# Módulo 1. Sistemas de detección y prevención de intrusiones (IDS/IPS)

CÓRDOBA CREA FUTURO

# Introducción

La detección y respuesta en redes, conocida como **Network Detection and Response (NDR)**, constituye uno de los pilares técnicos más relevantes dentro de la defensa moderna de infraestructuras digitales. En un entorno caracterizado por amenazas persistentes, movimientos laterales encubiertos, técnicas de evasión y ataques automatizados de alta velocidad, la capacidad de observar, analizar e interpretar el tráfico de red se convierte en una competencia crítica para cualquier profesional de seguridad.

Los **Sistemas de Detección y Prevención de Intrusiones (IDS/IPS)** representan el punto de partida operativo de estas capacidades. De acuerdo con el NIST, un IDPS tiene como finalidad monitorear eventos en sistemas o redes y analizar dichos eventos en busca de indicios de incidentes, generando alertas y, en algunos casos, bloqueando actividad maliciosa (Scarfone & Mell, 2007). Esta definición no solo delimita la función técnica del sistema, sino que introduce un principio fundamental: la seguridad efectiva depende de la visibilidad y del análisis continuo.

En este curso, el foco no estará únicamente en comprender qué es un IDS o un IPS, sino en desarrollar la capacidad de **operarlos, ajustarlos y optimizarlos** dentro de un entorno realista de SOC. La instalación por defecto de un motor IDS no garantiza protección efectiva. Sin personalización, sin ajuste fino de reglas y sin evaluación constante de rendimiento, el sistema puede convertirse en una fuente de ruido, degradación de performance o incluso en un punto ciego operativo.

Las herramientas open source como **Suricata**, desarrollada por la Open Information Security Foundation (OISF), ofrecen un motor de alto rendimiento capaz de operar como IDS, IPS y plataforma de monitoreo de seguridad de red (What is Suricata, s. f.). Su potencia radica no solo en la inspección profunda de paquetes, sino en su capacidad de

trabajar con firmas, protocolos de capa 7 y análisis contextual. Complementariamente, herramientas como **Zeek** permiten generar metadatos estructurados a partir del tráfico de red, facilitando la reconstrucción de sesiones y el análisis posterior (Zeek, s. f.).

La arquitectura donde estos componentes se integran también es determinante. Soluciones como Security Onion permiten desplegar sensores, nodos de gestión y nodos de búsqueda en configuraciones que pueden ir desde entornos de evaluación hasta arquitecturas distribuidas escalables (Architecture, s. f.). Comprender estas arquitecturas no es un aspecto accesorio, sino esencial para garantizar rendimiento, almacenamiento adecuado de logs y correlación efectiva.

Desde una perspectiva profesional, la relevancia de este módulo es directa. Un analista SOC debe ser capaz de:

- Ajustar reglas de detección para reducir falsos positivos.
- Interpretar alertas en contexto.
- Comprender cuándo una alerta no equivale a un incidente.
- Equilibrar profundidad de inspección con impacto en performance.
- Documentar evidencia y justificar decisiones técnicas.

Además, en entornos productivos, la integración con firewalls como pfSense permite aplicar bloqueos selectivos y medidas de contención controladas, evitando respuestas indiscriminadas que afecten procesos legítimos (The pfSense Documentation, 2026).

El propósito de este material es que desarrolles una comprensión integral de los IDS/IPS en el contexto de NDR, combinando fundamentos conceptuales con criterios operativos reales. No se trata únicamente de saber escribir una regla o habilitar una firma, sino de comprender cómo esa decisión impacta en la detección, en la performance y en la respuesta ante incidentes.

En las siguientes secciones, abordarás primero una situación profesional que funcionará como hilo conductor conceptual. A partir de allí, los bloques teóricos desarrollarán los principios técnicos necesarios para que puedas analizar, ajustar y defender una red con criterio profesional.

## Situación profesional

La empresa **NovaRetail S.A.** es una organización de comercio electrónico con presencia regional y un volumen de transacciones que ha crecido de manera sostenida durante los últimos dos años. Su infraestructura tecnológica combina servicios en la nube con servidores on-premise que alojan bases de datos críticas y sistemas internos de gestión.

El equipo de seguridad está compuesto por tres analistas SOC que trabajan en turnos rotativos. Recientemente, la dirección decidió implementar una plataforma de **Network Detection and Response (NDR)** basada en herramientas open source, con el objetivo de aumentar la visibilidad sobre el tráfico interno y detectar posibles movimientos laterales, exfiltraciones de datos o comunicaciones con infraestructura de comando y control.

Se desplegó un sensor de red utilizando Suricata en modo IDS y Zeek para generación de metadatos. La arquitectura inicial responde a un modelo de instalación centralizada similar a un despliegue standalone, donde los logs se almacenan en Elasticsearch y son consultados mediante dashboards analíticos . El firewall perimetral está gestionado con pfSense, lo que permite aplicar bloqueos selectivos ante direcciones IP o dominios maliciosos detectados.

Durante las primeras semanas de operación, comienzan a aparecer cientos de alertas diarias provenientes del conjunto de reglas Emerging Threats Open. Muchas de ellas corresponden a patrones genéricos asociados a tráfico HTTP sospechoso o a posibles intentos de escaneo. El volumen de alertas supera la capacidad de análisis del equipo, generando fatiga y retrasos en la investigación.

Al mismo tiempo, el equipo de infraestructura reporta que el sensor IDS está comenzando a impactar el rendimiento de la red en horarios pico. Se observan pérdidas de paquetes en el puerto espejo del switch y el throughput del sensor se acerca al límite de procesamiento. Los analistas detectan además inconsistencias entre lo que muestra Suricata y los logs de Zeek, lo que dificulta la correlación.

En este contexto, surgen múltiples interrogantes operativos:

- ¿Es suficiente habilitar un ruleset por defecto para garantizar detección efectiva?
- ¿Cómo distinguir entre una alerta informativa y un incidente real?
- ¿Qué criterios deben aplicarse para ajustar reglas sin generar puntos ciegos?
- ¿Hasta qué punto es conveniente activar bloqueo automático?
- ¿Cómo equilibrar profundidad de inspección y performance?

El CISO solicita un plan de acción técnico que incluya:

1. Estrategia de tuning de reglas.

2. Análisis del impacto en performance.
3. Propuesta de contención selectiva mediante firewall.
4. Procedimiento de documentación y trazabilidad.

**Este caso funcionará como eje conceptual del módulo. Cada bloque teórico que desarrollaremos a continuación te permitirá analizar esta situación con criterio técnico y proponer soluciones fundamentadas.**

CONTINUAR

# Unidad 1. Reglas y tuning

## 1.1 Reglas (firmas) de la comunidad Emerging Threats (ET) Open y personalización

Las reglas constituyen la unidad mínima de inteligencia operativa en un IDS basado en firmas. Cada regla es una hipótesis técnica codificada: describe una condición específica del tráfico que, si se cumple, puede representar una amenaza.

La estructura formal de una regla en Suricata incluye acción, encabezado y opciones (Rules Format, s. f.). Sin embargo, comprender su sintaxis no es suficiente. Es necesario entender su lógica interna y su impacto operativo.

La **acción** no solo determina el comportamiento técnico, sino que refleja una decisión estratégica. Una regla con acción alert genera visibilidad; una regla con drop o reject ejecuta una contención inmediata. La diferencia entre ambas puede implicar continuidad operativa o interrupción de servicio.

El **encabezado** delimita el alcance de la regla. Definir correctamente $HOME_NET y $EXTERNAL_NET es esencial para evitar que el sensor analice tráfico irrelevante o, peor aún, ignore tráfico crítico (Zeek, s. f.). En entornos híbridos como NovaRetail, donde coexisten segmentos cloud y on-premise, la definición precisa de estas variables puede requerir múltiples rangos y segmentaciones internas.

Las **opciones** constituyen el núcleo de la detección. Elementos como content, flow, http.method, classtype, sid y rev no son simples campos técnicos; son indicadores que permiten contextualizar la alerta. Una regla que detecta el patrón GET en HTTP no tiene valor por sí misma. Su valor surge cuando ese patrón está asociado a una URI específica, a un flujo establecido hacia el servidor y a una clasificación determinada.

El conjunto de reglas Emerging Threats Open es una base robusta y dinámica, ampliamente utilizada en la comunidad. No obstante, su amplitud es también su principal desafío. Incluye miles de firmas que abarcan múltiples vectores de ataque, tecnologías y protocolos. Activarlas todas sin criterio produce inevitablemente ruido.

La personalización implica un proceso estructurado que puede dividirse en fases:

- Evaluación inicial del entorno.
- Identificación de servicios expuestos.
- Desactivación de categorías irrelevantes.
- Ajuste de prioridades.
- Implementación de reglas propias.

En NovaRetail, por ejemplo, la exposición principal es HTTP/HTTPS. En consecuencia, las categorías relacionadas con explotación web deben tener mayor peso que aquellas asociadas a protocolos no utilizados.

Otro aspecto clave es la creación de reglas basadas en **Indicadores de Compromiso (IoCs)**. Si el equipo de threat intelligence identifica un dominio malicioso específico o una IP asociada a una campaña activa, la capacidad de crear rápidamente una regla personalizada permite adaptar la defensa en tiempo real.

El tuning no es un evento puntual. Es un ciclo continuo. Cada cambio en infraestructura, cada nueva aplicación desplegada y cada variación en comportamiento del usuario puede requerir ajustes adicionales.

En definitiva, la regla no es un elemento estático. Es un componente vivo dentro de un sistema defensivo dinámico.

## 1.2 Firmas versus comportamiento

La detección basada en firmas es determinista: busca coincidencias exactas o patrones definidos. Su fortaleza radica en su precisión frente a amenazas conocidas. Su debilidad es su dependencia del conocimiento previo.

El NIST identifica claramente los tres enfoques principales: firmas, anomalías y análisis de protocolos con estado (Scarfone & Mell, 2007). En la práctica, la combinación de estos métodos aumenta la resiliencia del sistema.

La detección basada en anomalías requiere establecer una línea base de comportamiento normal. En un entorno como NovaRetail, esa línea base podría incluir:

- volumen promedio de consultas DNS por hora.
- Frecuencia típica de conexiones HTTPS salientes.
- Tamaño promedio de transferencias de datos internas.
- Patrones horarios de actividad.

Cuando el tráfico se desvía significativamente de esta línea base, se genera una señal de alerta. Este enfoque permite identificar amenazas que no poseen firma conocida, como variantes nuevas de malware.

El análisis de protocolos con estado agrega otra dimensión. No solo verifica contenido, sino también conformidad estructural. Un paquete HTTP que viola el estándar del protocolo puede ser indicativo de explotación.

Suricata integra capacidades de inspección profunda de capa 7 (What is Suricata, s. f.) , mientras que Zeek genera registros estructurados que describen sesiones, certificados TLS, transacciones DNS y más (Zeek, s. f.). Esta combinación permite pivotar desde una alerta puntual hacia un análisis contextual más amplio.

En NovaRetail, una firma podría detectar un patrón asociado a exfiltración. Pero Zeek podría mostrar que esa transferencia se produjo hacia un dominio recién registrado, con un certificado TLS auto firmado y un patrón periódico de comunicación. La correlación entre ambos elementos transforma una alerta aislada en un indicio fuerte de incidente.

Es crucial comprender que ninguna metodología es autosuficiente. Las firmas ofrecen precisión; el comportamiento ofrece adaptabilidad; el análisis de protocolo ofrece profundidad técnica.

La madurez de un SOC se mide en su capacidad de integrar estas perspectivas y no depender exclusivamente de una sola.

## 1.3 *Performance* y *bypass* consciente

La inspección profunda de tráfico tiene un costo computacional. Cada paquete analizado implica consumo de CPU, memoria y recursos de I/O. Si la arquitectura no está dimensionada adecuadamente, la consecuencia es pérdida de paquetes.

El NIST enfatiza la importancia de considerar el impacto de performance al implementar IDPS (Scarfone & Mell, 2007). Este aspecto no es meramente técnico, sino estratégico.

Un sensor saturado genera una falsa sensación de seguridad. Si el tráfico no es capturado, no puede ser inspeccionado. Zeek reporta tanto pérdida de captura como pérdida de paquetes (Zeek, s. f.), lo que permite monitorear la salud operativa del sistema.

En NovaRetail, la pérdida observada en horarios pico revela que el sensor está operando al límite. Esto obliga a revisar:

- número de workers configurados.
- Distribución NUMA.
- Filtrado previo mediante BPF.
- Optimización de reglas pesadas.
- Segmentación del tráfico inspeccionado.

El concepto de **bypass consciente** implica decidir estratégicamente qué tráfico no necesita inspección profunda. Por ejemplo, tráfico interno entre servidores de backup podría excluirse del análisis intensivo si su riesgo es bajo y su volumen elevado.

No inspeccionar todo no significa reducir seguridad. Significa asignar recursos donde el riesgo es mayor.

Asimismo, activar acciones como drop en modo IPS incrementa la exigencia de procesamiento (Rules Format, s. f.). Cada paquete bloqueado requiere procesamiento adicional. En entornos de alto volumen, esto puede impactar significativamente el throughput.

El equilibrio entre visibilidad y performance es uno de los desafíos centrales de la operación NDR (Performance, s. f.).

## 1.4 Falsos positivos y gestión operativa

Los falsos positivos son inevitables en cualquier sistema de detección. El objetivo no es eliminarlos por completo, sino reducirlos a un nivel manejable.

Un sistema con demasiados falsos positivos pierde credibilidad interna. Los analistas comienzan a ignorar alertas repetitivas. Este fenómeno, conocido como fatiga de alertas, es un riesgo operativo significativo.

Suricata permite aplicar mecanismos como thresholding y supresión para limitar la frecuencia de alertas repetitivas (Suricata User Guide, s. f.). Sin embargo, aplicar estos mecanismos sin análisis previo puede ocultar actividad relevante.

La gestión adecuada de falsos positivos requiere:

- clasificación por criticidad.
- Correlación con otros logs.
- Análisis de impacto en activos críticos.
- Revisión periódica de reglas activas.

En NovaRetail, las alertas genéricas HTTP podrían ser normales en el contexto de una aplicación web intensiva. Desactivarlas completamente podría eliminar ruido, pero también eliminar señales tempranas de explotación.

El enfoque correcto consiste en:

1. analizar muestras representativas.
2. Determinar si la alerta refleja comportamiento esperado.
3. Ajustar la regla o aplicar threshold.
4. Documentar la decisión.

La documentación es clave. Cada regla deshabilitada debe estar justificada. Cada cambio debe registrarse. Sin trazabilidad, la mejora continua es imposible.

**En síntesis, la gestión de falsos positivos no es un ajuste técnico menor. Es una práctica estructural que determina la eficiencia del SOC y la calidad de la detección.**

CONTINUAR

# Unidad 2. Eventos, integración y respuesta en red

La detección en red no finaliza cuando una regla genera una alerta. De hecho, ese es apenas el punto de partida. La verdadera capacidad NDR se consolida cuando los eventos se integran, se correlacionan, se investigan y, eventualmente, se traducen en acciones de contención justificadas y trazables.

En esta unidad abordaremos cuatro dimensiones centrales:

- Integración de eventos hacia Security Onion / SIEM.
- Correlación y trazabilidad del incidente.
- Bloqueos selectivos mediante firewall.
- Diseño de playbooks de contención.

## 2.1 Eventos hacia Security Onion y SIEM

Un IDS aislado tiene un valor limitado. La arquitectura moderna de monitoreo requiere centralización y correlación. En despliegues basados en Security Onion, los sensores generan logs que son enviados a Elasticsearch mediante Elastic Agent y Logstash, donde son parseados e indexados para su análisis posterior (Architecture, s. f.).

La arquitectura distribuida recomendada contempla nodos manager, sensores y nodos de búsqueda, lo cual permite escalar procesamiento y almacenamiento de eventos (Architecture, s. f.). Esta separación no es meramente arquitectónica; tiene implicancias directas en:

- Rendimiento.
- Capacidad de retención de logs.
- Tiempos de consulta.
- Disponibilidad operativa.

En el caso de NovaRetail, el despliegue inicial fue standalone. Esto facilita la implementación rápida, pero limita la escalabilidad. A medida que el volumen de tráfico aumenta, la necesidad de separar funciones se vuelve evidente.

Suricata genera alertas en formato EVE JSON, que incluyen información estructurada sobre:

- Timestamp.
- IP origen y destino.
- Puerto y protocolo.
- Clasificación de la amenaza.

- Identificador único de regla (SID).

Zeek, por su parte, produce logs detallados de conexión, DNS, SSL/TLS, archivos y otros protocolos (Zeek, s. f.). Estos registros no necesariamente implican amenaza, pero aportan contexto invaluable.

La integración con un SIEM permite:

- Correlacionar eventos de red con logs de endpoint.
- Asociar IP con usuario autenticado.
- Detectar secuencias de ataque.
- Medir métricas de detección.

Aquí emerge un principio central de NDR: **la trazabilidad del incidente**. Una alerta aislada no tiene suficiente peso operativo. La correlación contextual transforma eventos en evidencia.

En NovaRetail, una alerta HTTP detectada por Suricata puede correlacionarse con:

- Un proceso ejecutado en un endpoint.
- Una autenticación sospechosa.
- Una transferencia de archivo inusual.
- Una comunicación repetitiva hacia un dominio externo.

La integración no es opcional. Es el paso que convierte detección en investigación (Security Onion Documentation, s. f.).

## 2.2 Correlación y reconstrucción del incidente

El NIST establece que la fase de detección y análisis requiere identificación, coordinación y evaluación antes de cualquier acción de contención (Cybersecurity Incident & Vulnerability Response Playbooks, 2021). Esto implica que no toda alerta debe conducir inmediatamente a un bloqueo.

La reconstrucción de un incidente en red suele requerir:

1. Identificar el flujo original.
2. Determinar el host involucrado.
3. Analizar el comportamiento posterior.
4. Verificar persistencia o repetición.

Zeek facilita este proceso mediante metadatos estructurados. Por ejemplo:

- conn.log permite observar duración, bytes transferidos y estado de la conexión.
- dns.log revela dominios consultados.
- ssl.log permite examinar certificados.

Esta riqueza de información permite detectar patrones como beaconing periódico, característico de comunicación con infraestructura C2.

En NovaRetail, si un host interno consulta cada 60 segundos un dominio recientemente registrado, con bajo ranking reputacional, y simultáneamente Suricata genera una alerta relacionada con exfiltración, el peso probatorio aumenta considerablemente.

La correlación adecuada permite responder a preguntas críticas:

- ¿Es actividad aislada o sostenida?
- ¿Afecta un solo host o múltiples?
- ¿Existe transferencia significativa de datos?
- ¿Hay indicios de movimiento lateral?

Sin esta fase de análisis, la contención puede ser prematura o ineficiente.

## 2.3 Bloqueos selectivos mediante firewall (pfSense)

La contención en red suele materializarse a través del firewall. En entornos gestionados con pfSense, es posible aplicar reglas de bloqueo basadas en IP, dominio o red completa (The pfSense Documentation, 2026).

Sin embargo, el bloqueo indiscriminado puede generar efectos adversos:

- interrupción de servicios legítimos.
- Impacto en clientes externos.
- Bypass sencillo mediante cambio de infraestructura atacante.

El concepto de **bloqueo selectivo** implica:

- validación previa de la amenaza.
- Aplicación temporal del bloqueo.
- Registro detallado de la acción.
- Monitoreo posterior.

pfSense permite crear reglas específicas asociadas a interfaces WAN o LAN, aplicar NAT, monitorear estados de firewall y registrar eventos de bloqueo (The pfSense Documentation, 2026).

En el caso de NovaRetail, ante confirmación de comunicación con una IP maliciosa, el procedimiento podría ser:

1. confirmar actividad mediante correlación Suricata + Zeek.
2. Crear regla temporal de bloqueo en pfSense.
3. Documentar acción en sistema de ticketing.
4. Notificar a equipo de infraestructura.
5. Continuar monitoreo.

Este enfoque evita respuestas impulsivas como “deny all por IP”, que podrían afectar CDN compartidas o servicios cloud legítimos.

Es importante destacar que Suricata también permite operar en modo IPS con acciones drop o reject (Rules Format, s. f.). No obstante, habilitar bloqueo automático sin validación puede derivar en denegaciones de servicio involuntarias.

La decisión entre bloqueo manual y automatizado depende del nivel de madurez del SOC.

## 2.4 *Playbooks* de contención en red

La respuesta eficaz no depende únicamente de herramientas, sino de procedimientos. CISA establece que la respuesta ante incidentes debe seguir fases estructuradas: detección, análisis, contención, erradicación y recuperación (Cybersecurity Incident & Vulnerability Response Playbooks, 2021).

Un **playbook** formaliza estas fases en una secuencia operativa clara. En un contexto NDR, un playbook de contención puede incluir:

- Validación de alerta.
- Correlación con otros logs.
- Determinación de criticidad.
- Decisión de bloqueo.
- Registro y documentación.
- Revisión posterior.

El valor del playbook radica en la estandarización. Reduce la improvisación y mejora la consistencia entre analistas.

En NovaRetail, la ausencia inicial de playbooks generó incertidumbre. Algunos analistas proponían bloqueos inmediatos; otros preferían esperar confirmación adicional. La falta de criterio unificado ralentizaba la respuesta.

Un playbook bien diseñado debe equilibrar:

- Rapidez.
- Precisión.
- Minimización de impacto operativo.
- Conservación de evidencia.

Asimismo, debe incluir métricas de desempeño:

- Tiempo medio de detección (MTTD).
- Tiempo medio de respuesta (MTTR).
- Tasa de falsos positivos.
- Porcentaje de incidentes correctamente contenidos.

La automatización mediante herramientas SOAR puede acelerar estos procesos, pero la sobre-automatización representa un riesgo. Un bloqueo automático sin validación puede interrumpir procesos críticos.

La madurez operativa implica saber cuándo automatizar y cuándo requerir validación humana.

CONTINUAR

# Cierre

La implementación de capacidades de **Detección y Respuesta en Redes (NDR)** no consiste únicamente en desplegar un sensor IDS o activar un conjunto de reglas. Implica construir una arquitectura de visibilidad, análisis y contención alineada con el riesgo real del entorno.

A lo largo de este módulo analizaste cómo las reglas constituyen el núcleo operativo de los sistemas IDS/IPS, pero también comprendiste que su eficacia depende del **tuning continuo**. Un ruleset por defecto no es sinónimo de protección. La personalización, la priorización y la revisión periódica son condiciones indispensables para transformar un motor de firmas en una herramienta realmente defensiva.

Asimismo, profundizaste en la diferencia entre **firma, anomalía y análisis de protocolo**, comprendiendo que la detección moderna es necesariamente híbrida. La combinación de Suricata como motor de inspección y Zeek como generador de metadatos estructurados permite superar la lógica binaria de “alerta sí / alerta no” y avanzar hacia una interpretación contextual.

La noción de **performance y bypass consciente** introduce una dimensión estratégica. Defender no significa inspeccionar todo sin criterio, sino inspeccionar lo que aporta valor defensivo. Un sensor saturado, con pérdida de paquetes, genera una ilusión de seguridad. La visibilidad real exige equilibrio entre profundidad de inspección y capacidad de procesamiento.

La integración de eventos hacia plataformas como Security Onion y SIEM permite vincular tráfico, procesos, usuarios y activos. Esta trazabilidad es la base de cualquier decisión de contención.

En este punto se vuelve evidente que la respuesta técnica debe estar respaldada por procedimientos formales. Los **playbooks de contención** estructuran la acción del SOC, reducen improvisación y permiten medir desempeño mediante indicadores como MTTD y MTTR.

El caso de NovaRetail S.A. ilustra una situación frecuente en organizaciones en crecimiento: despliegue rápido de herramientas sin madurez operativa suficiente. El exceso de alertas, la pérdida de paquetes y la falta de criterios de bloqueo revelan que la tecnología, por sí sola, no garantiza defensa efectiva.

La madurez en NDR se alcanza cuando:

- Las reglas reflejan el contexto real del negocio.
- Las alertas se correlacionan antes de actuar.
- Los bloqueos son selectivos y temporales.
- Cada acción queda documentada.
- La arquitectura escala junto con el tráfico.
- Las decisiones técnicas están justificadas por métricas.

En definitiva, la NDR no es una herramienta, sino una capacidad organizacional.

Como futuro profesional en un SOC, tu desafío no será simplemente activar detecciones, sino **operarlas con criterio**, comprender sus limitaciones y defender cada decisión técnica frente a impacto operativo y riesgo de negocio.

Este módulo establece los fundamentos de esa competencia.

CONTINUAR

# Referencias

**Architecture Documentation**. (s. f.). Security Onion Solutions. https://docs.securityonion.net/en/2.4/architecture.html

**Cybersecurity and Infrastructure Security Agency** (CISA). (2021). *Cybersecurity Incident & Vulnerability Response Playbooks*. https://www.cisa.gov/sites/default/files/2024-08/Federal_Government_Cybersecurity_Incident_and_Vulnerability_Response_Playbooks_508C.pdf

**Electric Sheep Fencing LLC & Rubicon Communications LLC**. (2026). The pfSense Documentation. Netgate.

**Performance** (s. f.). https://docs.suricata.io/en/latest/performance/

**Rules Format – Suricata Documentation** (s.f.). Open Information Security Foundation. https://docs.suricata.io/en/latest/rules/intro.html

**Security Onion Documentation** (s. f.). Security Onion Solutions. https://docs.securityonion.net/en/2.4/

**Scarfone, K., & Mell, P.** (2007). *Guide to Intrusion Detection and Prevention Systems (IDPS)* (NIST Special Publication 800-94). National Institute of Standards and Technology.

**Suricata User Guide** (s.f.). Open Information Security Foundation. https://docs.suricata.io/en/suricata-8.0.2/

**Zeek Integration Documentation**. (s.f.). Security Onion Solutions. https://docs.securityonion.net/en/2.4/zeek.html

CONTINUAR