Módulo 4. Observabilidad de red

La observabilidad de la red es la práctica que permite obtener visibilidad integral y en tiempo real sobre el rendimiento, el comportamiento y el estado interno de una red informática, a partir del análisis de su resultado.

En los equipos de operaciones o administración IT, la diferencia entre reaccionar y resolver suele depender de la calidad de la evidencia. No alcanza con saber que un enlace está activo o que un servidor responde: es necesario explicar por qué se degradó un servicio, qué parte de la red lo provocó, a quién afectó y qué señales lo demostraron. Esa capacidad de respuesta basada en evidencia se conoce como visibilidad, y constituye la base del trabajo con observabilidad.

La visibilidad cobra especial relevancia porque los servicios actuales rara vez se alojan en un único entorno. Una misma transacción puede atravesar una red local, una WAN, un túnel VPN, un firewall, un balanceador, un proveedor cloud y un conjunto de microservicios.

Cuando el usuario expresa que «anda lento» o que «se corta», el desafío consiste en convertir esa percepción en datos: identificar el tramo afectado, caracterizar el tipo de degradación y proponer una acción que

reduzca el impacto. Ese trabajo se asemeja más a una investigación breve y estructurada que a la simple verificación de indicadores visuales.

En este módulo se abordará cómo construir visibilidad a partir de dos enfoques complementarios. Por un lado, se analizará la definición y lectura de métricas de red —como disponibilidad, throughput, latencia, errores, consumo de recursos y otros indicadores de salud—, así como su representación en consolas de monitoreo continuo. Por otro, se incorporará el análisis de flujos y telemetría (NetFlow, IPFIX, sFlow) como un mapa de conversaciones y volúmenes que permite descubrir patrones, picos y comportamientos anómalos sin necesidad de capturar cada paquete.

El objetivo no es memorizar herramientas, sino desarrollar criterio operativo: saber qué medir, cómo medirlo, cómo interpretar una gráfica, cuándo confiar en un indicador y cuándo solicitar más contexto; cómo pasar de una alerta a una hipótesis; y cómo utilizar los flujos para priorizar investigaciones, detectar saturación, reconocer la degradación de un servicio y aportar señales tempranas frente a posibles amenazas.

Según los enfoques actuales de observabilidad, la práctica se basa en recolectar señales diversas —métricas, eventos, logs, trazas y datos de red — y transformarlas en decisiones accionables para distintas audiencias.

A lo largo de la lectura, se verá que la observabilidad no es un concepto abstracto, sino que se manifiesta en decisiones concretas: decidir si un umbral debe medirse por promedio o percentil; determinar qué punto de la red debe exportar flujos; establecer cuánto tiempo conservar los datos; identificar qué alertas son críticas y cuáles deben agruparse; o definir cómo comunicar un hallazgo sin ambigüedades. Al final, un desafío integrador permitirá aplicar este enfoque en un caso realista.

Unidad 2. Registro de dispositivos

Referencias

Unidad 1. Flujos y SNMP

1.1. Herramientas prácticas de implementación

En redes, el monitoreo tradicional suele responder a una pregunta directa: «¿está funcionando?». Se basa en indicadores de estado —como la disponibilidad de un equipo, el uso de CPU, la ocupación de una interfaz, la presencia de errores o caídas— y permite detectar condiciones que requieren atención.

La observabilidad, en cambio, se orienta a una pregunta diferente: «¿por qué está pasando lo que pasa?». Supone una capacidad explicativa más amplia, al permitir conectar señales heterogéneas para comprender causas y relaciones.

Una forma práctica de distinguir ambos enfoques es considerar el tipo de incidentes que resuelven mejor. Cuando un problema es conocido y repetible —por ejemplo, un enlace que se satura todos los lunes a la misma hora—, los umbrales y alertas del monitoreo suelen ser suficientes. En

cambio, si el problema es intermitente, involucra múltiples dominios (red, aplicación, autenticación, firewall, nube) o se manifiesta como una degradación difícil de reproducir, se requiere observabilidad: datos más granulares, con mayor contexto y capacidad de correlación.

La observabilidad de red permite detectar, diagnosticar y responder a problemas de infraestructura de forma proactiva.

En entornos modernos, donde las redes son cada vez más distribuidas y complejas, disponer de herramientas eficaces y accesibles resulta particularmente valioso para pequeñas y medianas empresas (PYME). A continuación, se presenta un conjunto de soluciones prácticas, gratuitas y ampliamente utilizadas para el monitoreo de flujos, el análisis de tráfico y las métricas de rendimiento.

Herramientas clave de implementación

A continuación, se describen algunas soluciones ampliamente utilizadas para la implementación práctica de observabilidad de red, con foco en el análisis de flujos, monitoreo de tráfico y métricas de rendimiento.

- **Ntopng** (https://www.ntop.org/products/traffic-analysis/ntop/)

Ntopng es una herramienta de monitoreo de tráfico de red basada en web, desarrollada como evolución del clásico ntop. Permite analizar en tiempo real el comportamiento del tráfico y proporciona información detallada sobre hosts, protocolos y posibles cuellos de botella. Su interfaz es intuitiva y accesible desde cualquier navegador. Es compatible con flujos como NetFlow, sFlow e IPFIX, y ofrece opciones de exportación en formatos como JSON o InfluxDB. En sistemas Ubuntu o Debian, puede instalarse fácilmente mediante los comandos apt. Se recomienda su uso para visualizar tráfico en tiempo real desde interfaces específicas.

- **LibreNMS** (https://www.librenms.org/)

LibreNMS es una solución de monitoreo de redes de código abierto que se apoya en SNMP y otras tecnologías para recolectar datos de dispositivos. Permite el descubrimiento automático de equipos, la generación de alertas, el envío de notificaciones y la integración con mapas geográficos. Además, ofrece soporte para mensajes Syslog y SNMP traps,

lo que amplía la capacidad de observación de eventos de red. Para su implementación, requiere un servidor con MySQL, PHP y un servidor web como Nginx o Apache.

- **Zabbix** (https://www.zabbix.com/)

Zabbix es una plataforma completa de monitoreo para redes, servidores y servicios. Soporta una variedad de métodos de recolección de datos, entre ellos SNMP, IPMI y agentes personalizados. Su interfaz web permite definir umbrales, generar alertas y crear paneles visuales. Es escalable y adecuada para infraestructuras distribuidas. Entre sus ventajas se destaca la disponibilidad de plantillas específicas para dispositivos como switches, routers y otros equipos SNMP. El sitio oficial es

- **Prometheus + Grafana**

Prometheus, junto con Grafana, conforma una solución robusta de monitoreo basada en métricas y series temporales. Prometheus recolecta datos en tiempo real, mientras que Grafana se encarga de su visualización mediante paneles altamente personalizables. Esta

combinación es compatible con múltiples exportadores, como node_exporter y SNMP exporter, y resulta ideal para supervisar recursos como CPU, memoria, tráfico de red y capacidad general del sistema. Su enfoque flexible y modular se adapta bien a arquitecturas dinámicas.

- **Wireshark**

Wireshark no es una herramienta de observabilidad centralizada, pero sigue siendo fundamental en el análisis puntual de tráfico. Permite capturar y examinar paquetes en detalle, lo que la convierte en una herramienta clave para el diagnóstico preciso de fallos, comportamientos anómalos o eventos puntuales en la red. Su uso es especialmente útil cuando se requiere una inspección detallada que complemente los datos agregados de otras herramientas.

Buenas prácticas

Para implementar estas herramientas de manera eficaz, se recomienda adoptar una serie de buenas prácticas que garanticen tanto la seguridad como la estabilidad del entorno de monitoreo. Es aconsejable utilizar máquinas virtuales para llevar a cabo pruebas en entornos aislados, lo

que permite evaluar el funcionamiento de cada solución sin comprometer la red de producción. Siempre que sea posible, se debe asegurar el tráfico SNMP mediante el uso de SNMPv3, que incorpora mecanismos de autenticación y cifrado más robustos que las versiones anteriores. También es fundamental documentar detalladamente la configuración de cada herramienta instalada, lo cual facilita futuras tareas de mantenimiento, actualización o resolución de problemas. Por último, se sugiere comenzar con una red piloto —es decir, un segmento pequeño y controlado— antes de escalar la solución al resto de la infraestructura.

Actividad práctica

Esta actividad es opcional y está pensada como un ejercicio complementario para aplicar los conceptos desarrollados en la unidad.

El objetivo es implementar una solución básica de monitoreo de red en un entorno local utilizando LibreNMS o ntopng.

Para ello, se sugiere instalar VirtualBox junto con una distribución Linux, preferentemente Ubuntu Server 22.04. En el caso de optar por LibreNMS, se deberá seguir la guía oficial de instalación disponible en: https://docs.librenms.org/Installation/Install-LibreNMS/.

Una vez instalado el sistema, se recomienda configurar SNMP en un switch o router virtualizado (por ejemplo, mediante GNS3) o en un dispositivo físico, si se dispone de uno. El siguiente paso será agregar el dispositivo a LibreNMS y verificar su disponibilidad en la consola. Finalmente, se deberá explorar la visualización de interfaces, alertas generadas y demás información relevante para el monitoreo del estado del equipo.

1.2. SNMP. Telemetría y umbrales

SNMP (*simple network management protocol*)

SNMP (*simple network management protocol*) es un estándar de la capa de aplicación que permite gestionar y supervisar dispositivos de red como routers, switches, firewalls, servidores y otros componentes. Fue diseñado para recopilar información, monitorear el rendimiento, detectar fallos y, en ciertos casos, configurar parámetros de forma remota. Su funcionamiento se basa en el intercambio de información entre dispositivos mediante un modelo cliente-servidor, donde los agentes SNMP recolectan datos que son consultados por un sistema de gestión.

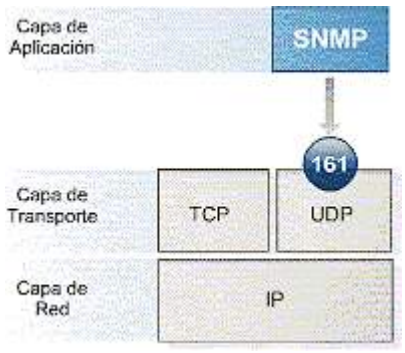
Actualmente, existen tres versiones del protocolo: SNMPv1, SNMPv2 y SNMPv3. Todas comparten características básicas en cuanto al funcionamiento general, pero difieren en sus capacidades. SNMPv2 introdujo operaciones adicionales para mejorar la interacción con los dispositivos, mientras que SNMPv3 incorporó mecanismos de seguridad más sólidos, incluyendo autenticación y cifrado, lo que lo convierte en la opción recomendada en entornos productivos.

Aspectos generales de SNMP

El protocolo SNMP permite a un administrador de red supervisar y controlar el funcionamiento de uno o más dispositivos a través de una red. Entre sus funciones principales se encuentran el monitoreo del rendimiento general, la observación del comportamiento de los dispositivos y la detección de fallos para facilitar su resolución.

SNMP opera sobre el protocolo UDP y utiliza, por defecto, el puerto 161. La figura siguiente ilustra su ubicación dentro de la pila de protocolos TCP/IP.

Figura 1. Ubicación del protocolo SNMP en la pila de protocolos TCP/IP



Fuente: [imagen sin título sobre ubicación del protocolo SNMP en la pila de protocolos TCP/IP], (s.f.).

Componentes clave de SNMP

SNMP define los siguientes elementos:

- **Estaciones de administración de red (NMS)**

Las estaciones de administración de red, conocidas como NMS (por *network management stations*), constituyen el componente encargado de la mayor parte del procesamiento y de los requerimientos del sistema de administración. Puede haber más de una NMS dentro de una infraestructura, y todas actúan como puntos centrales de supervisión.

Cada estación de administración se conecta a los dispositivos gestionados y recolecta la información disponible mediante un gestor (manager), que es el sistema central encargado de recopilar datos. Este gestor suele estar implementado como un servidor de monitoreo, con soluciones como Zabbix, LibreNMS o Nagios. Su

función principal es concentrar la información proveniente de los dispositivos de red y procesarla para que sea accesible y comprensible para los operadores.

En el gestor se definen y administran los siguientes elementos clave. La *management information base* (MIB) es una base de datos estructurada de manera jerárquica que contiene los objetos gestionables a través de SNMP. Cada uno de esos objetos está identificado mediante un *object identifier* (OID), que actúa como un identificador único dentro de la MIB.

- **Dispositivos administrados**

Un dispositivo administrado es un *host* que contiene un agente SNMP. Estos dispositivos recolectan información, la almacenan y la ponen a disposición de las estaciones de administración de red (NMS). Pueden desempeñar este rol distintos tipos de equipos, como routers, switches, hubs, estaciones de trabajo o impresoras, siempre que cuenten con un agente habilitado para responder a las solicitudes del gestor.

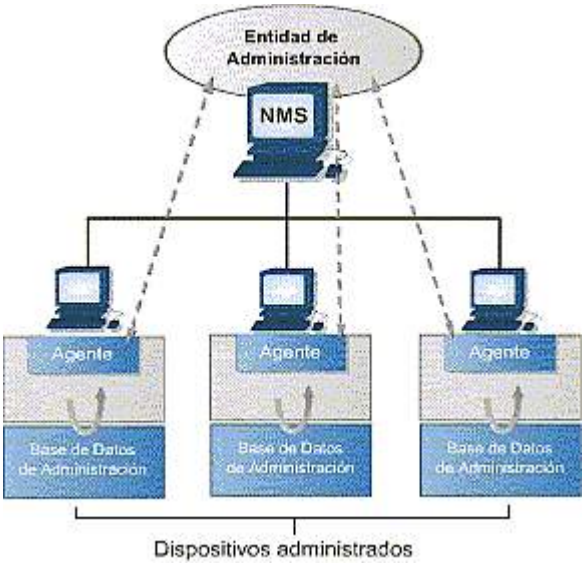
- **Agentes**

Un agente es un *software* que se ejecuta en los dispositivos administrados y se encarga de reportar información al gestor. Tiene conocimiento local del estado y los datos del

dispositivo, y los traduce a un formato compatible con el protocolo SNMP. De esta forma, el agente actúa como intermediario entre el dispositivo y la estación de administración de red, permitiendo el acceso remoto a la información relevante para el monitoreo.

La siguiente figura muestra la relación entre estos componentes.

Figura 2. Componentes de SNMP



Fuente: [imagen sin título sobre componentes de SNMP], (s.f.).

Consideraciones sobre seguridad

Las versiones SNMPv1 y SNMPv2 presentan limitaciones importantes en términos de seguridad, ya que no ofrecen mecanismos de autenticación ni confidencialidad. Por esta razón, no permiten establecer un entorno seguro para la gestión de red.

Una de las principales debilidades es la ausencia de encriptación en los datos transmitidos. En redes que utilizan medios compartidos, esto implica que una persona no autorizada podría interceptar la información y acceder a detalles sensibles, como modelos de dispositivos, direccionamiento IP o nombres de comunidades, lo que facilitaría la obtención de información crítica sobre la red.

Además, SNMPv1 y SNMPv2 carecen de un sistema de autenticación robusto. La única verificación que realizan los agentes SNMP antes de responder a una solicitud de la estación de administración es la coincidencia del nombre de comunidad. Esta situación representa un riesgo considerable, dado que muchas implementaciones utilizan por defecto nombres como *public* y *private*, donde el primero suele permitir acceso de solo lectura y el segundo, acceso con permisos de modificación. De este modo, un usuario malicioso podría intentar adivinar el nombre de la comunidad y acceder al dispositivo administrado, incluso con privilegios elevados.

La falta de autenticación también afecta la integridad de la comunicación entre las estaciones de administración y los dispositivos administrados. Un atacante podría modificar los datos enviados, falsear información o alterar reportes de estado, lo que comprometería seriamente la confiabilidad del sistema de monitoreo.

Frente a estas limitaciones, SNMPv3 incorpora mejoras significativas: ofrece autenticación mediante algoritmos como MD5 y garantiza la confidencialidad mediante encriptación de datos. Estas características permiten establecer un entorno más seguro, resistente a los ataques mencionados. Sin embargo, al tratarse de una versión más reciente del protocolo, es importante verificar que tanto los dispositivos como las estaciones de administración sean compatibles con SNMPv3 antes de su implementación.

Buenas prácticas en la configuración de SNMP

Para garantizar una implementación segura de SNMP, se recomienda adoptar una serie de buenas prácticas durante la configuración de los dispositivos. Siempre que sea posible, se debe utilizar la versión SNMPv3, ya que incorpora mecanismos de autenticación y cifrado que no están presentes en las versiones anteriores.

Es aconsejable configurar los dispositivos con comunidades de solo lectura, lo que reduce el riesgo de modificaciones no autorizadas en la red. Otorgar privilegios de escritura debe evitarse, salvo que exista una necesidad específica y se cuente con medidas de seguridad adicionales.

También es importante restringir el acceso a SNMP mediante filtros por dirección IP, de modo que solo hosts autorizados puedan realizar consultas al agente. En entornos con dispositivos críticos, puede resultar útil definir identificadores de objeto (OID) personalizados que permitan monitorear métricas específicas, adaptadas a los requerimientos operativos de cada organización.

Telemetría de red con SNMP

La telemetría de red se refiere a la recolección remota de datos sobre el funcionamiento de los dispositivos que integran una infraestructura de red. A través del protocolo SNMP, es posible capturar múltiples métricas que ofrecen visibilidad sobre el estado y el rendimiento de los equipos. Entre los datos más comunes se encuentran el uso de CPU y memoria, el tráfico entrante y saliente en cada interfaz —medido en bytes o en cantidad de paquetes—, los errores de red (como colisiones, paquetes descartados o errores de

CRC) y el estado general de los enlaces, incluyendo su disponibilidad.

Estos datos, recolectados de forma continua, permiten analizar el comportamiento de la red a lo largo del tiempo y detectar tanto patrones anómalos como señales tempranas de degradación en el servicio. Por ejemplo, un switch gestionado puede reportar, mediante SNMP, el tráfico de cada una de sus interfaces. Herramientas como LibreNMS permiten visualizar esa información en forma de gráficos, lo que facilita la identificación de tendencias y posibles cuellos de botella en determinados horarios o segmentos de la red.

Umbrales y alertas

Los umbrales son parámetros preestablecidos que permiten identificar condiciones anómalas dentro de una red. A través de SNMP, es posible aplicar umbrales sobre distintas métricas críticas y generar alertas automáticas cuando dichos valores se superan. Por ejemplo, si el tráfico en una interfaz excede el 85 % de su capacidad, el sistema puede notificar automáticamente a la estación de administración.

A continuación, se mencionan algunos ejemplos habituales de umbrales utilizados en tareas de monitoreo:

- utilización de CPU superior al 90 %;
- interfaces con errores que superen un porcentaje establecido;
- latencia medida indirectamente a través del comportamiento de la interfaz.

Para la implementación de estos mecanismos, se utilizan herramientas específicas orientadas a la gestión de alertas y al análisis de métricas, entre las cuales se destacan las siguientes:

LIBRENMS.	ZABBIX:	NAGIOS CON COMPLEMENTOS SNMP:
Solución gratuita para configurar alertas personalizadas mediante expresiones complejas.		

LIBRENMS.	ZABBIX:	NAGIOS CON COMPLEMENTOS SNMP:
plataforma de monitoreo con *dashboards* y plantillas SNMP para		

una amplia variedad de dispositivos.

LIBRENMS.	ZABBIX:	NAGIOS CON COMPLEMENTOS SNMP:
opción orientada a la generación de alertas y a la supervisión continua.		

Actividad práctica

Esta actividad es opcional y tiene como finalidad aplicar los conceptos vistos sobre monitoreo de tráfico mediante SNMP.

El objetivo consiste en monitorear el tráfico de un switch utilizando SNMP a través de LibreNMS.

Para llevar a cabo la actividad, se propone instalar LibreNMS en una máquina virtual, siguiendo la documentación oficial disponible en: https://docs.librenms.org/Installation/.

Luego, se deberá configurar SNMP en el switch. A modo de ejemplo, en un dispositivo Cisco puede utilizarse la siguiente configuración:

snmp-server community public RO

snmp-server location "Oficina Central"

snmp-server contact admin@empresa.com

Una vez completada la configuración, se deberá agregar el dispositivo en LibreNMS e identificar las interfaces activas desde la consola de monitoreo. Como parte del ejercicio, se recomienda crear una alerta que se dispare cuando alguna interfaz supere el 90 % de utilización de tráfico durante un período mayor a cinco minutos. Finalmente, se deberán evaluar los datos recolectados durante al menos una hora, con el objetivo de analizar el comportamiento del tráfico y validar el funcionamiento de las alertas configuradas.

1.3. Tableros de monitoreo de capacidad y uso

En redes, suele existir la tentación de medir aquello que resulta más sencillo de obtener, como la disponibilidad de los equipos, el uso de CPU o la utilización de interfaces. Estos indicadores aportan información relevante, pero no siempre alcanzan para explicar la experiencia real del usuario.

Cuando un servicio digital depende de múltiples tramos, lo que realmente importa para el negocio suele expresarse como un compromiso de calidad, ya sea mediante un SLA (*service level agreement*) o, en términos operativos, un SLO (*service level objective*). Aunque estos conceptos se formalizan de distintas maneras, comparten una misma lógica: transformar la calidad percibida en medidas observables.

Desde este enfoque, la observabilidad propone un criterio práctico: en lugar de medir todo, se priorizan aquellas señales que están alineadas con los resultados esperados. Un equipo puede estar disponible y, aun así, incumplir un objetivo de latencia. De igual modo, un enlace puede no estar saturado y, sin embargo, generar suficiente *jitter* como para degradar un servicio de voz.

Por este motivo, medir la experiencia implica observar indicadores de calidad y de comportamiento, entre los que se incluyen la latencia, el *jitter*, la pérdida de paquetes, las retransmisiones, la estabilidad de las rutas y los tiempos de respuesta de los servicios. Estas métricas adquieren un enfoque orientado al negocio cuando se definen por servicio —por ejemplo, «pagos en sucursal»—, con umbrales basados en el impacto y con tableros que puedan interpretarse sin necesidad de conocer en detalle cada dispositivo de la red.

Un error frecuente consiste en confundir un «SLA de red» con un «SLA de servicio». La red representa solo un tramo dentro de una cadena más amplia. Un SLA definido únicamente como disponibilidad de un router puede cumplirse y, aun así, generar reclamos, ya que el usuario no adquiere un router, sino una experiencia.

Las prácticas de observabilidad destacan que la conexión entre las señales técnicas y los resultados es lo que convierte a un tablero en una herramienta estratégica. Esa conexión no se logra únicamente mediante métricas, sino que requiere correlación y narrativa: mostrar qué cambió, a quién afectó y por qué.

Operativamente, una forma de avanzar consiste en mapear los servicios críticos y, para cada uno de ellos, seleccionar un conjunto mínimo de señales que permita evaluar su comportamiento. Entre estas señales se consideran:

1. un indicador de salud, asociado a la disponibilidad;

2. un indicador de capacidad, vinculado con la utilización o el *throughput;*
3. un indicador de calidad, como latencia, *jitter* o pérdida; y
4. un indicador de comportamiento, basado en flujos o *top talkers*.

Esta combinación permite sostener conversaciones tanto con el área de negocio como con los equipos técnicos, ya que facilita explicar si un problema está relacionado con saturación, con degradación de calidad en un tramo específico, con cambios de política o con comportamientos anómalos en la red. Este enfoque constituye el núcleo conceptual del módulo.

Diseño de *dashboards*: vistas de salud, de saturación y de investigación

Un tablero de monitoreo resulta útil en la medida en que responde a una audiencia específica y a un objetivo concreto. No todas las vistas cumplen la misma función ni deben responder a las mismas preguntas.

En la operación diaria, una vista de salud global debe permitir detectar rápidamente dónde existe riesgo. Su propósito es señalar enlaces en saturación, interfaces con errores, equipos con recursos comprometidos o sedes con pérdida de conectividad. Este tipo de vista no busca explicar las causas del problema, sino alertar y facilitar la priorización de acciones.

Una vista de saturación, en cambio, profundiza en los enlaces críticos y en su comportamiento. Allí se observan métricas como utilización, latencia, discards, jitter —cuando corresponde— y top talkers, especialmente si existe integración con flujos. Su función es vincular el síntoma con un posible mecanismo. Por ejemplo, si la latencia aumenta a medida que la utilización se acerca al límite, el tablero debe reflejar esa relación de forma clara.

La vista de investigación está orientada a responder preguntas puntuales y no previstas de antemano. Permite analizar qué ocurrió en un intervalo específico, identificar cambios repentinos, detectar flapping, reconocer la aparición de nuevos destinos o evaluar el crecimiento de determinados protocolos.

Desde la perspectiva de la observabilidad, estas vistas deben facilitar el pasaje progresivo de un nivel de información a otro, mediante mecanismos de exploración o *drill-down*, que permitan ir del indicador al detalle, del detalle al contexto y, finalmente, del contexto a la acción.

Un criterio adicional es la consistencia semántica. Si un *dashboard* utiliza nombres diferentes para el mismo enlace o etiqueta sedes de manera inconsistente, el analista pierde tiempo interpretando la información. En este punto, la observabilidad se vincula directamente con el gobierno de datos, que abarca aspectos como la nomenclatura, el uso de etiquetas, el inventario actualizado y la trazabilidad de los cambios.

En el contexto de las redes empresariales, los tableros de monitoreo (*dashboards*) constituyen herramientas fundamentales para visualizar, analizar y responder en tiempo real al estado de los dispositivos, enlaces y servicios. Cuando están bien diseñados, permiten a los equipos de infraestructura y seguridad cumplir distintos objetivos operativos, entre ellos:

- detectar cuellos de botella;

- identificar tendencias de saturación;
- anticipar fallas;
- justificar decisiones de escalabilidad o inversión.

Los tableros permiten sintetizar información proveniente de diversas fuentes, como flujos NetFlow o sFlow, SNMP, Syslog u otros mecanismos de recolección, y presentarla mediante gráficas claras y adaptadas a las necesidades de cada audiencia.

En la siguiente tabla se presentan algunos ejemplos de métricas utilizadas habitualmente en tableros de monitoreo de capacidad y uso.

Tabla 1. Métricas comunes para tableros de monitoreo de red

Categoría	Métrica clave
Uso de red	Ancho de banda por interfaz (entrante/saliente), top talkers
Capacidad de enlaces	Utilización media y picos de uso por VLAN o segmento

SNMP	CPU, memoria, estado de interfaces
Seguridad	Actividades anómalas, tasa de alertas, IDS
Conectividad	Disponibilidad de hosts o servicios críticos

Fuente: elaboración propia

Herramientas gratuitas para dashboards de red

Existen herramientas gratuitas, ampliamente adoptadas en la comunidad de redes y ciberseguridad, que permiten construir tableros de monitoreo para visualizar métricas, analizar tendencias y apoyar la toma de decisiones operativas. A continuación, se describen algunas de las más utilizadas.

- **Grafana** (https://grafana.com/)

Grafana es un motor de visualización de datos que se integra con múltiples fuentes, como Prometheus, InfluxDB o Zabbix. Permite crear dashboards interactivos y configurar alertas

personalizadas, lo que resulta especialmente útil para la visualización de series temporales asociadas al uso de red, la actividad de servicios y el consumo de recursos. En un escenario típico, un administrador puede configurar paneles para observar en tiempo real el tráfico entrante y saliente de los switches, correlacionándolo con alertas SNMP y con el estado de la CPU de los dispositivos.

- **Zabbix** (https://www.zabbix.com/)

Zabbix es una solución integral de monitoreo que abarca redes, servidores y aplicaciones dentro de una misma plataforma. Incluye componentes para la recolección, el procesamiento y la visualización de datos, e integra mecanismos como SNMP, traps, scripts personalizados y APIs externas. Por su enfoque unificado, resulta adecuada para pequeñas y medianas empresas que necesitan monitorear su infraestructura sin depender de múltiples sistemas separados. Además, ofrece dashboards preconfigurados y alertas automáticas basadas en umbrales definidos.

- **LibreNMS** (https://www.librenms.org/)

LibreNMS es una alternativa de código abierto orientada principalmente al monitoreo mediante SNMP. Incorpora funciones de autodetección de dispositivos, generación de alertas y dashboards personalizables. Su arquitectura permite escalar en entornos de distinto tamaño y visualizar históricos de consumo por interfaz y por dispositivo, lo que facilita el análisis de capacidad y la identificación de tendencias a lo largo del tiempo.

Buenas prácticas para tableros de red

Para que los tableros de monitoreo sean realmente útiles en entornos operativos, es recomendable seguir una serie de buenas prácticas orientadas a mejorar la lectura, la interpretación y la toma de decisiones.

Agrupar por función o rol.

Una práctica recomendada consiste en diseñar tableros diferenciados según la función que cumplen dentro de la infraestructura. Puede existir un dashboard orientado a switches core, otro enfocado en equipos de borde o distribución y uno específico para componentes críticos de seguridad, como firewalls o sistemas de detección de intrusiones.

Usar colores y alertas claras.

El uso consistente de colores permite interpretar el estado de la red de forma inmediata. Habitualmente, el color verde se asocia con condiciones normales de operación, el amarillo con situaciones de advertencia y el rojo con alertas o umbrales superados.

Visibilidad simplificada.

Un error frecuente es sobrecargar los dashboards con demasiadas métricas o visualizaciones complejas. Resulta más efectivo mostrar únicamente los indicadores relevantes para la toma de decisiones, priorizando aquellos que reflejan impacto operativo o riesgo para el servicio.

Documentar umbrales.

Documentar claramente qué representa cada alerta, cómo se calcula el umbral asociado y qué impacto tiene sobre el servicio permite mejorar la comprensión del tablero. Además, mantener trazabilidad sobre los cambios realizados facilita el análisis ante modificaciones futuras.

Ejemplo de panel práctico con Grafana

Como ejemplo de aplicación, se propone el diseño de un panel en Grafana orientado a visualizar métricas operativas clave de una red empresarial. En este escenario, se busca representar información vinculada con el estado y el rendimiento de los dispositivos y enlaces de borde.

Entre las métricas que se desean visualizar se encuentran:

- el tráfico SNMP por interfaz;
- la tasa de errores en puertos físicos;
- la disponibilidad de routers de borde.

Para implementar este panel, se pueden seguir una serie de pasos generales orientados a la integración de datos y a la construcción del dashboard. En primer lugar, se debe configurar Prometheus o InfluxDB como backend de métricas. Luego, es necesario instalar un SNMP Exporter como fuente de datos o, alternativamente, utilizar LibreNMS como origen de la información. A partir de estas fuentes, se crea un dashboard en Grafana que incluya distintos paneles, tales como:

- un gráfico de líneas que muestre el ancho de banda por interfaz;
- un indicador de latencia promedio;

- un semáforo de disponibilidad de enlaces.

Como material de apoyo, puede consultarse el tutorial oficial de Grafana orientado al uso de Prometheus como fuente de datos:

Fuente: **Grafana Labs**. (s. f.). *Prometheus data source*. https://grafana.com/docs/grafana/latest/datasources/prometheus/

Actividad práctica

Esta actividad es opcional y puede realizarse en un entorno virtual o físico. El objetivo es implementar un *dashboard* básico para monitorear una interfaz de red en un *router* o *switch*.

Requisitos

Para realizar la actividad, se requiere lo siguiente:

- PC con VirtualBox y Ubuntu.
- Instalación de Grafana e InfluxDB o Prometheus.
- Configuración de SNMP en un equipo (puede utilizarse otra instancia de Ubuntu simulada).

- Captura de tráfico SNMP.
- Visualización en el dashboard de ancho de banda y disponibilidad.

Paso a paso

A continuación, se describe un procedimiento general para implementar el *dashboard*:

- Instalar Grafana: *sudo apt install grafana*
- Instalar y configurar InfluxDB: *sudo apt install influxdb*
- Habilitar SNMP en el equipo remoto: *sudo apt install snmp snmpd*
- Crear un panel en Grafana con gráficos de tráfico.

Caso práctico: pyme que monitorea su red con *software* libre

Una empresa mediana cuenta con una infraestructura compuesta por dos switches, un *firewall* y tres servidores. Con el objetivo de mejorar la visibilidad sobre el funcionamiento de su red, decide implementar LibreNMS en un servidor con Ubuntu como plataforma central de monitoreo.

A partir de esta implementación, la empresa utiliza los *dashboards* preconfigurados de la herramienta para observar el comportamiento del tráfico durante las horas pico, identificar a los usuarios que consumen mayor ancho de banda y detectar errores de red asociados a enlaces físicos. Esta información permite al equipo de sistemas contar con una visión clara y actualizada del estado de la infraestructura, sin necesidad de recurrir a soluciones propietarias.

Como resultado, la organización logra reducir en un 30 % el tiempo promedio de detección de incidentes y mejora significativamente la planificación de ampliaciones y actualizaciones de la red, basándose en datos históricos y en tendencias de uso reales. Este caso ilustra cómo el uso de software libre puede aportar valor concreto en entornos empresariales con recursos limitados, fortaleciendo tanto la operación diaria como la toma de decisiones a mediano plazo.

1.4. Alertas y mantenimiento

Hablar de observabilidad en redes no implica simplemente contar con más herramientas, sino disponer de la capacidad de responder preguntas concretas a partir de evidencia. En la práctica, esa evidencia proviene de distintas familias de datos que se complementan entre sí. Algunas describen el estado y el rendimiento de la infraestructura; otras registran eventos puntuales; y otras reflejan el movimiento real del tráfico en la red.

Una red se vuelve más observable cuando estas fuentes pueden relacionarse entre sí y cuando sus alcances y limitaciones están claramente definidos. Cada tipo de dato presenta fortalezas, zonas de ceguera y costos asociados, por lo que comprender su rol resulta fundamental para un análisis efectivo.

Las métricas permiten describir valores numéricos a lo largo del tiempo, como disponibilidad, *throughput*, uso de CPU y memoria, utilización de enlaces, errores de interfaz, tiempos de respuesta, colas o descartes. Estas métricas son esenciales

para detectar procesos de degradación y anticipar fallas, ya que facilitan la identificación de tendencias y desviaciones respecto de una línea base.

No obstante, una observabilidad efectiva no se limita a la recolección de datos. También requiere la capacidad de interpretar eventos relevantes y actuar en consecuencia. Para ello, resulta indispensable configurar un sistema de alertas automatizado y definir un esquema de mantenimiento proactivo que acompañe la operación diaria.

En este apartado se abordan los criterios para definir alertas útiles, establecer umbrales adecuados, evitar la fatiga por alertas (*alert fatigue*) y aplicar buenas prácticas de mantenimiento preventivo, con el objetivo de asegurar la continuidad operativa y fortalecer la ciberresiliencia de la red.

Tipos de alertas

Los sistemas de monitoreo y observabilidad de red permiten definir distintos tipos de alertas según el nivel de impacto y la urgencia de respuesta requerida. En términos generales, estas alertas pueden clasificarse de la siguiente manera:

- **Alertas informativas**. Comunican eventos no críticos que aportan contexto operativo, como el inicio de sesión de un usuario fuera del horario habitual.

- **Alertas de advertencia** (*warning*): indican una condición que podría derivar en una falla si no se interviene, por ejemplo, un uso de CPU sostenido por encima del 80 % durante más de quince minutos.

- **Alertas críticas**: señalan situaciones que requieren atención inmediata, como la caída de un enlace troncal o la pérdida de conectividad de un dispositivo clave.

Definición de umbrales

Los umbrales son parámetros preestablecidos que permiten disparar alertas cuando se superan determinados valores, tal como se mencionó previamente. Para que su uso sea efectivo y aporte valor operativo, se recomienda considerar los siguientes criterios:

- Utilizar valores dinámicos basados en el comportamiento histórico de la red.
- Evitar umbrales excesivamente estrictos que puedan generar falsos positivos y aumentar la fatiga por alertas.
- Revisar los umbrales de forma periódica para ajustarlos a cambios en la infraestructura o en las cargas de trabajo.

Tabla 2. Ejemplos de umbrales relevantes

Métrica	Umbral sugerido
Uso de CPU	> 85 % durante 10 minutos
Pérdida de paquetes	> 1 % sostenida por más de 5 minutos
Latencia	> 150 ms entre segmentos críticos
Temperatura de hardware	> 80°C

Fuente: elaboración propia

Herramientas de alerta y notificación

En entornos de redes y seguridad, existen diversas herramientas que permiten generar alertas y notificaciones a partir de eventos, métricas y comportamientos anómalos. Estas soluciones se utilizan para advertir de forma oportuna sobre incidentes operativos o de seguridad y facilitar una respuesta adecuada. Entre las más utilizadas se encuentran:

- **Zabbix**. Solución de monitoreo que permite configurar alertas personalizadas y utilizar plantillas SNMP y NMS para distintos tipos de dispositivos.
- **Prometheus + Alertmanager**: combinación orientada a arquitecturas modernas y entornos en la nube, que permite definir reglas de alerta basadas en métricas y gestionar notificaciones de forma centralizada.
- **Grafana**, integrado con Prometheus o InfluxDB: ofrece paneles con alertas visuales y la posibilidad de enviar notificaciones a distintos canales, como correo electrónico, Slack o Discord.

- **Wazuh**: plataforma que incorpora alertas de seguridad basadas en logs en tiempo real, con capacidad de integración en entornos SIEM.

Canales de notificación

Para que las alertas cumplan su función operativa, es fundamental que lleguen al personal adecuado a través de canales confiables y acordes al nivel de urgencia. En la práctica, pueden utilizarse distintos medios de notificación, entre ellos, los siguientes:

- Correo electrónico institucional.
- Notificaciones móviles, como bots de Telegram.
- Integración con plataformas colaborativas como Slack o Microsoft Teams.
- *Dashboards* visuales centralizados para seguimiento continuo

Además, resulta conveniente establecer niveles de escalamiento, de modo que, si una alerta crítica no es

atendida dentro de un tiempo determinado, se derive automáticamente a niveles superiores de responsabilidad. Este enfoque ayuda a asegurar la respuesta oportuna ante incidentes relevantes y a reducir el riesgo de que una alerta importante pase inadvertida.

Prevención de fatiga por alertas

La fatiga por alertas se produce cuando los equipos reciben una cantidad excesiva de notificaciones irrelevantes, repetitivas o poco accionables, lo que termina reduciendo la efectividad del sistema de monitoreo. Para prevenir este problema, resulta necesario optimizar el esquema de notificaciones mediante el ajuste de umbrales, la priorización de eventos y la automatización de respuestas, de modo que las alertas aporten valor operativo y faciliten la toma de decisiones.

Entre las medidas más habituales para reducir la fatiga por alertas se encuentran:

- implementar mecanismos de correlación de eventos, evitando generar una alerta por cada evento aislado de bajo impacto:

- agrupar alertas similares para presentar una visión consolidada del problema;
- definir ventanas de mantenimiento durante las cuales ciertas alertas se supriman de forma controlada;
- revisar periódicamente las reglas de alertado con el objetivo de ajustarlas y optimizarlas según la evolución de la infraestructura.

Mantenimiento proactivo del sistema de monitoreo

Para asegurar que el sistema de observabilidad funcione de manera confiable y continua, resulta necesario definir un plan de mantenimiento proactivo que permita anticipar problemas y sostener la calidad de la información recolectada. Este plan debe contemplar distintas tareas periódicas, entre ellas, las siguientes:

- Actualización del *software* de monitoreo, con el fin de incorporar mejoras, corregir errores y reducir la exposición a vulnerabilidades.
- Verificación de la integridad de los agentes, asegurando que los dispositivos monitoreados

continúen reportando datos de forma correcta.

- Revisión de logs, para confirmar que se generan y almacenan adecuadamente y que no existen fallas en los mecanismos de registro.

- Pruebas de alertado, orientadas a validar que las notificaciones llegan a los destinatarios correspondientes dentro de los tiempos esperados.

- Ajuste de dashboards y métricas, especialmente a medida que se incorporan nuevos equipos, servicios o cambios en la infraestructura.

Actividad práctica sugerida

Esta actividad no es de carácter obligatorio y se propone como un ejercicio complementario para aplicar los conceptos desarrollados sobre alertas y notificación.

El objetivo consiste en configurar alertas críticas en Zabbix para monitorear el tráfico de una interfaz de red y la carga de CPU de un servidor Linux, así como establecer mecanismos

de notificación mediante correo electrónico institucional y Slack.

Para llevar a cabo la actividad, se sugiere seguir una secuencia de pasos generales.

- En primer lugar, se debe instalar Zabbix Server junto con el agente correspondiente.
- Luego, se agrega un *host* —en este caso, un servidor Linux— al sistema de monitoreo. A continuación, se define un trigger que dispare una alerta cuando el uso de CPU supere el 85 % durante un período de cinco minutos.
- Una vez configuradas las alertas, se establece el canal de notificación por correo electrónico institucional y, de forma complementaria, la integración con Slack.
- Para validar el funcionamiento del esquema de alertado, se recomienda generar carga artificial en el servidor, por ejemplo, mediante la herramienta stress, y verificar la recepción oportuna de las notificaciones.

- Finalmente, se sugiere documentar todo el proceso, incluyendo la configuración realizada y los resultados obtenidos.

CONTINUAR

Unidad 2. Registro de dispositivos

2.1. Syslog centralizado de equipos de red

El protocolo Syslog, definido en la RFC 5424, es un estándar ampliamente adoptado para el envío de mensajes de registro en sistemas informáticos, dispositivos de red —como switches, routers y firewalls—, servidores y aplicaciones. Su principal función es permitir la recopilación y centralización de logs en un servidor de registros, lo que facilita tareas de monitoreo, auditoría y análisis de seguridad.

Los eventos y logs aportan contexto y permiten establecer relaciones de causa y efecto dentro de la infraestructura. En ellos quedan registradas acciones como decisiones de firewall, procesos de autenticación, cambios de configuración, reinicios de servicios, alertas de sistemas de detección de intrusiones, registros del sistema operativo, mensajes de aplicaciones o eventos del plano de control.

Desde una perspectiva narrativa, los logs representan la historia de lo que el propio sistema declara haber ejecutado.

Las trazas, cuando están disponibles, permiten seguir una transacción a través de distintos componentes y comprender en qué tramo se introduce latencia o se producen errores. Si bien en redes tradicionales la instrumentación de trazas suele estar más asociada al ámbito de las aplicaciones, las prácticas modernas de observabilidad señalan que correlacionar trazas con datos de red contribuye a acelerar el diagnóstico en servicios digitales complejos.

Por último, los datos de tráfico —ya sea a nivel de paquetes o de flujos— describen el comportamiento real de la comunicación en la red. La captura de paquetes ofrece el máximo nivel de detalle y resulta adecuada para análisis forense, aunque su recolección a gran escala implica costos elevados. Los flujos, en cambio, resumen las comunicaciones según pares de origen y destino, protocolo, puerto y otras claves, lo que permite escalar el análisis y responder preguntas como quién se comunica con quién, cuánto tráfico se intercambia, durante cuánto tiempo y con qué patrón. Por este motivo, en muchas organizaciones los flujos se utilizan como primer nivel de análisis y se recurre a la captura de paquetes únicamente cuando la hipótesis requiere una inspección más profunda.

Beneficios del registro centralizado de logs

El uso de un sistema de registro centralizado aporta múltiples ventajas operativas y de seguridad, especialmente en entornos con infraestructuras distribuidas y heterogéneas. Entre los principales beneficios se destacan los siguientes:

Visibilidad integral.

A consolidar los registros provenientes de múltiples dispositivos, se obtiene una visión unificada del comportamiento de la red.

Auditoría y cumplimiento:

numerosas normativas y marcos de referencia, como ISO/IEC 27001, NIST o PCI-DSS, requieren el almacenamiento y la trazabilidad de los logs.

Análisis forense:

ante incidentes de seguridad, los logs centralizados permiten reconstruir eventos y comprender la secuencia de acciones ocurridas.

Alertas y correlación:

mediante la integración con sistemas SIEM, es posible correlacionar eventos y generar alertas automáticas ante comportamientos anómalos.

Componentes básicos de una solución Syslog

Una solución de Syslog centralizado se compone de distintos elementos que trabajan de manera conjunta para permitir la recolección, el transporte, el almacenamiento y el análisis de los mensajes de registro generados por los sistemas y dispositivos de red. Entre sus componentes principales se incluyen los siguientes:

- **Clientes Syslog**: dispositivos o sistemas que generan logs, como *routers*, *firewalls*, servidores u otros equipos de infraestructura.
- **Servidor Syslog**: componente encargado de recibir, almacenar y procesar los mensajes enviados por los clientes.

- **Protocolos de transporte**: mecanismos utilizados para el envío de *logs*, típicamente UDP en el puerto 514, aunque también pueden emplearse TCP o TLS para mejorar la fiabilidad y la seguridad.

- **Niveles de severidad**: categorías definidas por Syslog que indican la gravedad del mensaje, numeradas de 0 a 7, donde 0 corresponde a emergencia y 7 a depuración.

- **Facilidades**: categorías que identifican el origen del mensaje, como kern, auth, daemon, entre otras.

Herramientas recomendadas

En la siguiente tabla se presentan algunas herramientas ampliamente utilizadas para implementar soluciones de registro centralizado de logs en entornos de red.

Tabla 2. Herramientas recomendadas para la centralización y análisis de logs Syslog

Herramienta	Descripción

rsyslog	Reemplazo moderno del syslogd tradicional. Muy configurable y estable. Presente por defecto en muchas distribuciones Linux.
syslog-ng	Alternativa potente con soporte avanzado para filtros y formatos.
Graylog	Plataforma de análisis de logs con visualización, búsquedas y alertas. Requiere Java y MongoDB.
Logstash + ELK	Parte del stack ELK (Elasticsearch, Logstash, Kibana). Ideal para análisis de grandes volúmenes de logs.
Wazuh	Además de ser un SIEM/XDR, puede centralizar logs de red.

Fuente: elaboración propia

Ejemplo de implementación con rsyslog

A modo ilustrativo, se presenta un ejemplo básico de implementación de un sistema Syslog centralizado utilizando rsyslog como servidor receptor y un dispositivo Cisco como cliente emisor de logs. El objetivo es mostrar el flujo general de configuración, tanto del lado del servidor como del dispositivo de red.

1 Configuración en el servidor receptor (Ubuntu/Linux)

En el servidor que actuará como receptor de logs, se debe editar el archivo de configuración principal de rsyslog:

sudo nano /etc/rsyslog.conf

Dentro del archivo, se habilita la recepción de mensajes Syslog a través de la red utilizando UDP, mediante la carga del módulo correspondiente y la definición del puerto de escucha:

module(load="imudp") # habilita recepción por UDP

input(type="imudp" port="514")

Una vez realizados los cambios, se guarda el archivo y se reinicia el servicio para que la configuración tenga efecto:

sudo systemctl restart rsyslog

2 Configuración en un dispositivo Cisco (cliente Syslog)

En el dispositivo Cisco que enviará los logs al servidor centralizado, se accede al modo de configuración y se define la dirección del servidor Syslog, junto con el nivel de severidad de los mensajes a enviar:

conf t

logging 192.168.1.10

logging trap informational

Con esta configuración, el dispositivo enviará al servidor Syslog ubicado en la dirección IP 192.168.1.10 todos los eventos con nivel informativo o superior, lo que permite centralizar mensajes relevantes para monitoreo, auditoría y análisis de incidentes.

Buenas prácticas

Para garantizar un funcionamiento confiable, seguro y sostenible de una solución de Syslog centralizado, es recomendable adoptar una serie de buenas prácticas orientadas tanto a la seguridad como a la gestión operativa del sistema. Entre las más relevantes podemos mencionar las siguientes:

- **Uso de protocolos seguros**. Emplear Syslog sobre TCP con cifrado TLS para proteger la confidencialidad e integridad de los mensajes transmitidos.

- **Organización de los registros**: separar los logs en carpetas o archivos distintos según el tipo de dispositivo o servicio, lo que facilita el análisis y la auditoría.

- **Rotación de logs**: implementar mecanismos de rotación mediante herramientas como logrotate para evitar el crecimiento descontrolado del almacenamiento.

- **Políticas de retención**: definir períodos claros de conservación de logs, como 30, 90 o 180

días, de acuerdo con la sensibilidad de la información y los requerimientos normativos.

- **Monitoreo del servidor Syslog**: supervisar de forma continua la carga del servidor, ya que el volumen de mensajes puede incrementarse rápidamente y afectar su rendimiento.

2.2. Normalización y enriquecimiento

El registro centralizado de eventos de red no se limita a la simple recopilación de logs, sino que alcanza su verdadero valor cuando los datos recolectados se transforman en información útil para el análisis operativo y de seguridad. En este contexto, intervienen dos procesos fundamentales dentro de las prácticas de observabilidad y seguridad de red:

- **Normalización**. Proceso mediante el cual los eventos se convierten a un formato unificado y legible, independientemente del origen o del tipo de dispositivo que los genera.

- **Enriquecimiento**. Incorporación de información contextual adicional que permite interpretar los eventos de forma más rápida y precisa.

La aplicación conjunta de estos procesos facilita la correlación entre eventos provenientes de distintas fuentes, mejora la capacidad de detección de incidentes y amenazas, y reduce la carga operativa asociada a las tareas de análisis manual.

Normalización: qué es y por qué es clave

Los dispositivos de red generan *logs* con formatos muy disímiles, lo que dificulta su análisis cuando se recopilan desde múltiples fuentes. Por ejemplo, un *router* Cisco puede emitir mensajes en formato Syslog con estructuras propias; un firewall pfSense registra eventos mediante códigos abreviados específicos; un sistema Windows exporta eventos en formatos como XML o EVT; y herramientas como *iptables* o *auditd* en Linux utilizan campos y esquemas particulares.

La normalización consiste en traducir todos estos formatos heterogéneos a un esquema común y coherente, que permita comparar, correlacionar y analizar los eventos de manera uniforme, independientemente de su origen. Gracias a este proceso, los logs se vuelven legibles para las herramientas de análisis y para los equipos operativos, reduciendo ambigüedades y acelerando la interpretación.

De forma general, el esquema de normalización incluye un conjunto básico de campos que se repiten en la mayoría de los eventos, y que sirven como base para el análisis posterior.

Tabla 3. Campos comunes en un esquema de normalización de logs

Campo normalizado	Descripción
timestamp	Fecha y hora del evento
source_ip, dest_ip	IP de origen y destino
protocol, port	Protocolo y puertos utilizados
event_type	Tipo de evento (login, deny, scan, etc.)
device_type	Tipo de dispositivo que generó el log
message	Mensaje descriptivo

Fuente: elaboración propia

Herramientas de normalización recomendadas

Para implementar procesos de normalización en entornos de red y seguridad, resulta conveniente utilizar motores que permitan un procesamiento flexible de eventos y la transformación de logs provenientes de múltiples fuentes. Estas herramientas facilitan la unificación de formatos y la preparación de los datos para su análisis y correlación.

- **Logstash**: componente del ELK Stack que incorpora filtros de normalización y transformación altamente configurables.
- **Wazuh Manager**: plataforma que normaliza eventos de forma automática a partir de diversas fuentes, con foco en seguridad y cumplimiento.
- **Graylog**: solución que permite definir pipelines de procesamiento para normalizar y enriquecer logs antes de su almacenamiento.
- **Fluentd**: motor de recolección y procesamiento que ofrece filtros configurables para aplicar transformaciones sobre los eventos.

Enriquecimiento de datos

El enriquecimiento de datos consiste en añadir contexto adicional a los eventos normalizados, con el objetivo de facilitar su interpretación y acelerar el análisis operativo y de seguridad. Este proceso permite que un mismo evento aporte más información sin necesidad de recurrir a fuentes externas en cada análisis. Entre las formas más habituales de enriquecimiento se incluyen las siguientes.

- **Geolocalización.** Consiste en añadir información geográfica asociada a la dirección IP de origen, lo que permite detectar accesos sospechosos provenientes de ubicaciones inusuales o no esperadas.

- **Información de *host*.** Permite vincular direcciones IP con datos internos de la organización, como nombres de *host*, áreas o usuarios responsables. Por ejemplo, una IP puede asociarse a un servidor específico o a un equipo perteneciente a un determinado departamento.

- **Categorización de eventos**. Implica asignar una categoría lógica a cada evento para facilitar su correlación y análisis posterior.

Algunas categorías habituales incluyen *authentication_failure*, *malware_detected, port_scan* y *configuration_change*.

- **Enlace con inteligencia de amenazas.** Consiste en comparar direcciones IP, dominios o hashes con fuentes externas de inteligencia de amenazas, como AbuseIPDB, VirusTotal o feeds basados en STIX/TAXII. A partir de esta comparación, los eventos pueden etiquetarse con indicadores adicionales, por ejemplo: *"ioc_match": true, "threat_type": "Command-and-Control"*

Este tipo de enriquecimiento permite priorizar incidentes, mejorar la detección temprana de amenazas y reducir significativamente el tiempo necesario para comprender el contexto de un evento.

Ejemplo práctico de pipeline de normalización y enriquecimiento con Logstash

A continuación, se presenta un ejemplo simplificado de un pipeline de Logstash que combina procesos de normalización y enriquecimiento sobre eventos Syslog recibidos por red.

```
input {
 udp {
  port => 514
  type => "syslog"
 }
}
filter {
 grok {
                match => { "message" => "%{SYSLOGTIMESTAMP:timestamp} %{HOSTNAME:hostname} %{DATA:program}: %{GREEDYDATA:log_message}" }
 }
 geoip {
  source => "source_ip"
```

```
 }

 if [program] == "sshd" {

   mutate { add_field => { "event_type" => "authentication" } }

 }

}

output {

 elasticsearch {

   hosts => ["localhost:9200"]

   index => "normalized-logs"

 }

}
```

A partir de los procesos de normalización y enriquecimiento, es posible observar una serie de ventajas operativas que mejoran de forma significativa la capacidad de análisis y respuesta ante eventos de red y de seguridad.

- **Correlación rápida**: facilita la vinculación de eventos provenientes de distintos sistemas y fuentes.
- **Alertas inteligentes**: permite definir reglas de detección más avanzadas y precisas, basadas en múltiples criterios.
- **Visualización clara**: mejora la construcción de dashboards y reportes comprensibles para distintos perfiles de usuario.
- **Reducción de falsos positivos**: el contexto adicional incorporado a los eventos incrementa la precisión del alertado.
- **Automatización de respuestas**: los sistemas SOAR o XDR requieren datos estructurados para ejecutar acciones automáticas de forma confiable.

Actividad práctica

Esta actividad no es de carácter obligatorio; sin embargo, sirve para aplicar de manera práctica los conceptos de

normalización y enriquecimiento de logs trabajados en la unidad.

Objetivo

Normalizar y enriquecer logs de un *firewall* pfSense y de un servidor Linux en una vista unificada.

Para desarrollar la actividad, se propone trabajar con el siguiente entorno de pruebas:

- pfSense como *firewall* virtual.
- Ubuntu Server con syslog activado.
- Servidor ELK Stack o Wazuh.

Una vez desplegado el entorno, se deberán realizar las siguientes tareas:

- Configurar la recolección de logs en Logstash o Wazuh;
- Crear una regla para unificar campos comunes, como direcciones IP y puertos;

- Enriquecer los eventos incorporando información de geolocalización y nombre de host;

- Visualizar los logs procesados en Kibana o en el panel de Wazuh.

Estos mecanismos facilitan la identificación y la clasificación de desvíos respecto del comportamiento normal de la red. A partir de la disponibilidad de datos de flujos, el análisis suele comenzar con una pregunta central: «¿qué cambió?». Los cambios pueden manifestarse en el volumen de tráfico, los destinos, los puertos utilizados, la duración de las comunicaciones o la diversidad de las fuentes. Para responder a estas preguntas, se emplean rankings, series temporales y técnicas de agrupamiento.

En el caso de picos legítimos, suele observarse un número limitado de conversaciones de gran volumen, generalmente asociadas a tareas esperables, como actualizaciones de software, procesos de sincronización o backups. Aun así, estos picos pueden resultar problemáticos si ocurren en horarios sensibles o sobre enlaces críticos. El análisis de flujos permite abordar estas situaciones con evidencia objetiva y ubicar con precisión el origen del tráfico.

En cambio, cuando se trata de patrones sospechosos, suelen aparecer señales recurrentes que permiten orientar la investigación, entre ellas:

- un escaneo horizontal, que se manifiesta como una única fuente contactando múltiples destinos a través del mismo puerto;
- un escaneo vertical, caracterizado por una fuente que contacta un mismo destino utilizando numerosos puertos distintos;
- un ataque DDoS de tipo volumétrico, que puede reflejarse como un aumento abrupto de tráfico proveniente de muchas fuentes hacia un único destino;
- una posible exfiltración de información, que suele evidenciarse como un crecimiento inusual y sostenido del tráfico saliente hacia un destino nuevo.

2.3. Búsquedas útiles (*top talkers*, *ports*)

El concepto de *top talkers* (principales conversadores) se utiliza para identificar los dispositivos, direcciones IP o flujos que generan la mayor cantidad de tráfico en una red, ya sea

en términos de bytes transferidos, cantidad de paquetes o tasa de transmisión. En la práctica, estos elementos suelen ser los mayores consumidores de ancho de banda dentro de un sistema o segmento de red.

En tareas de monitoreo, el análisis de *top talkers* resulta útil para detectar cuellos de botella, identificar usos excesivos de recursos y reconocer comportamientos anómalos que pueden estar asociados a incidentes de seguridad, como infecciones de malware o actividades no autorizadas. En términos simples, se trata de identificar qué equipos o comunicaciones están «hablando» más dentro de la red.

Este tipo de búsquedas se emplea con distintos fines operativos. En el monitoreo del rendimiento y la optimización de la red, permiten observar qué dispositivos o aplicaciones consumen más recursos, facilitando la localización de saturaciones y ayudando a comprender qué está afectando el desempeño general. Desde el punto de vista de la seguridad, contribuyen a detectar anomalías, como un equipo que comienza a enviar grandes volúmenes de datos de manera inesperada, lo que podría indicar un compromiso de seguridad, la presencia de malware o la participación en un ataque distribuido.

También resultan valiosas para el diagnóstico y la resolución de problemas, ya que permiten investigar por qué un enlace o el acceso a internet presenta lentitud, identificando si existen descargas masivas, copias de seguridad fuera de horario o transferencias no planificadas. Asimismo, el análisis de *top talkers* aporta información relevante para la planificación de capacidad, al ofrecer datos concretos sobre el uso real de la red y facilitar decisiones vinculadas con la ampliación de ancho de banda o la reasignación de recursos.

La identificación de *top talkers* se basa en métricas como el volumen de bytes transferidos, la cantidad de paquetes o la tasa de tráfico entre distintos puntos de la red. Para ello, se utilizan mecanismos de recopilación y análisis de datos, como flujos de red, que permiten generar informes y visualizar patrones de comunicación de forma agregada. En síntesis, los *top talkers* representan a los principales consumidores de ancho de banda dentro de una red, y su identificación constituye una práctica fundamental tanto para la gestión operativa como para la seguridad y la planificación de la infraestructura.

Otra búsqueda útil consiste en el análisis de puertos, es decir, en examinar la red para identificar puertos abiertos y los servicios que se ejecutan sobre ellos. Las herramientas avanzadas de análisis de puertos permiten evaluar tanto

puertos TCP como UDP, que funcionan como mecanismos de comunicación entre procesos de red. Este tipo de análisis facilita asociar los puertos detectados con servicios conocidos y comprender qué aplicaciones están operando en cada dispositivo.

Desde el punto de vista de la administración de redes, el análisis de puertos permite identificar servicios activos y relacionarlos con el espacio de direcciones IP de la organización. Mediante analizadores de IP, es posible obtener información detallada sobre una dirección específica y su comportamiento en las interfaces de red, lo que contribuye a una mejor visibilidad de la infraestructura.

Por otro lado, el escaneo de puertos es una técnica comúnmente utilizada por actores maliciosos para descubrir puertas abiertas o posibles debilidades en una red. A través de este tipo de escaneo, se busca determinar qué puertos están abiertos y si se encuentran recibiendo o enviando datos. Asimismo, las respuestas obtenidas pueden revelar la presencia de dispositivos de seguridad, como firewalls, y aportar información sobre su configuración.

Cuando se envía un mensaje a un puerto determinado, la respuesta recibida permite inferir si el puerto está activo y si existe alguna vulnerabilidad potencial que pueda ser explotada. No obstante, estas mismas técnicas también son empleadas de forma legítima por las organizaciones para evaluar su propia postura de seguridad.

Las empresas pueden realizar escaneos controlados enviando paquetes a puertos específicos y analizando las respuestas obtenidas con el objetivo de detectar configuraciones inseguras o servicios innecesarios expuestos. Para ello, se utilizan herramientas ampliamente conocidas como Nmap o Netcat, que permiten verificar el estado de los puertos y contribuir a que la red y los sistemas se mantengan adecuadamente protegidos.

Un puerto es un punto lógico en una computadora a través del cual se produce el intercambio de información entre los programas y otros dispositivos o sistemas en Internet. Para garantizar la coherencia y simplificar los procesos de programación, a estos puertos se les asignan números. Dichos números, en conjunto con una dirección IP, constituyen información fundamental que los proveedores de servicios de Internet utilizan para gestionar y responder a las solicitudes de comunicación.

Los números de puerto van de 0 a 65 535 y se clasifican según su uso. Los puertos numerados del 0 al 1023 se denominan «puertos bien conocidos» y suelen estar reservados para servicios estándar de Internet, aunque también pueden destinarse a fines específicos. Estos puertos son asignados por la Internet Assigned Numbers Authority (IANA) y se encuentran asociados a servicios ampliamente utilizados y mantenidos por organizaciones y proveedores de software.

La gestión de los puertos se realiza principalmente a través de dos protocolos. Por un lado, el protocolo de control de transmisión (TCP) define cómo se establece y mantiene una comunicación confiable entre aplicaciones. Por otro, el protocolo de datagramas de usuario (UDP) se utiliza en escenarios que requieren baja latencia y tolerancia a la pérdida de paquetes. Entre los puertos más utilizados se encuentran los siguientes:

- **Puerto 20 (UDP)**: protocolo de transferencia de archivos (FTP), utilizado para la transferencia de datos.

- **Puerto 22 (TCP)**: protocolo Secure Shell (SSH), empleado para accesos remotos seguros,

reenvío de puertos y transferencia de archivos.

- **Puerto 23 (TCP)**: protocolo Telnet, utilizado para comunicación remota sin cifrado.
- **Puerto 53 (UDP)**: sistema de nombres de dominio (DNS), encargado de traducir nombres de dominio en direcciones IP.
- **Puerto 80 (TCP)**: protocolo de transferencia de hipertexto (HTTP), utilizado por la World Wide Web.

Los puertos numerados del 1024 al 49 151 se conocen como «puertos registrados» y se encuentran asociados a aplicaciones o servicios específicos desarrollados por empresas de software. Finalmente, los puertos que van del 49 152 al 65 535 se consideran puertos dinámicos o privados, y pueden ser utilizados de manera temporal por aplicaciones cliente para establecer comunicaciones en Internet.

Técnicas de escaneo de puertos

Además del análisis de puertos con fines de administración y diagnóstico, existen distintas técnicas de escaneo que permiten identificar sistemas activos, puertos abiertos y

posibles debilidades en una red. Estas técnicas pueden emplearse tanto de forma legítima, por ejemplo en tareas de auditoría y evaluación de seguridad, como con fines maliciosos. A continuación, se describen algunas de las técnicas de escaneo de puertos más conocidas.

Un escaneo de puertos analiza los paquetes enviados a distintos números de puerto de destino utilizando diversos métodos. Entre las técnicas más habituales se encuentran las siguientes:

- **Escaneos de *ping***. Se consideran la técnica más simple de escaneo. También se conocen como solicitudes del protocolo de mensajes de control de Internet (Internet Control Message Protocol, ICMP). Consisten en enviar múltiples solicitudes ICMP a distintos servidores para verificar si responden. Los administradores pueden utilizar este tipo de escaneo para tareas de diagnóstico, aunque los firewalls suelen bloquear o deshabilitar este tipo de mensajes.

- **Escaneo de vainilla**: es una técnica básica que intenta conectarse a todos los puertos disponibles enviando solicitudes de sincronización (SYN). Cuando se recibe una

respuesta SYN-ACK, el escáner completa la conexión con un ACK. Este método es preciso, pero fácilmente detectable, ya que los firewalls suelen registrar las conexiones completas.

- **Escaneo SYN**: también conocido como escaneo semiabierto, envía un paquete SYN al puerto de destino y espera una respuesta SYN-ACK. En caso de recibirla, el escáner no completa la conexión, por lo que esta no queda registrada como una sesión TCP establecida. Es una técnica rápida y comúnmente utilizada para identificar puertos abiertos.

- **Escaneos XMAS y FIN**: son métodos más discretos. El escaneo XMAS recibe su nombre por el conjunto de banderas activadas en el paquete, que, al observarse en un analizador de protocolos, recuerda a un árbol de Navidad. Estos paquetes pueden revelar información sobre el estado de los puertos y el comportamiento del firewall. El escaneo FIN, por su parte, envía un paquete con la bandera FIN activada a un puerto específico, y la respuesta obtenida permite inferir si el puerto está abierto o cerrado.

- **Escaneo de rebote FTP**: esta técnica permite ocultar la ubicación del escáner utilizando un servidor FTP como intermediario para reenviar los paquetes hacia el objetivo.

- **Escaneo de barrido**: es una técnica preliminar que envía tráfico a un puerto específico a través de múltiples direcciones dentro de una red con el objetivo de identificar qué sistemas están activos. No proporciona información detallada sobre los puertos, pero permite detectar hosts en uso.

Escaneo de puertos frente a escaneo de red

El escaneo de red es un proceso orientado a identificar los hosts activos dentro de una red y asociarlos a sus respectivas direcciones IP. Este paso suele realizarse antes de ejecutar un escaneo de puertos, ya que permite delimitar qué sistemas se encuentran disponibles para un análisis posterior más detallado.

Este procedimiento también se conoce como descubrimiento de hosts y, en muchos casos, constituye la primera etapa de una evaluación de seguridad o de una auditoría de red. Para llevarlo a cabo, se utilizan

principalmente dos tipos de protocolos: el protocolo de resolución de direcciones (ARP) y distintos tipos de mensajes ICMP. El escaneo mediante ARP permite asociar direcciones IP con direcciones MAC y resulta efectivo para identificar hosts activos dentro de una red de área local (LAN). Sin embargo, este método solo funciona dentro de la red interna, por lo que requiere acceso directo a ella.

Para el escaneo de redes fuera de la LAN, se emplean distintos mensajes ICMP, como solicitudes de eco, marca de tiempo o información de dirección. La identificación de hosts depende de la recepción de respuestas desde los sistemas objetivo. La ausencia de respuesta puede indicar que no existe un host en la dirección consultada o que la solicitud fue bloqueada por un firewall o un filtro de paquetes.

Una vez finalizado el escaneo de red y compilada la lista de hosts disponibles, se procede al escaneo de puertos. En esta etapa, se analizan los puertos de cada host con el fin de identificar qué servicios se encuentran expuestos. Como resultado, los puertos suelen clasificarse como abiertos, cerrados o filtrados, lo que permite comprender mejor la superficie de ataque o el estado de la configuración de seguridad de la red.

¿Cómo prevenir ataques de escaneo de puertos?

El escaneo de puertos es una técnica ampliamente utilizada por actores maliciosos para identificar servidores vulnerables y evaluar el nivel de seguridad de una organización. A través de este tipo de análisis, es posible inferir la existencia y efectividad de firewalls, detectar servicios expuestos y, en algunos casos, ocultar el origen del atacante mediante determinados métodos TCP.

Durante un escaneo, se observa cómo responden los puertos ante distintas solicitudes, lo que permite comprender qué mecanismos de protección están activos y qué servicios se encuentran accesibles. Por este motivo, la prevención de ataques de escaneo de puertos requiere una combinación de visibilidad, monitoreo continuo y controles de seguridad adecuados.

Una estrategia efectiva de prevención se apoya en el uso de inteligencia de amenazas actualizada y alineada con el panorama de riesgos en evolución. Asimismo, resulta necesario contar con software de seguridad robusto, herramientas de análisis y sistemas de alertas que permitan monitorear el estado de los puertos y detectar comportamientos sospechosos antes de que un atacante acceda a la red. Entre las herramientas utilizadas con fines defensivos se encuentran analizadores de IP y utilidades de escaneo controlado, como Nmap o Netcat.

Además, pueden aplicarse otros mecanismos de defensa complementarios:

- **Uso de un *firewall* robusto**. Un *firewall* permite controlar el acceso a la red privada, gestionar la visibilidad de los puertos y detectar intentos de escaneo en curso para bloquearlos oportunamente.

- **Implementación de envoltorios TCP**: estos mecanismos ofrecen flexibilidad para permitir o denegar el acceso a servicios según direcciones IP o nombres de dominio específicos.

- **Identificación periódica de exposiciones innecesarias**: mediante verificadores o escáneres de puertos, las organizaciones pueden detectar puertos abiertos que no resultan necesarios y revisar de forma regular sus sistemas para identificar posibles debilidades o configuraciones inseguras.

2.4. Retención y privacidad

Características clave de las herramientas de observabilidad de red

Las soluciones de observabilidad de red suelen adaptarse a las necesidades particulares de cada organización. No obstante, la mayoría de las herramientas comparten un conjunto de características y capacidades comunes:

- **Recopilación, retención y análisis de datos.** Las soluciones de observabilidad de red recopilan, almacenan y analizan datos de telemetría, incluidos detalles a nivel de paquete, registros de flujo y métricas de dispositivos provenientes de diversas fuentes de la red. Las herramientas modernas de observabilidad se integran de manera fluida con hardware de red, redes definidas por software (SDN) y plataformas en la nube para garantizar una recopilación de datos integral. El análisis de estos datos permite comprender mejor el funcionamiento y las tendencias de la red, simplificar la generación de informes y el cumplimiento, y realizar análisis exhaustivos de causa raíz.

- **Paneles de control y visualizaciones.** Las herramientas de observabilidad de red

proporcionan paneles de control y mecanismos de visualización que presentan datos complejos en formatos intuitivos. Los mapas de calor, los diagramas de flujo de tráfico y las métricas de rendimiento en tiempo real permiten evaluar rápidamente el estado general de la red.

- **Alertas y notificaciones.** Las alertas son notificaciones automatizadas que se activan en función de condiciones o umbrales específicos de la red. Las soluciones de observabilidad incorporan mecanismos de alertado que distinguen entre incidentes críticos y anomalías menores, lo que contribuye a reducir la fatiga por alertas y a priorizar los problemas de mayor impacto.. Junto con las notificaciones, que informan a las partes interesadas sobre eventos relevantes, las alertas permiten abordar de forma proactiva los problemas de red y mantener una alta disponibilidad de los sistemas.

- **Análisis continuo del rendimiento.** El análisis continuo del rendimiento implica la medición permanente de métricas clave en distintos segmentos de la red. Estas evaluaciones proporcionan información sobre el

comportamiento y las tendencias a lo largo del tiempo, lo que facilita la toma de decisiones vinculadas con actualizaciones, optimizaciones y planificación de capacidad.

- **Asignación de topología.** El mapeo topológico ofrece representaciones visuales de la arquitectura de la red y muestra cómo se interconectan los distintos componentes en entornos locales, virtuales y en la nube. En muchos casos, estas representaciones se actualizan de manera dinámica a medida que se producen cambios, lo que permite contar con una visión actualizada de la red. Estas características contribuyen a mejorar la planificación al facilitar la comprensión de cómo los cambios impactan en la arquitectura general.

- **IA y análisis predictivo.** Las tecnologías de inteligencia artificial y *machine learning* permiten analizar grandes volúmenes de datos generados por las redes e identificar patrones anómalos y comportamientos del sistema. Estas capacidades facilitan la correlación de datos de telemetría entre dispositivos y capas, lo que agiliza el análisis de causa raíz. Mediante modelos de *machine learning*, las soluciones de observabilidad

también pueden aplicar análisis predictivo para anticipar problemas de rendimiento antes de que se manifiesten de forma crítica.

- **Monitorización de cambios.** La monitorización de cambios permite realizar un seguimiento en tiempo real de modificaciones en la red, como actualizaciones de configuración, parches de software o cambios de hardware, y evaluar su impacto en el rendimiento. Este enfoque facilita la identificación de interrupciones o degradaciones asociadas a nuevas configuraciones, especialmente cuando los datos de cambios se correlacionan con métricas de rendimiento.

- **Integración con otras herramientas.** Las herramientas de observabilidad de red suelen integrarse con otros sistemas de monitoreo, registro y alertado. Estas integraciones permiten obtener una visión más amplia de la infraestructura tecnológica y mejorar la visibilidad general de la red.

Cuestionario de repaso (con respuestas)

A continuación, se presenta una serie de preguntas con respuestas breves, pensadas como disparadores para que el alumno las desarrolle con mayor profundidad. El objetivo es que funcionen como material de repaso y apoyo al autoaprendizaje.

- **¿Qué es SNMP y para qué se utiliza?**
  Es un protocolo que permite monitorear y gestionar dispositivos de red mediante la recolección de métricas como tráfico, estado y errores.

- **¿Cuál es la comunidad por defecto más utilizada en SNMP v2?**
  public

- **¿Qué puerto utiliza SNMP por defecto?**
  UDP 161

- **¿Qué herramienta permite visualizar el tráfico en tiempo real sin configuración compleja?**
  Netdata

- **¿Qué componente de Linux recibe los logs del sistema?**
  rsyslog

- **¿Qué tipo de datos puede generar SNMP?**
  Telemetría sobre interfaces, tiempo de actividad, uso de CPU, temperatura, entre otros.

- **¿Qué herramienta permite crear dashboards a partir de múltiples fuentes de datos?**
  Grafana

- **¿Qué comando se utiliza para consultar datos mediante SNMP?**
  snmpwalk

- **¿Cómo se llama el protocolo estándar para centralizar *logs*?**
  Syslog

- **¿Qué es la normalización de *logs*?**
  Es el proceso de transformar logs provenientes de distintos formatos a un esquema homogéneo que facilite su análisis y búsqueda.

- **¿Qué es un «*top talker*»?**
  Un dispositivo o dirección IP que genera el mayor volumen de tráfico dentro de una red.

- **¿Por qué resulta útil separar los *logs* por origen?**

Porque facilita la identificación rápida de problemas en componentes o sistemas específicos.

- **¿Qué función cumple Telegraf en este contexto?**
  Actúa como agente recolector de datos, por ejemplo, mediante SNMP, para su envío a sistemas como InfluxDB o Grafana.

- **¿Qué representa una alerta de umbral en Grafana?**
  Una condición definida que, al superarse, dispara una notificación.

- **¿Qué significa enriquecer un *log*?**
  Agregar información adicional al mensaje original, como clasificación por severidad, área o dispositivo.

CONTINUAR

Referencias

Grafana Labs. (s. f.). *Prometheus data source*. https://grafana.com/docs/grafana/latest/datasources/prometheus/

Referencias bibliográficas de consulta

Fortinet. (s. f.). Cyber threat intelligence. https://www.fortinet.com/lat/resources/cyberglossary/cyber-threat-intelligence

Fortinet. (s. f.). Internet control message protocol (ICMP). https://www.fortinet.com/lat/resources/cyberglossary/internet-control-message-protocol-icmp

Fortinet. (s. f.). What is ARP? https://www.fortinet.com/lat/resources/cyberglossary/what-is-arp

Fortinet. (s. f.). What is DNS? https://www.fortinet.com/lat/resources/cyberglossary/what-is-dns

Fortinet. (s. f.). What is HTTPS? https://www.fortinet.com/lat/resources/cyberglossary/what-is-https

Fortinet. (s. f.). What is a port scan? How to prevent port scanning attacks. https://www.fortinet.com/lat/resources/cyberglossary/what-is-port-scan

IBM. (s. f.). Network observability. https://www.ibm.com/es-es/think/topics/network-observability

IBM. (s. f.). Network traffic analysis. https://www.ibm.com/mx-es/think/topics/network-traffic-analysis

Juniper Networks. (2023). Monitoreo de estadísticas de tráfico de puerto. https://www.juniper.net/documentation/mx/es/software/network-

director7.1/network-director/topics/task/port-traffic-statistics-monitoring.html

ManageEngine. (s. f.). Escáner de puertos avanzado. https://www.manageengine.com/latam/oputils/escaner-de-puertos-avanzado.html

Network Startup Resource Center. (s. f.). Usar NfSen para identificar los nodos más activos. https://nsrc.org/activities/agendas/sp/nmm-4-days-es/netmgmt/es/netflow/exercise3-nfsen-top-talkers.es.html

Paessler. (s. f.). NetFlow. https://www.paessler.com/es/it-explained/netflow

Prometheus. (s. f.). SNMP exporter. https://github.com/prometheus/snmp_exporter

CONTINUAR