas de temperatura irregulares de un grupo de sensores de IoT. Los logs recopilados de la plataforma de administración de dispositivos muestran:

1. ***Aumento inusual*** *en el tráfico de red dirigido a los sensores.*
2. ***Intentos de manipulación de firmware*** *registrados por las comprobaciones de integridad integradas del dispositivo.*
3. ***Conexiones*** *salientes de los sensores a direcciones IP no autorizadas.*

La investigación descubrió que los sensores tenían un firmware obsoleto con una vulnerabilidad conocida. La aplicación rápida de parches y el bloqueo de las direcciones IP maliciosas en el firewall mitigaron una mayor exfiltración de datos y posibles daños al sistema.

Resumen comparativo

Aspecto	***Montón***	***SCADA /OT***
Enfoque principal	*Sensores C de dispositivos inteligentes*	*Control de procesos industriales*
Formatos de registro	*A menudo propietarios o mínimos*	*Syslog, propietario (p. ej., logs de PLC)*
Protocolos	*MQTT, CoAP, HTTP(S)*	*Modbus, DNP3, OPC-UA*
Seguridad	*Varía ampliamente; A menudo sin parches*	*Seguridad, disponibilidad, operaciones en tiempo real*
Desafíos	*Limitaciones de recursos*	*Sistemas heredados, redes aisladas*

Pasos prácticos para los analistas de SOC

1. ***Identifique las fuentes de registro clave****: priorice los controladores críticos (PLC, RTU) y los dispositivos IoT de alto impacto.*
2. ***Establezca un reenvío seguro de*** *logs: utilice canales cifrados (por ejemplo, túneles TLS, SSH) al enviar logs a través de los límites de la red.*
3. ***Crear perfiles*** *de referencia: comprenda el funcionamiento normal de los*

dispositivos y detecte desviaciones. Por ejemplo, si un PLC normalmente recibe comandos solo durante el horario comercial, se puede activar una alerta en los cambios fuera del horario laboral.

4. ***Combine la supervisión basada en la red y en el host**: Muchos ataques contra los sistemas de OT implican movimiento lateral o pivote desde el lado de la TI. Incluya logs de NetFlow, firewall y endpoints en su análisis.*

5. ***Revise la guía del proveedor**: Los principales proveedores industriales como Siemens, Rockwell Automation y Schneider Electric publican documentación sobre las mejores prácticas de registro y seguridad. Manténgase al día con los parches y avisos de los proveedores.*

3. *Práctica clave de monitoreo*

La supervisión eficaz de los logs depende de procesos sólidos, herramientas configuradas correctamente y objetivos claros. Los analistas de SOC deben centrarse en estrategias que ayuden a distinguir los comportamientos normales de los sospechosos, preservar y proteger los datos relevantes y aprovechar la automatización siempre que sea posible. En las siguientes prácticas se describen las consideraciones básicas para detectar anomalías, conservar logs de forma segura y emplear tecnologías de soporte.

1. *Detección de anomalías e incidencias (alertas, correlación)*

Detección de anomalías frente a detección basada en firmas

La detección de anomalías implica establecer una línea de base de las operaciones normales y marcar las desviaciones. Esto es útil para identificar amenazas de día cero o comportamientos inusuales de los usuarios. Por el contrario, la detección basada en firmas se basa en indicadores conocidos de compromiso (IoC), como direcciones IP específicas, valores hash o patrones de ataque. La mayoría de los SOC modernos utilizan un enfoque híbrido para capturar tanto las amenazas desconocidas (anomalías) como las actividades maliciosas conocidas (firmas).

Análisis contextual de logs

Al investigar eventos, rara vez es suficiente examinar un único origen de registro. La correlación de datos a través de múltiples fuentes (por ejemplo, firewall, detección y respuesta de puntos finales [EDR] y logs de Active Directory) puede revelar ataques sofisticados. Por ejemplo, ver errores repetidos de autenticación de usuario en un registro de Active Directory y conexiones salientes inusuales simultáneas en un registro de firewall podría apuntar a un intento de fuerza bruta seguido de exfiltración de datos.

Umbrales de alerta y ajuste fino

Los equipos SOC a menudo implementan reglas de alerta en los sistemas SIEM o IDS/IPS para notificarles sobre actividades sospechosas. Equilibrar estos umbrales es fundamental:

- ***Demasiado estricto**: corra el riesgo de inundar el SOC con falsos positivos, lo que provocaría fatiga de alertas y amenazas reales pasadas por alto.*
- ***Demasiado relajado**: permite que los incidentes de seguridad importantes pasen desapercibidos, lo que retrasa la respuesta y la corrección.*

Encontrar el equilibrio adecuado a menudo requiere un ajuste iterativo basado en datos históricos, detalles del entorno y procesos empresariales conocidos.

Ejemplo del mundo real

Considere una situación en la que una cuenta de usuario accede repentinamente a cientos de archivos en un servidor de archivos a horas inusuales. Las reglas de detección de anomalías pueden marcar este comportamiento si se desvía del patrón de uso normal de ese usuario. Mientras tanto, una regla basada en firmas podría detectar que algunos de estos archivos coinciden con kits de herramientas malintencionadas conocidas (por ejemplo, Mimikatz o similares). La correlación de ambas alertas permite a los analistas de SOC identificar un posible compromiso de la cuenta y un incidente de robo de datos mucho más rápido.

Ejemplo SIEM Query (Splunk)

```
index=windows_logs sourcetype=WinEventLog:Security
EventCode=4625 OR EventCode=4624
| stats count by Account_Name, EventCode
| where count > 20
```

En este ejemplo, la consulta comprueba si los inicios de sesión se han realizado correctamente (4624) o si hay errores (4625) y busca cualquier registro de cuenta varias veces más allá de un umbral, lo que podría indicar intentos de fuerza bruta o movimiento lateral.

3.2. Retención y seguridad de los logs

Políticas de retención

Los logs deben conservarse durante un período específico, a menudo definido por las políticas de la organización, las regulaciones (por ejemplo, PCI DSS, HIPAA) y los estándares de cumplimiento. Los períodos de retención típicos oscilan entre 90 días y varios años, en función de la confidencialidad de los datos y los requisitos del sector. Los analistas de SOC deben verificar que las políticas de retención se alineen tanto con las necesidades de búsqueda de amenazas como con las obligaciones legales.

Consideraciones sobre el almacenamiento de logs

- ***Almacenamiento centralizado**: El almacenamiento de logs en un único repositorio (por ejemplo, un SIEM o una plataforma de gestión de logs) simplifica la búsqueda, la correlación y la copia de seguridad.*
- ***Redundancia: el** uso de varias ubicaciones de almacenamiento o agrupación en clústeres garantiza que los logs permanezcan disponibles incluso si falla el hardware.*
- ***Cifrado**: el cifrado de logs en reposo (por ejemplo, mediante el cifrado a nivel de disco) y en tránsito (por ejemplo, TLS para el reenvío de logs) evita el acceso no autorizado.*
- ***Controles de acceso**: implemente controles de acceso basados en roles (RBAC) para que solo el personal autorizado pueda ver o manipular logs confidenciales.*

Manejo de la integridad del registro

Para preservar el valor probatorio, las organizaciones deben asegurarse de que los logs no se puedan manipular fácilmente:

1. ***Hashing**: La generación de hashes (por ejemplo, mediante SHA-256) para los archivos de registro y su almacenamiento por separado ayuda a detectar modificaciones no autorizadas.*
2. ***Almacenamiento Write-Once-Read-Many (WORM):** Algunas plataformas admiten una funcionalidad similar a WORM en la que los logs se pueden escribir pero no modificar posteriormente.*
3. ***Logs de auditoría**: realice un seguimiento de quién accedió al repositorio de logs, cuándo accedió y qué cambios (si los hubo) se realizaron.*

Ejemplo de almacenamiento seguro de logs

Una empresa puede utilizar un bucket de Amazon S3 con el control de versiones y el cifrado del lado del servidor habilitados para archivar logs de sistemas locales. AWS Key Management Service (KMS) proporciona almacenamiento seguro de claves, mientras que AWS Identity and Access Management (IAM) aplica permisos estrictos. La documentación oficial sobre esta configuración se puede encontrar en las páginas de documentación de AWS.

3.3. Herramientas de apoyo (SIEM, SOAR)

SIEM (Gestión de Eventos e Información de Seguridad)

Las soluciones SIEM recopilan, analizan y normalizan logs de varias fuentes, lo que permite a los analistas buscar, correlacionar y generar alertas casi en tiempo real. Las plataformas SIEM comunes incluyen Splunk Enterprise Security, IBM QRadar y Microsoft Sentinel. Características principales:

- ***Agregación y normalización de logs**: estandariza los eventos a un formato común.*
- ***Reglas de correlación**: Crea alertas cuando se producen varios*

indicadores en una secuencia definida.

- ***Paneles de control e informes**: Ofrece interfaces visuales para monitorear la postura de seguridad y presentar métricas a la gerencia.*

SOAR (Orquestación, Automatización y Respuesta de Seguridad)

Las plataformas SOAR automatizan tareas que, de otro modo, los analistas realizarían manualmente. Algunos ejemplos son Palo Alto Networks, Cortex, XSOAR (antes Demisto) y Splunk Phantom. Estas herramientas se pueden configurar para:

1. ***Enriquezca las alertas**: recopile automáticamente información del host o de la red de las fuentes de inteligencia sobre amenazas.*
2. ***Contener incidentes**: Por ejemplo, deshabilite una cuenta de usuario comprometida o aísle un*

   Punto de conexión malintencionado.

3. ***Organice las respuestas**: desencadene flujos de trabajo que impliquen varios sistemas de seguridad y TI.*

Automatización y manuales de estrategias

Un enfoque estándar en SOAR es desarrollar playbooks, flujos de trabajo automatizados que definen cómo responder a incidentes específicos. Por ejemplo, si una alerta indica que se ha ejecutado un script de PowerShell sospechoso en un servidor, un cuaderno de estrategias podría:

1. *Recupere los logs de puntos de conexión relevantes.*
2. *Compare el hash del script con una base de datos de inteligencia sobre amenazas.*
3. *Quarantine el host si el hash es malicioso.*
4. *Cree un ticket en el sistema de gestión de incidencias.*

Comparación de alto nivel de SIEM vs. SOAR

Característica	***SIEM***	***ELEVARSE***
Enfoque principal	*Centralización de logs, correlación, Alertas*	*Automatización y orquestación de respuestas*
Procesamiento de datos	*Agregación y análisis de grandes volúmenes de datos de registro*	*Integración con múltiples herramientas de seguridad/TI para enriquecer y actuar sobre las alertas*

Característica	***SIEM***	***ELEVARSE***
Salida típica	*Alertas de seguridad, paneles, informes*	*Automatización del flujo de trabajo, cuadernos de estrategias y acciones de contención*

Complejidad de uso	*Medio a Alto*	*De medio a alto (depende de la automatización deseada)*

En muchos casos, los SOC integran SIEM y SOAR para una cobertura completa. El SIEM maneja la ingesta y correlación a gran escala, mientras que la plataforma SOAR automatiza los pasos de investigación y respuesta. Esta integración reduce el tiempo medio de detección (MTTD) y el tiempo medio de respuesta (MTTR), lo que en última instancia fortalece la postura de seguridad de la organización.