Practica Blue Team

Infraestructura Blue Team

Centralización de logs desde LAN, DMZ y DMZ2

Alumno: Dani García
Bootcamp: Ciberseguridad — KeepCoding
Módulo: Blue Team
Fecha: 31/01/2026

Componentes principales del laboratorio:

- pfSense (segmentación y control de tráfico)
- Red LAN (Windows)
- Red DMZ (Honeypot)
- Red DMZ2 (Suricata)
- Elastic SIEM (Elastic Cloud)

Contexto del ejercicio

El objetivo de esta práctica es diseñar e implementar una **infraestructura de red segmentada** orientada a **monitorización y detección**, siguiendo un enfoque Blue Team.

La arquitectura debe incluir:

- Segmentación de red mediante **pfSense**
- Separación lógica en **LAN, DMZ y DMZ2**
- Fuentes de logs heterogéneas en cada red
- Centralización y visualización de logs en **Elastic SIEM**

Esta práctica permite comparar el comportamiento de distintas fuentes de logs y sistemas operativos dentro de un entorno controlado.

Resumen de la infraestructura implementada

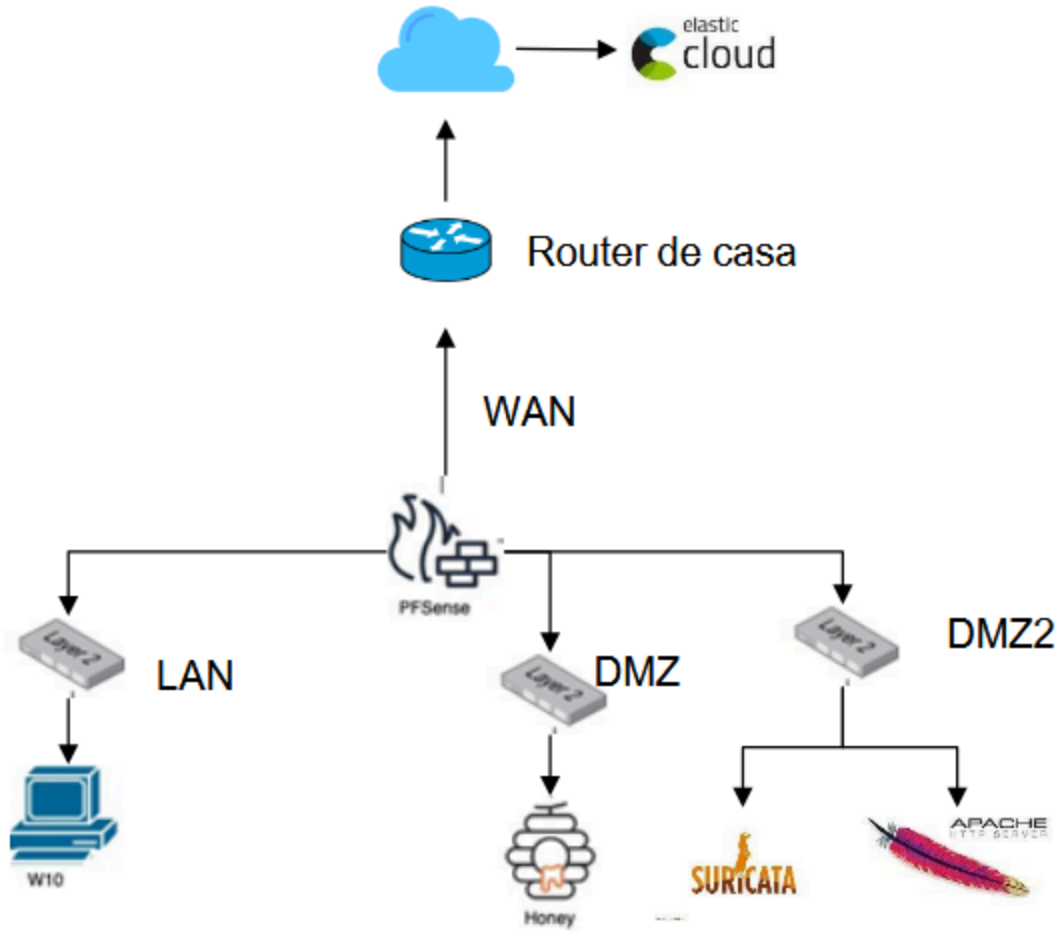
La infraestructura final implementada consta de los siguientes elementos:

- **pfSense** como firewall y router central, encargado de interconectar y segmentar las redes **LAN**, **DMZ** y **DMZ2**, aplicando las correspondientes reglas de tráfico entre ellas.
- **Red LAN**:
  - Equipo **Windows 10** actuando como sistema interno.
  - Envío de logs y métricas al SIEM mediante **Elastic Agent**, permitiendo la monitorización del sistema desde la red interna.
- **Red DMZ**:
  - **Honeypot** accesible desde la red **WAN**, expuesto de forma controlada al exterior.
  - Sin acceso a redes internas (**LAN** y **DMZ2**), cumpliendo el aislamiento requerido.
  - Envío de logs personalizados a Elastic mediante **Elastic Agent**, ejecutándose sobre una máquina **Kali Linux**.
- **Red DMZ2**:
  - Sensor **Suricata** como fuente adicional e independiente de logs.
  - Generación de alertas mediante reglas personalizadas ante tráfico potencialmente inseguro.
  - Uso de una máquina **Kali Linux** con **Elastic Agent** para el envío de eventos al SIEM.
- **Elastic SIEM (Elastic Cloud)**:
  - Recepción, almacenamiento y visualización centralizada de logs y métricas procedentes de las tres redes.

Arquitectura de red y segmentación

La infraestructura se ha diseñado siguiendo un modelo de **segmentación por zonas**, con el objetivo de aislar los distintos activos según su nivel de exposición y función dentro del sistema.

La interconexión entre redes se realiza mediante **pfSense**, que actúa como firewall y router central, controlando todo el tráfico entre segmentos.



Segmentos de red definidos

La arquitectura se divide en los siguientes segmentos:

- **WAN**
  - Red externa (máquina host / router doméstico).
  - Punto de entrada desde el exterior.
- **LAN**
  - Red interna de confianza.
  - Alberga el sistema Windows utilizado como equipo interno.
  - Acceso restringido desde redes externas.

- **DMZ**
  - Zona desmilitarizada.
  - Alberga el honeypot.
  - Expuesta al exterior (WAN) de forma controlada.
  - Sin acceso a redes internas.
- **DMZ2**
  - Red aislada para sensores.
  - Alberga el sistema Suricata como fuente adicional de logs.
  - Sin exposición directa al exterior.

Esquema lógico de la red

A continuación se muestra el esquema lógico de la infraestructura implementada:

En dicho esquema se aprecia:

- pfSense como punto central de interconexión.
- Separación clara entre redes internas, expuestas y de monitorización.
- Flujo de logs desde todos los segmentos hacia Elastic SIEM.

Direccionamiento IP por segmento

Cada red cuenta con su propio rango de direcciones IP, permitiendo identificar de forma clara el origen de los logs:

Red	Interfaz pfSense	Propósito	Rango IP
LAN	em1	Red interna	192.168.100.1/24
DMZ	em2	Honeypot	192.168.200.1/24
DMZ2	em3	Sensor Suricata	192.168.250.1/24

La interfaz WAN conecta pfSense con la red externa (máquina host / router doméstico) y actúa como frontera de seguridad. Esta interfaz no se considera un segmento interno del laboratorio, pero es clave para la exposición controlada de la DMZ.

Este direccionamiento será clave para validar posteriormente que los logs recibidos en Elastic provienen del segmento de red correcto.

Configuración de firewall y reglas de tráfico (pfSense)

Aquí el objetivo **no es listar reglas sin más**, sino demostrar **criterio de seguridad** y que cada red **cumple su función** según el enunciado.

Enfoque general de seguridad

La política de filtrado aplicada en pfSense sigue el principio de **deny by default**, permitiendo únicamente el tráfico estrictamente necesario entre redes.

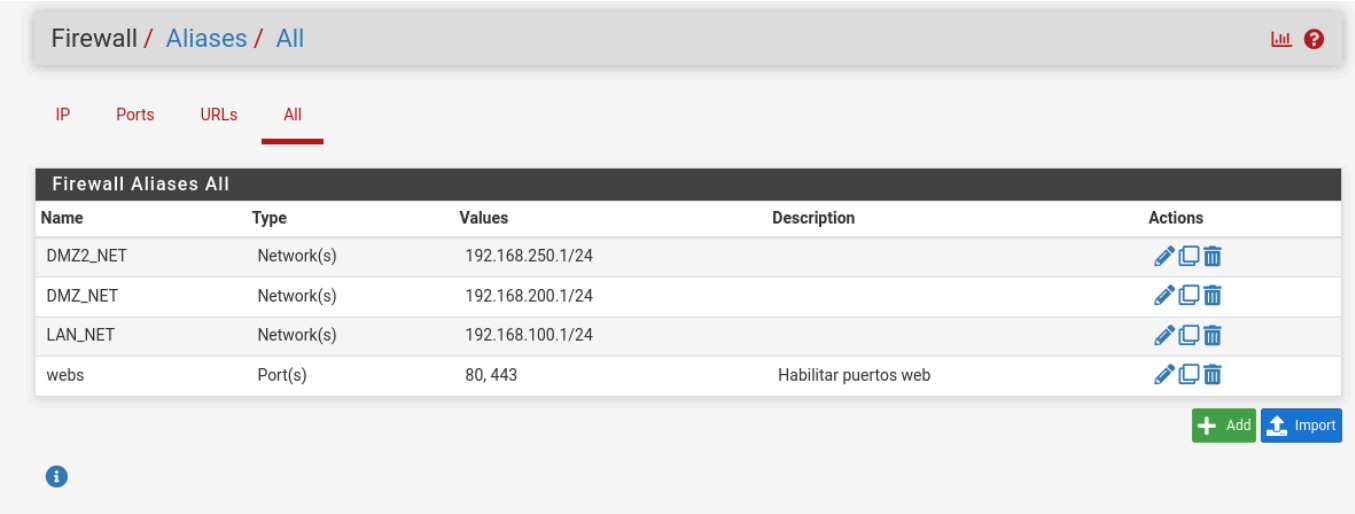
Cada interfaz (WAN, LAN, DMZ y DMZ2) dispone de reglas específicas acordes a su rol dentro de la arquitectura.

Aliases (pfSense) para simplificar y asegurar reglas

Para mantener las reglas de firewall más claras, mantenibles y menos propensas a error, se configuraron **Aliases** en pfSense (hosts, redes y puertos).
Estos aliases se reutilizan en reglas de WAN/LAN/DMZ/DMZ2 y en NAT/Port Forwarding, evitando duplicar valores y facilitando futuros cambios.

Tipos de aliases utilizados:

- **Hosts/Networks**: redes LAN/DMZ/DMZ2 y hosts específicos (Windows, Honeypot, Suricata).
- **Ports**: puertos/servicios expuestos hacia DMZ (honeypot) y puertos necesarios para salida a Elastic.



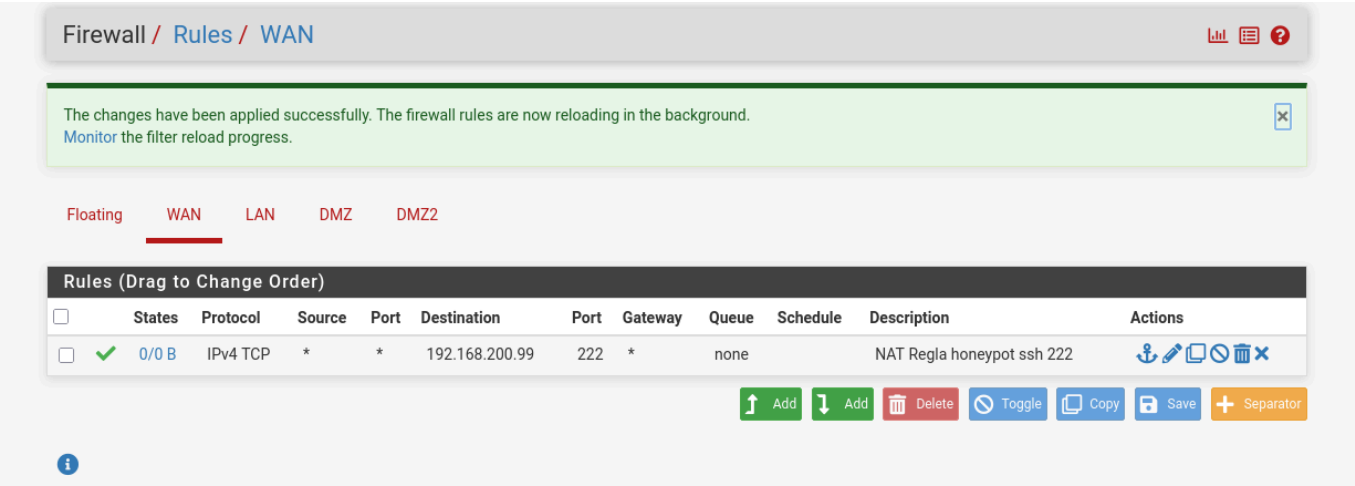
El uso de aliases mejora la legibilidad del firewall y reduce el riesgo de inconsistencias al aplicar cambios.

Reglas por interfaz

WAN (red externa)

Objetivo: permitir acceso controlado únicamente a la DMZ.

- Permitir tráfico entrante desde WAN hacia el **honeypot en DMZ** (puertos/servicios expuestos).
- Bloquear cualquier acceso directo desde WAN hacia **LAN** y **DMZ2**.

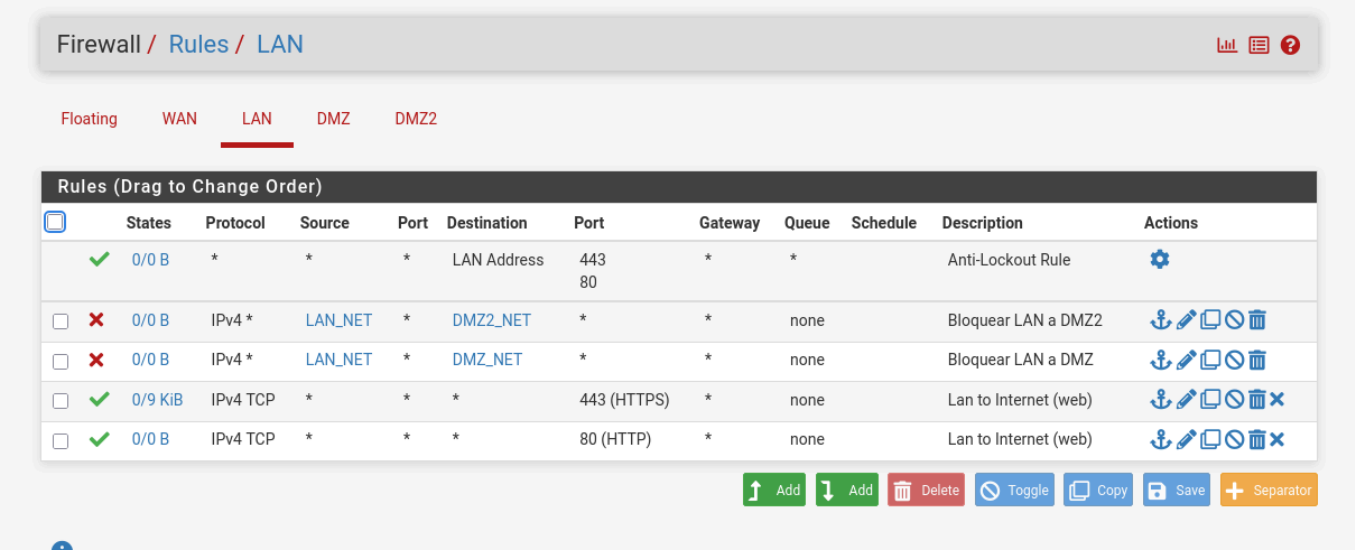


LAN (red interna)

Objetivo: proteger el sistema interno y permitir salida controlada.

- Permitir tráfico desde **LAN hacia WAN**.
- Permitir tráfico desde **LAN hacia Elastic Cloud** (envío de logs).
- Bloquear tráfico entrante desde **DMZ y DMZ2** hacia LAN.

- No exponer servicios de LAN hacia WAN.

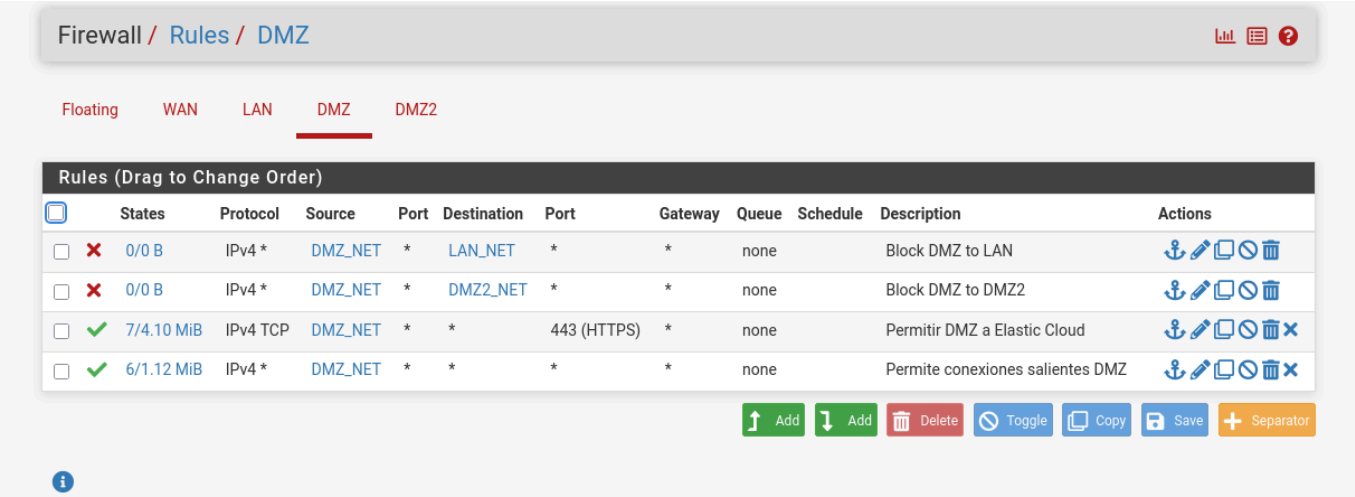


DMZ (zona desmilitarizada)

Objetivo: exposición controlada y aislamiento interno.

- Permitir tráfico desde **WAN hacia DMZ** (honeypot).
- Permitir salida desde **DMZ hacia Elastic Cloud** (envío de logs).
- Bloquear tráfico desde DMZ hacia **LAN**.
- Bloquear tráfico desde DMZ hacia **DMZ2**.

Esta configuración garantiza que un posible compromiso del honeypot no afecte a redes internas. Se minimiza la posibilidad de entrada por Honeypot y escalado de privilegios o acceso a data sensible.

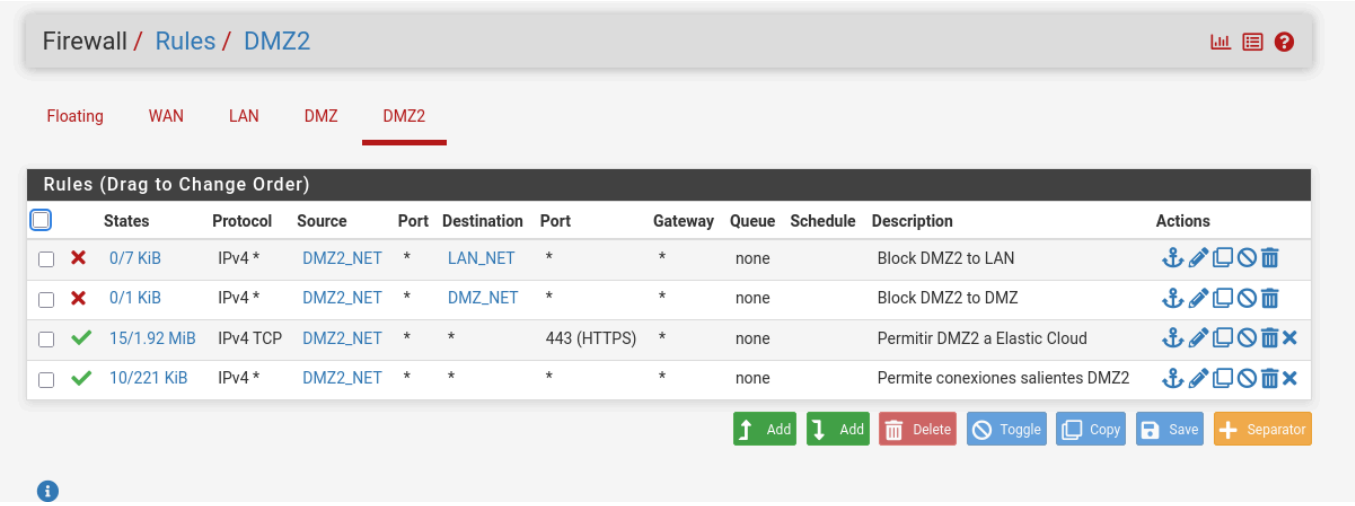


DMZ2

Objetivo: monitorización sin exposición directa.

- Permitir salida desde **DMZ2 hacia Elastic Cloud**.
- Bloquear tráfico entrante desde WAN.
- Bloquear tráfico desde **LAN y DMZ** hacia DMZ2.
- No exponer servicios de DMZ2 al exterior.

Esta red queda completamente aislada, actuando únicamente como **fuente de detección**.

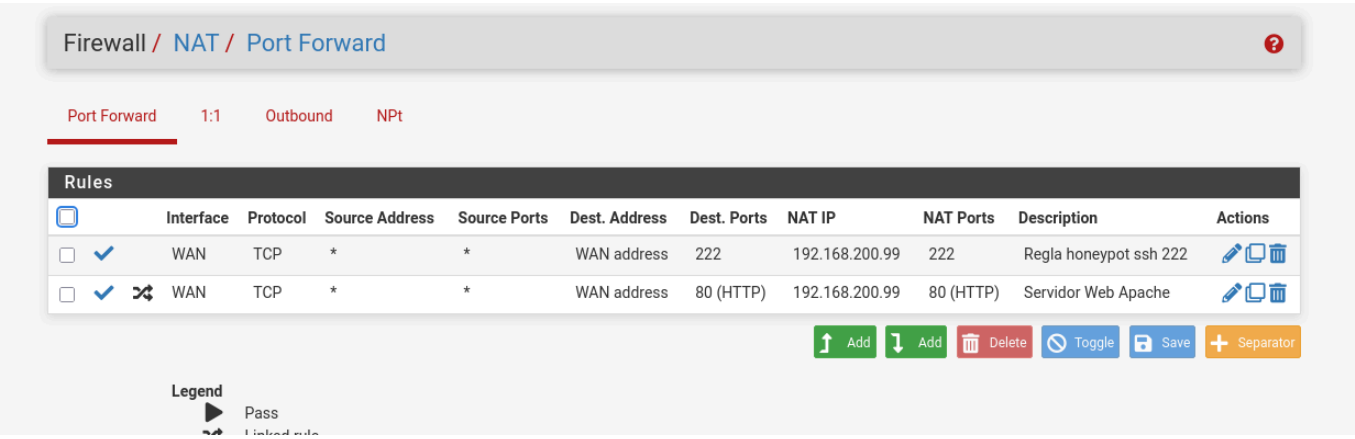


NAT

Se configuró **Port Forwarding** en pfSense para permitir el acceso desde **WAN** hacia el honeypot ubicado en **DMZ**, de forma controlada y limitada a los servicios necesarios.

Criterios aplicados:

- Exposición únicamente de servicios del honeypot (DMZ).
- No se expone LAN ni DMZ2.
- Reglas asociadas de firewall en WAN vinculadas al Port Forward.
- Registro y trazabilidad del tráfico entrante según las reglas definidas.



Con el objetivo de facilitar la comprensión del diseño de seguridad implementado, a continuación se muestra una tabla resumen con las **reglas de firewall más relevantes**,

organizadas por interfaz.

Esta tabla no pretende sustituir la configuración completa, sino destacar las reglas que definen el comportamiento principal de cada segmento de red.

Interfaz	Origen	Destino	Puertos / Servicio	Acción	Descripción
WAN	Any	DMZ_NET	webs	Allow	Acceso web desde WAN al honeypot en DMZ mediante NAT
WAN	Any	Any	Any	Deny	Bloqueo por defecto del tráfico entrante
LAN	LAN_NET	Elastic Cloud	443	Allow	Envío de logs y métricas desde Windows al SIEM
LAN	LAN_NET	DMZ_NET	Any	Deny	Bloqueo de acceso desde LAN a la DMZ
LAN	LAN_NET	DMZ2_NET	Any	Deny	Bloqueo de acceso desde LAN a la DMZ2
DMZ	DMZ_NET	Any	Any	Deny	Aislamiento del honeypot respecto a otras redes
DMZ2	DMZ2_NET	Elastic Cloud	443	Allow	Envío de alertas de Suricata al SIEM
DMZ2	DMZ2_NET	Any	Any	Deny	Aislamiento completo del sensor

Este conjunto de reglas implementa un modelo de **deny by default**, permitiendo únicamente el tráfico estrictamente necesario para el funcionamiento del laboratorio.

Validación del aislamiento entre redes

Una vez configuradas las reglas de firewall y NAT en pfSense, se realizaron pruebas de conectividad para verificar el correcto **aislamiento entre los distintos segmentos de red**.

El objetivo de estas pruebas fué para comprobar que:

- cada red solo puede comunicarse con los destinos explícitamente permitidos,
- no existe conectividad lateral entre redes internas no autorizadas.

Pruebas de conectividad realizadas

Se realizaron pruebas de conectividad mediante **ping** y accesos directos entre los distintos segmentos:

LAN → DMZ/DMZ2

- Acceso no permitido.
- Resultado: tráfico bloqueado según las reglas definidas.

```
C:\Users\sergio>ping 192.168.200.99

Haciendo ping a 192.168.200.99 con 32 bytes de datos:
Tiempo de espera agotado para esta solicitud.
Tiempo de espera agotado para esta solicitud.
Tiempo de espera agotado para esta solicitud.

Estadísticas de ping para 192.168.200.99:
    Paquetes: enviados = 3, recibidos = 0, perdidos = 3
    (100% perdidos),
Control-C
^C
C:\Users\sergio>ping 192.168.250.X
La solicitud de ping no pudo encontrar el host 192.168.250.X. Compruebe el nombre y
vuelva a intentarlo.
```

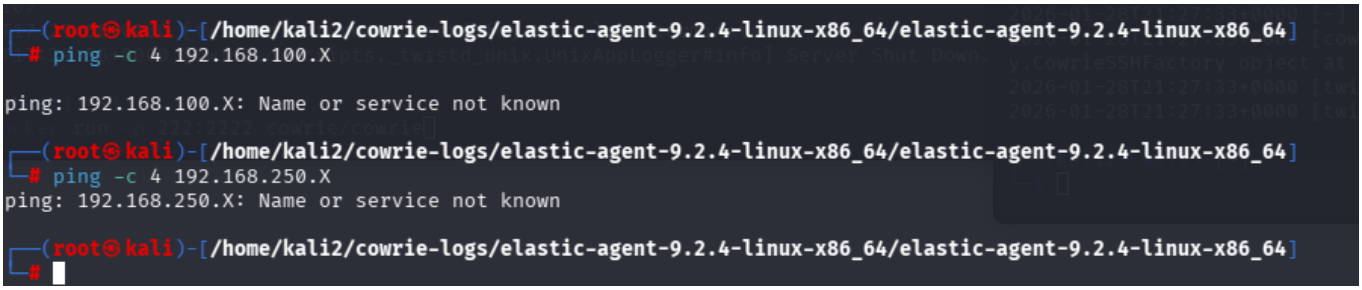
DMZ → LAN/DMZ2

- Acceso no permitido.
- Resultado: aislamiento efectivo de la red interna.

```
┌──(dani㉿BootCamp-Kali)-[~]
└─$ ping -c 4 192.168.200.99

PING 192.168.200.99 (192.168.200.99) 56(84) bytes of data.
── 192.168.200.99 ping statistics ──
4 packets transmitted, 0 received, 100% packet loss, time 3067ms


┌──(dani㉿BootCamp-Kali)-[~]
└─$ ping -c 4 192.168.100.X

ping: 192.168.100.X: Name or service not known

┌──(dani㉿BootCamp-Kali)-[~]
└─$
```

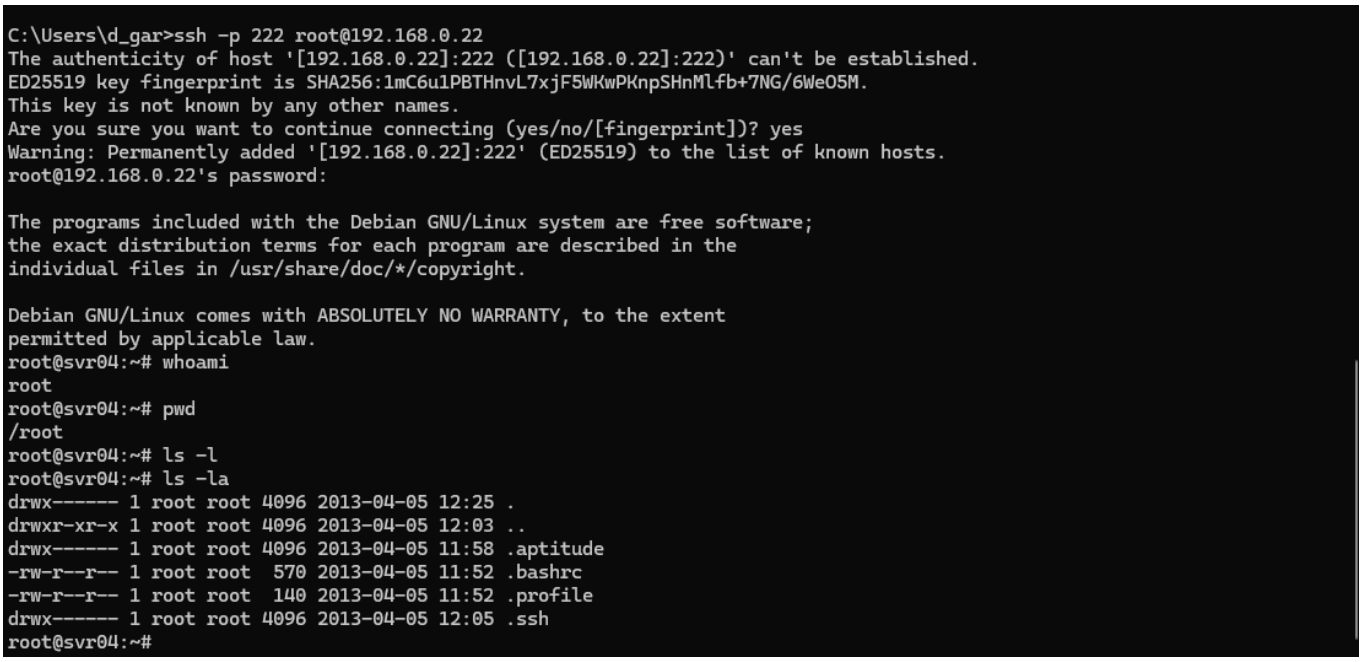
DMZ2 → LAN/DMZ

- Acceso no permitido.
- Resultado: aislamiento efectivo de la red interna.



Acceso permitido (control positivo)

- Acceso desde cada red hacia **Elastic Cloud** permitido.
- Acceso desde WAN hacia el honeypot en DMZ permitido.



Elastic SIEM: arquitectura, Fleet y políticas de agentes

Aquí vamos a ver la **arquitectura del SIEM**, la gestión centralizada de agentes y la asignación de **políticas diferenciadas** según el segmento de red, sentando la base para la posterior recepción y análisis de logs.

Arquitectura del SIEM

Se utiliza **Elastic SIEM en Elastic Cloud** como plataforma centralizada para la recepción, almacenamiento y visualización de logs y métricas procedentes de los distintos segmentos de red del laboratorio.

La comunicación entre los sistemas monitorizados y Elastic Cloud se realiza mediante **Elastic Agent**, gestionado de forma centralizada a través de **Fleet**.

Este enfoque permite:

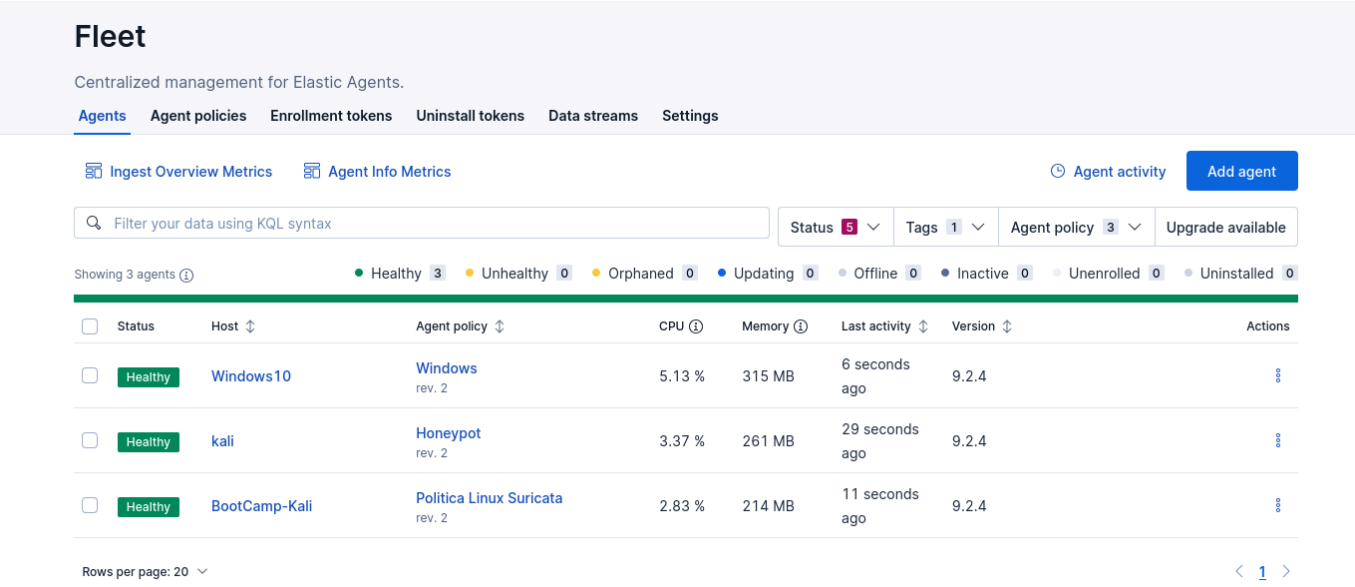
- gestión unificada de agentes,
- asignación de políticas específicas por tipo de sistema,
- control y visibilidad sobre el estado de cada agente.

Gestión de agentes con Fleet

Fleet actúa como punto central para:

- el enrolamiento de agentes,
- la asignación de políticas,
- la supervisión del estado de los agentes desplegados.

Cada sistema del laboratorio dispone de su **Elastic Agent propio**, correctamente enrolado y en estado **Healthy**, lo que garantiza la correcta comunicación con el SIEM.



Relación entre políticas y segmentación de red

La separación de políticas permite:

- identificar claramente el origen de los logs,
- aplicar integraciones específicas según el rol del sistema,
- correlacionar eventos teniendo en cuenta el **segmento de red** desde el que se generan.

Esta segmentación lógica dentro del SIEM es coherente con la segmentación física implementada mediante pfSense.

Fuente de logs en DMZ2: Suricata

Una vez el sistema se terminó de configurar y probar, se empezó a recibir la info de la **fuente de logs ubicada en la red DMZ2**, cumpliendo el requisito de disponer de una **tercera fuente de logs distinta** a Windows y al honeypot.

Rol de la red DMZ2

La red **DMZ2** se ha diseñado como una red **aislada**, destinada en este caso exclusivamente a **sensores de detección**.
No se encuentra expuesta a la red WAN ni accesible desde otras redes internas.

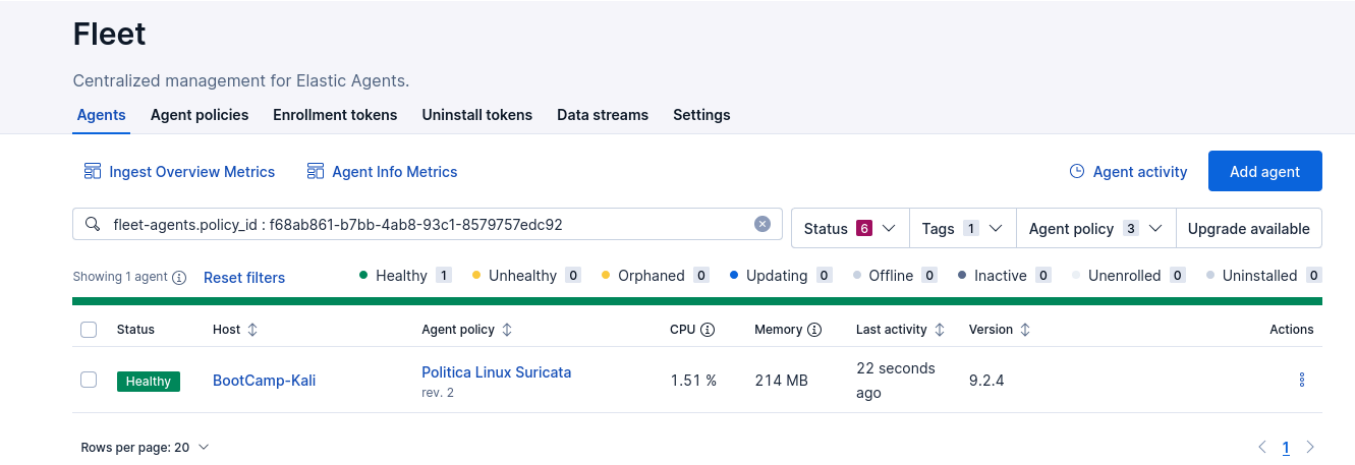
En esta red se ha desplegado **Suricata**, actuando como sistema de detección de intrusiones (IDS) y como fuente adicional de eventos de seguridad.

Despliegue de Suricata

Suricata se ejecuta sobre una máquina **Linux (Kali)** ubicada en la red DMZ2, con las siguientes características:

- Sensor dedicado a la inspección de tráfico.
- Generación de eventos ante patrones sospechosos.
- Envío de logs al SIEM mediante **Elastic Agent**.

El agente se encuentra correctamente enrolado en Fleet y asociado a la **política DMZ2**, específica para este tipo de fuente.



Generación de eventos y alertas

Para validar el correcto funcionamiento del sensor, se configuraron **reglas personalizadas en Suricata**, diseñadas para generar alertas ante tráfico considerado potencialmente inseguro.

Estas reglas permiten comprobar:

- que Suricata detecta eventos de interés,
- que los eventos se generan en la red correcta (DMZ2),
- que los logs llegan correctamente a Elastic.

```
┌──(dani㉿BootCamp-Kali)-[/etc/suricata/rules]
└─$ cat suricata.rules
#alert tcp any any -> any any (msg:"trafico detectado"; sid:1;)
alert tcp any any -> 192.168.1.65 22 (msg:"Trafico SSH  detectado"; sid:2; classtype:attempted-admin;)
alert http any any -> any any (msg:"Archivo PDF Detectado"; flow:established,to_client; file_data; content:"%PDF-"; within:5; filestore ;
 sid:3; classtype:file-download;)
```

Evidencias de recepción de logs en Elastic

Se procede a inicializar el Suricata, para hacer unos test internos para verificar que los logs se crean correctamente. Dadas las reglas aplicadas mostradas anteriormente, se realiza un test descargando un PDF desde una dirección HTTP no segura (no HTTPS), haciendo que se genere el log de alerta.

```
┌──(dani㉿BootCamp-Kali)-[~]
└─$ sudo suricata -c /etc/suricata/suricata.yaml -i eth0
[sudo] password for dani:
i: suricata: This is Suricata version 8.0.3 RELEASE running in SYSTEM mode
i: mpm-hs: Rule group caching - loaded: 1 newly cached: 0 total cacheable: 1
i: threads: Threads created -> W: 6 FM: 1 FR: 1   Engine started.
```

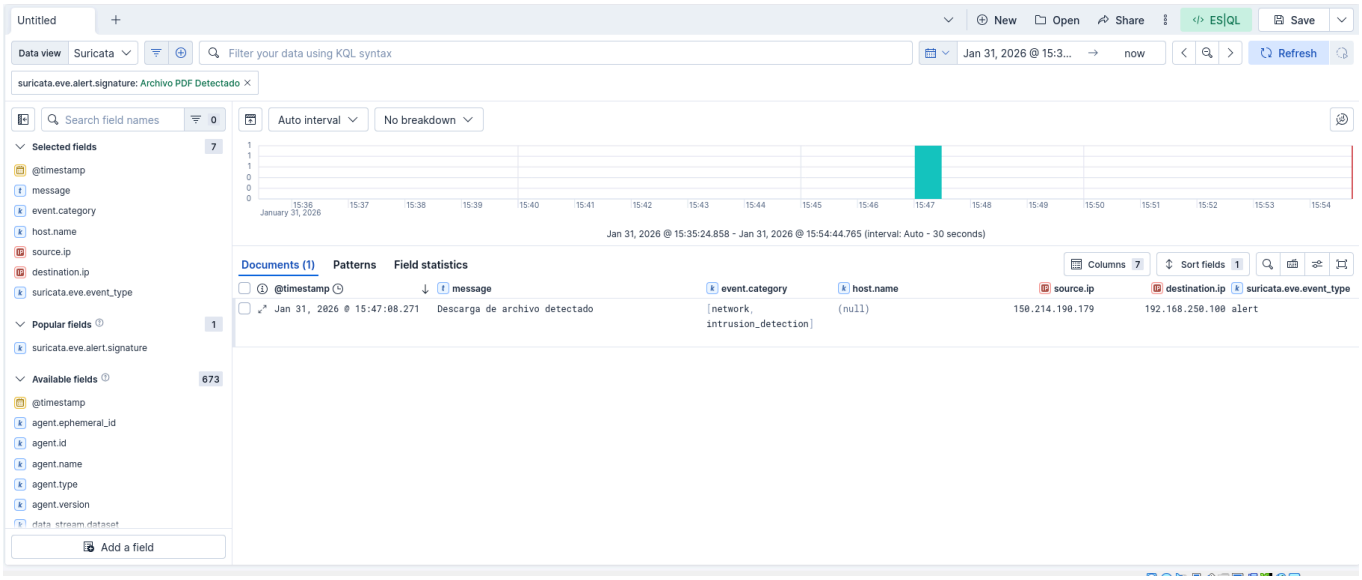
```
┌──(dani㉿BootCamp-Kali)-[/var/log/suricata]
└─$ sudo tail -f fast.log
01/26/2026-22:22:57.646317  [**] [1:1:0] trafico detectado [**] [Classification: (null)] [Priority: 3] {TCP} 34.36.54.80:443 -> 192.168.0.24:36582
01/26/2026-22:22:57.645975  [**] [1:1:0] trafico detectado [**] [Classification: (null)] [Priority: 3] {TCP} 34.36.54.80:443 -> 192.168.0.24:36566
01/26/2026-22:22:57.643059  [**] [1:1:0] trafico detectado [**] [Classification: (null)] [Priority: 3] {TCP} 192.168.0.24:36594 -> 34.36.54.80:443
01/26/2026-22:22:57.647316  [**] [1:1:0] trafico detectado [**] [Classification: (null)] [Priority: 3] {TCP} 34.36.54.80:443 -> 192.168.0.24:36594
01/26/2026-22:22:57.647534  [**] [1:1:0] trafico detectado [**] [Classification: (null)] [Priority: 3] {TCP} 34.36.54.80:443 -> 192.168.0.24:36596
01/26/2026-22:22:57.649232  [**] [1:1:0] trafico detectado [**] [Classification: (null)] [Priority: 3] {TCP} 34.36.54.80:443 -> 192.168.0.24:36612
01/26/2026-22:22:57.649010  [**] [1:1:0] trafico detectado [**] [Classification: (null)] [Priority: 3] {TCP} 34.36.54.80:443 -> 192.168.0.24:36624
01/26/2026-22:22:57.694531  [**] [1:1:0] trafico detectado [**] [Classification: (null)] [Priority: 3] {TCP} 192.168.0.24:40832 -> 34.107.221.82:80
01/26/2026-22:22:57.703651  [**] [1:1:0] trafico detectado [**] [Classification: (null)] [Priority: 3] {TCP} 34.107.221.82:80 -> 192.168.0.24:40832
01/28/2026-19:28:34.092398  [**] [1:3:0] Archivo PDF Detectado [**] [Classification: Descarga de archivo detectado] [Priority: 2] {TCP} 163.117.139.
115:80 -> 192.168.0.24:40554
```

Una vez se probó esto, pasamos a comprobar que se recibian los mismos logs en el sistema **Elastic**.

Se verificó la recepción de los logs generados por Suricata en Elastic SIEM mediante consultas en **Discover**, observándose documentos con las siguientes características:

- IP de origen perteneciente a la red **DMZ2**.
- Dataset correspondiente a **Suricata**.
- Marca temporal coherente con el momento de generación del evento.


Explore in Discover
Inspect
Settings
Save
Data view
Suricata
Filter your data using KQL syntax
Jan 31, 2026 @ 15:0...
now
Refresh
suricata.eve.alert.signature: Archivo PDF Detectado
Search field names
agent.name
agent.type
agent.version
data_stream.dataset
data_stream.namespace
data_stream.type
destination.address
destination.as.number
destination.as.organization.name
destination.bytes
destination.domain
destination.geo.city_name
destination.geo.continent_name
destination.geo.country_iso_code
Count of records
Archivo PDF Detectado
Top 9 values of suricata.eve.alert.signature
Bar
Stacked
Suricata
Horizontal axis
Optional
Top 9 values of suricata.eve.alert.signature
Vertical axis
Count of records
Add or drag-and-drop a field
Breakdown
Optional
Add or drag-and-drop a field
Suggestions
Current visualization
Heat map
Treemap
Count of records
Pie
Waffle


Untitled
New
Open
Share
ES|QL
Save
Data view
Suricata
Filter your data using KQL syntax
Jan 31, 2026 @ 15:3...
now
Refresh
suricata.eve.alert.signature: Archivo PDF Detectado
Search field names
Auto interval
No breakdown
Selected fields
@timestamp
message
event.category
host.name
source.ip
destination.ip
suricata.eve.event_type
Popular fields
suricata.eve.alert.signature
Available fields
673
@timestamp
agent.ephemeral_id
agent.id
agent.name
agent.type
agent.version
Add a field
January 31, 2026
Jan 31, 2026 @ 15:35:24.858 - Jan 31, 2026 @ 15:54:44.765 (interval: Auto - 30 seconds)
Documents (1)
Patterns
Field statistics
Columns
Sort fields
@timestamp
message
event.category
host.name
source.ip
destination.ip
suricata.eve.event_type
Jan 31, 2026 @ 15:47:08.271
Descarga de archivo detectado
network, intrusion_detection
(null)
150.214.190.179
192.168.250.100
alert

```
    "namespace": "default",
    "type": "logs"
  },
  "destination": {
    "address": "192.168.250.100",
    "bytes": 46806,
    "domain": "lsi2.ugr.es",
    "ip": "192.168.250.100",
    "packets": 33,
    "port": 39742
  },
  "ecs": {
    "version": "8.17.0"
  },
  "elastic_agent": {
    "id": "f8a1090c-222a-4827-b2cb-fc75503d0771",
    "snapshot": false,
    "version": "9.2.4"
  },
  "event": {
    "agent_id_status": "verified",
    "category": [
      "network",
      "intrusion_detection"
    ],
    "created": "2026-01-31T14:47:08.410Z",
```


Ejemplo de log recibido (JSON)

A continuación se muestra un ejemplo de documento JSON correspondiente a un evento generado por Suricata:

`{"address": "192.168.250.100","bytes": 46806,"domain": "lsi2.ugr.es","ip": "192.168.250.100","packets": 33,"port": 39742}`

Este documento permite identificar claramente:

- el origen del evento,
- el tipo de detección realizada,
- el segmento de red desde el que se genera.

(El JSON completo se incluye en el anexo.)

Fuente de logs en DMZ: Honeypot

En este paso realizamos el mismo proceso de verificación y funcionamiento para la **fuente de logs ubicada en la red DMZ**, cuyo objetivo es **exponer de forma controlada servicios al exterior** y capturar interacciones potencialmente maliciosas.

Rol de la red DMZ

La DMZ se ha diseñado como una zona expuesta al exterior (WAN), pero aislada de las redes internas (LAN y DMZ2).
Su función principal es permitir la interacción desde el exterior sin comprometer activos internos.

En esta red se ha desplegado un honeypot, cuya finalidad es:

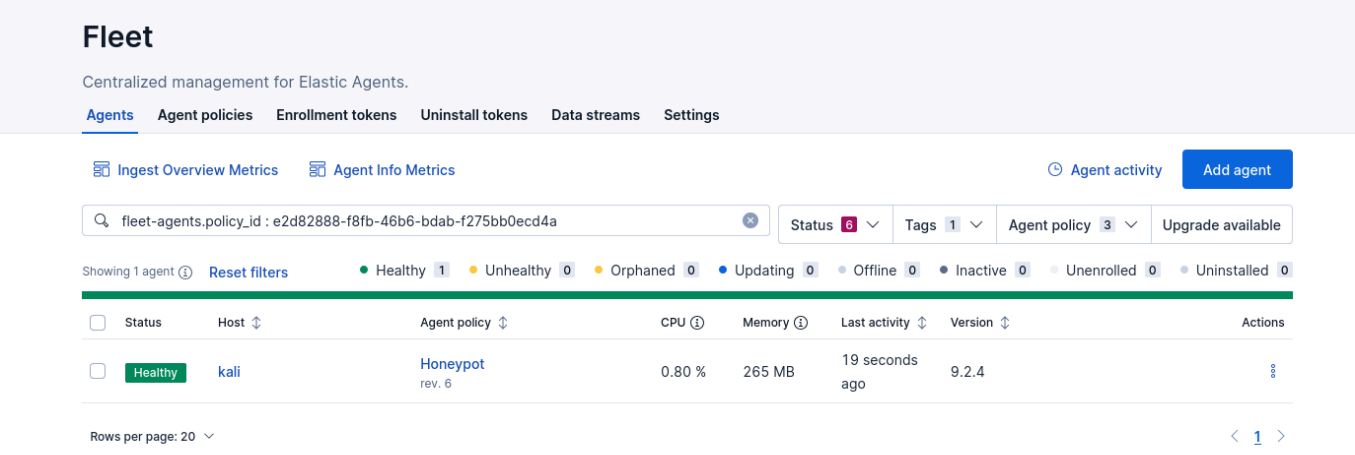
- atraer conexiones externas,
- registrar intentos de acceso,
- generar logs de interés para su análisis en el SIEM.

Despliegue del honeypot

El honeypot se ejecuta sobre una máquina Linux (Kali) ubicada en la red DMZ, con las siguientes características:

- Exposición controlada de servicios mediante NAT / Port Forwarding en pfSense.
- Accesibilidad desde la red WAN.
- Imposibilidad de acceso hacia LAN y DMZ2.
- Envío de logs a Elastic mediante Elastic Agent.

El agente está correctamente enrolado en Fleet y asociado a la política DMZ, específica para sistemas expuestos.



Honeypot utilizado: Cowrie

El honeypot desplegado en la red DMZ es Cowrie, un honeypot de interacción media diseñado para simular servicios SSH y Telnet vulnerables.

Cowrie permite:

- capturar comandos ejecutados por atacantes,
- analizar comportamientos posteriores a la intrusión,
- generar logs realistas de actividad maliciosa.

En este laboratorio, Cowrie actúa como sistema **deliberadamente expuesto** en la DMZ, permitiendo observar interacciones externas sin comprometer la red interna.

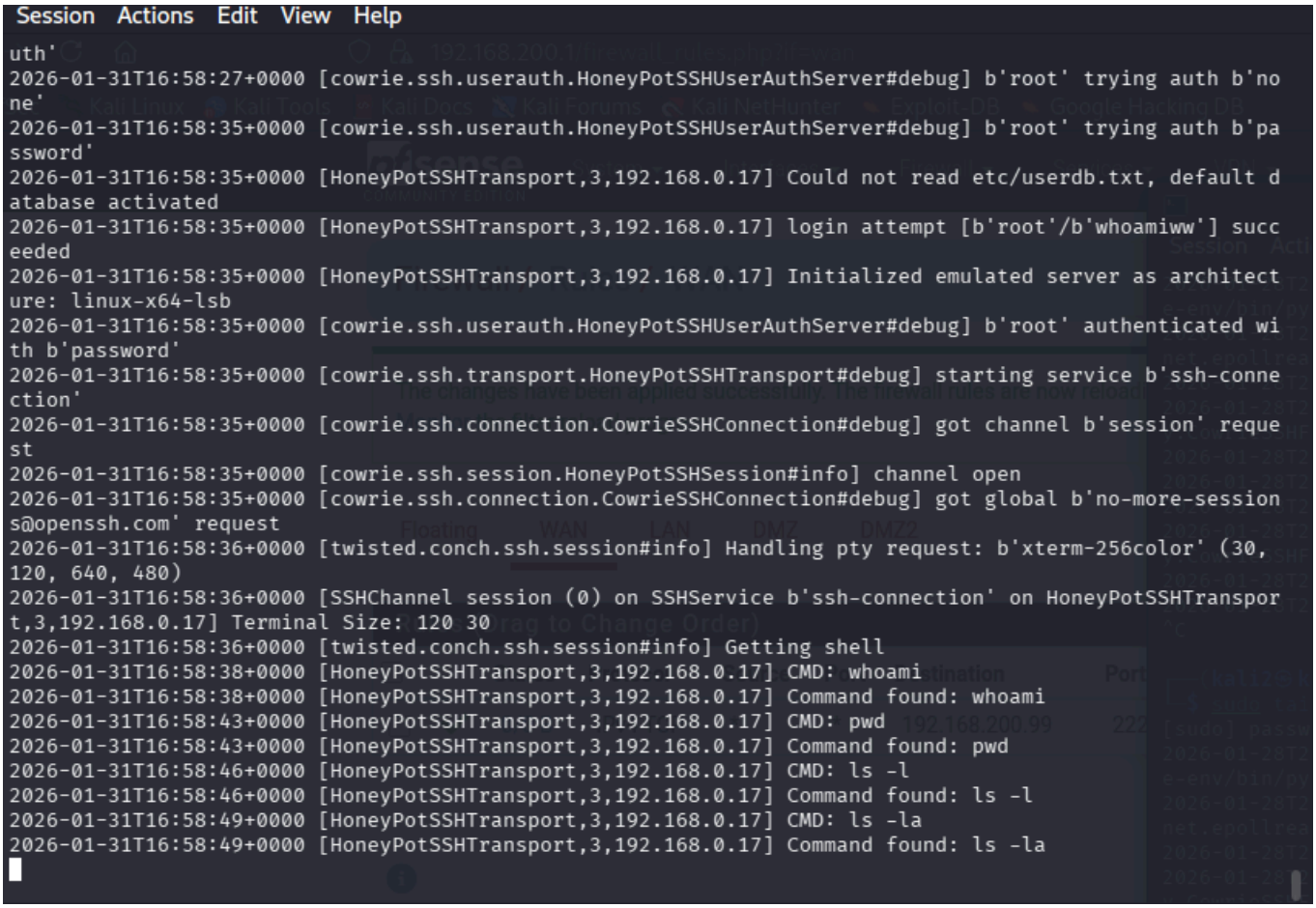
Interacciones externas y generación de eventos

Para validar el correcto funcionamiento del honeypot, se realizaron pruebas de acceso desde la red WAN, simulando conexiones externas hacia los servicios expuestos.

Estas interacciones permiten:

- generar eventos reales de acceso,
- comprobar la visibilidad de tráfico externo,
- validar que los logs se generan en el segmento de red correcto (DMZ).

Primero se comprobó igual que se hizo en la red DMZ2, que los logs aparecian y se guardaban correctamente.



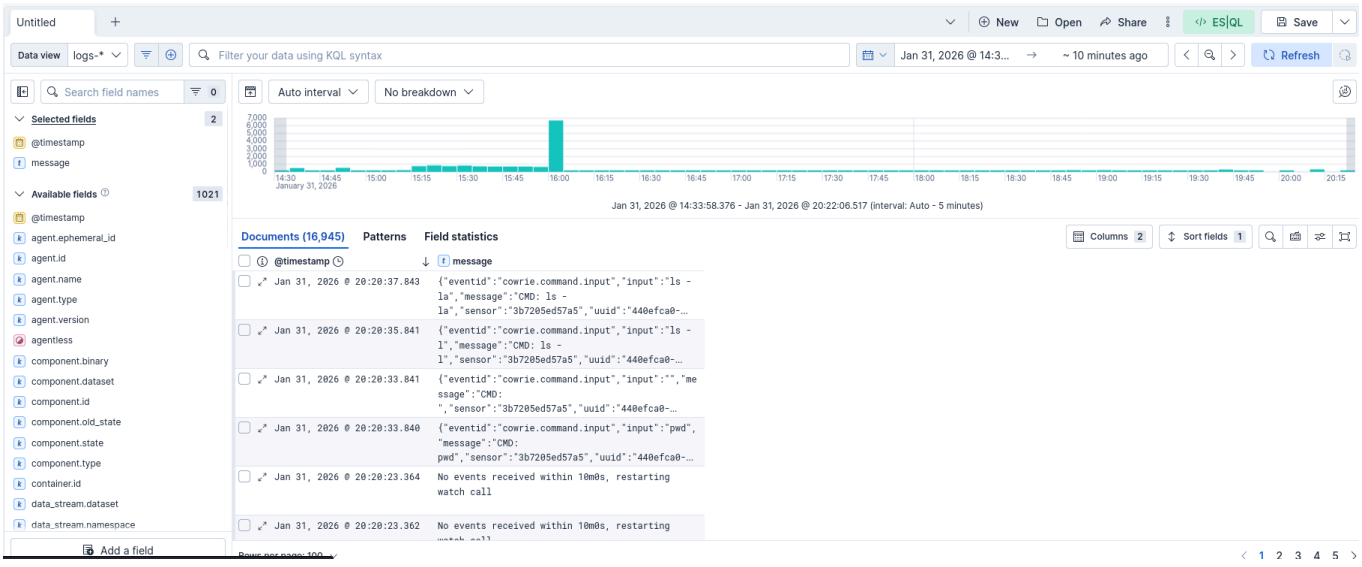
```
┌──(root㉿kali)-[/home/kali2/cowrie-logs/elastic-agent-9.2.4-linux-x86_64/elastic-agent-9.2.4-linux-x86_64]
└─# tail -n 5 /home/kali2/cowrie-logs/cowrie.json

{"eventid":"cowrie.session.params","arch":"linux-x64-lsb","message":[],"sensor":"3b7205ed57a5","uuid":"440efca0-fc58-11f0-849c-7aba4c68aaec","timestamp":"2026-01-31T19:20:10.788038Z","src_ip":"192.168.0.17","session":"e2bf9ec3d684","protocol":"ssh"}
{"eventid":"cowrie.command.input","input":"pwd","message":"CMD: pwd","sensor":"3b7205ed57a5","uuid":"440efca0-fc58-11f0-849c-7aba4c68aaec","timestamp":"2026-01-31T19:20:30.217088Z","src_ip":"192.168.0.17","session":"e2bf9ec3d684","protocol":"ssh"}
{"eventid":"cowrie.command.input","input":"","message":"CMD: ","sensor":"3b7205ed57a5","uuid":"440efca0-fc58-11f0-849c-7aba4c68aaec","timestamp":"2026-01-31T19:20:30.566924Z","src_ip":"192.168.0.17","session":"e2bf9ec3d684","protocol":"ssh"}
{"eventid":"cowrie.command.input","input":"ls -l","message":"CMD: ls -l","sensor":"3b7205ed57a5","uuid":"440efca0-fc58-11f0-849c-7aba4c68aaec","timestamp":"2026-01-31T19:20:34.117985Z","src_ip":"192.168.0.17","session":"e2bf9ec3d684","protocol":"ssh"}
{"eventid":"cowrie.command.input","input":"ls -la","message":"CMD: ls -la","sensor":"3b7205ed57a5","uuid":"440efca0-fc58-11f0-849c-7aba4c68aaec","timestamp":"2026-01-31T19:20:36.186002Z","src_ip":"192.168.0.17","session":"e2bf9ec3d684","protocol":"ssh"}

┌──(root㉿kali)-[/home/kali2/cowrie-logs/elastic-agent-9.2.4-linux-x86_64/elastic-agent-9.2.4-linux-x86_64]
└─#
```

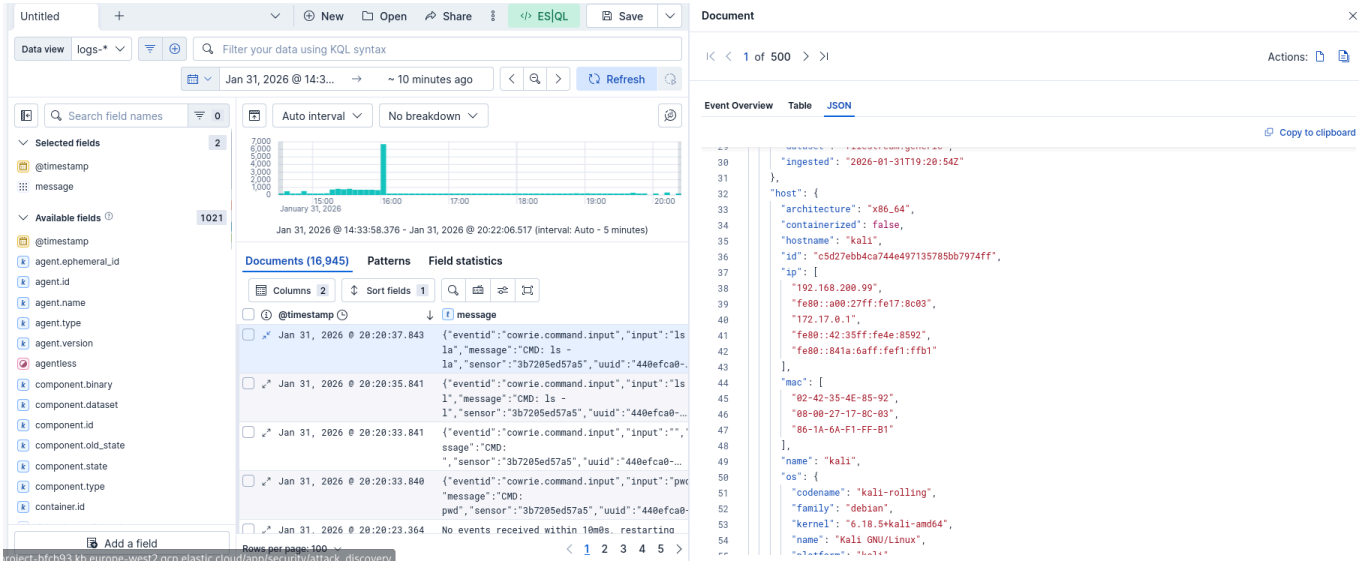
Evidencias de recepción de logs en Elastic

Aquí nos enfrentamos con unos problemas que expondremos al final, porque el agente estaba apuntando a una dirección IP y puerto que no era el que estaba configurado. Una vez configurado esto bien, se comenzaron a recibir los logs en Elastic.

Se verificó la correcta recepción de los logs generados por el honeypot en **Elastic SIEM**, observándose documentos con las siguientes características:

- IP de origen perteneciente a la red **DMZ**.
- Dataset asociado a los servicios del honeypot.
- Marca temporal coherente con las pruebas realizadas.





Ejemplo de log recibido (JSON)

A continuación se muestra un ejemplo de documento JSON correspondiente a un evento generado por el honeypot:


```
{ "host": {"architecture": "x86_64","containerized": false,"hostname": "kali","id": "c5d27ebb4ca744e497135785bb7974ff","ip": ["192.168.200.99","fe80::a00:27ff:fe17:8c03","172.17.0.1","fe80::42:35ff:fe4e:8592","fe80::841a:6aff:fef1:ffb1" }
```


Este log permite identificar:

- el origen externo de la conexión,
- el sistema expuesto que recibe la interacción,
- el segmento de red al que pertenece el evento.

(El JSON completo se incluye en el anexo.)

Fuente de logs en LAN: Windows

Finalmente entramos a crear y verificar la **fuente de logs ubicada en la red LAN**, correspondiente al **sistema interno Windows**, y completa la comparación entre fuentes y sistemas operativos dentro del entorno Blue Team.

Rol de la red LAN

La **LAN** es la red interna de confianza, destinada a sistemas no expuestos al exterior.
En este segmento se ubica el equipo **Windows 10**, que representa un activo interno típico dentro de una infraestructura corporativa.

Su función en la práctica es:

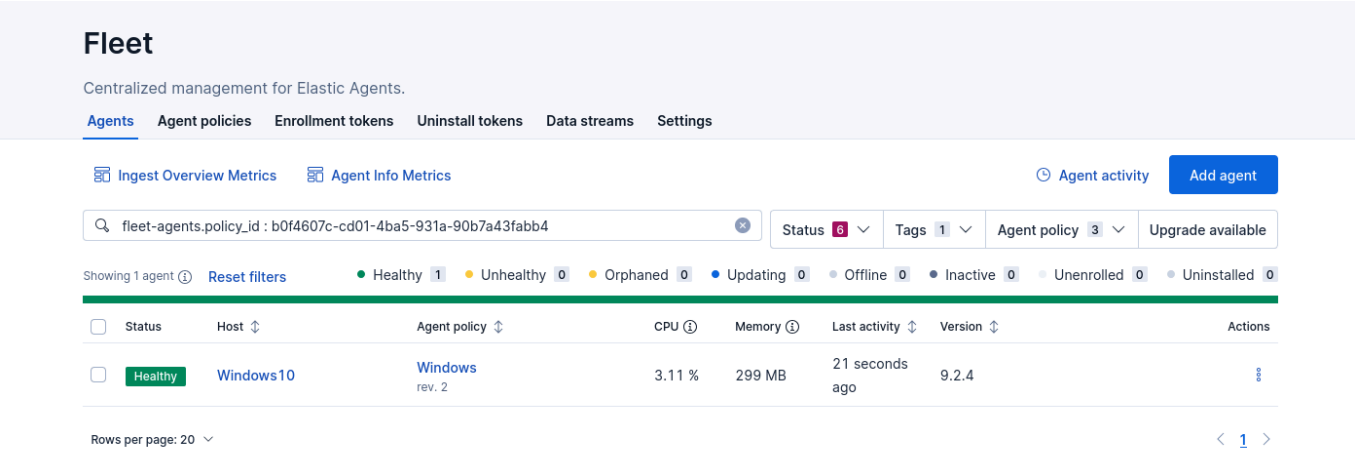
- generar logs y métricas propias de un sistema Windows,
- permitir comparar el tipo de datos recogidos frente a sistemas Linux,
- validar la correcta segmentación y envío de datos al SIEM.

Despliegue del agente en Windows

El equipo **Windows 10** dispone de **Elastic Agent** instalado y correctamente enrolado en **Fleet**, asociado a la **política LAN**.

Características relevantes:

- Agente en estado **Healthy**.
- Envío de logs y métricas del sistema.
- Comunicación saliente permitida únicamente hacia **Elastic Cloud**.



Tipología de logs y métricas recogidas

Desde el sistema Windows se recogen principalmente:

- **Métricas del sistema** (CPU, procesos, uso de recursos).
- **Logs del sistema y del agente**, según las integraciones configuradas en la política LAN.

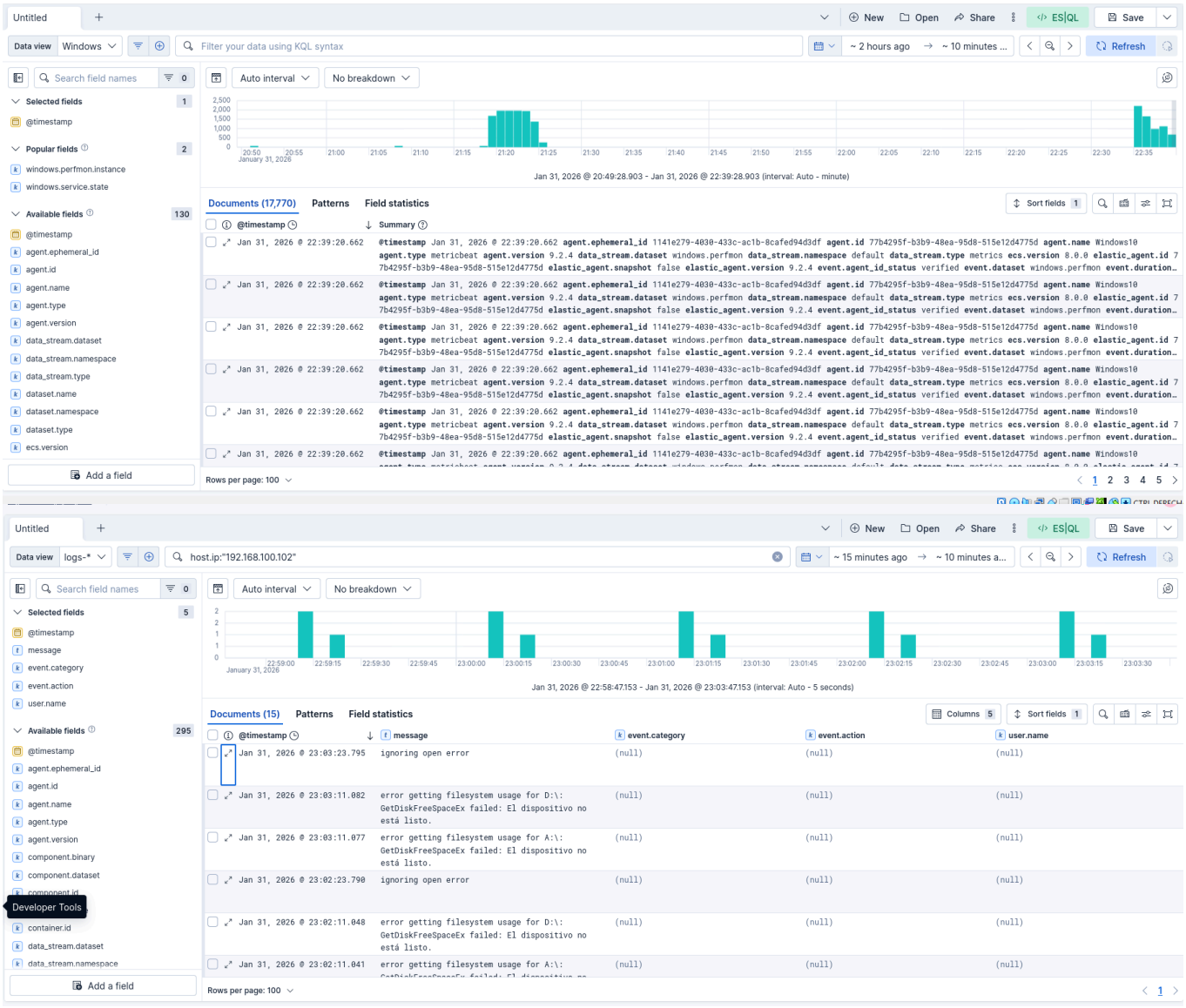
Estos datos permiten:

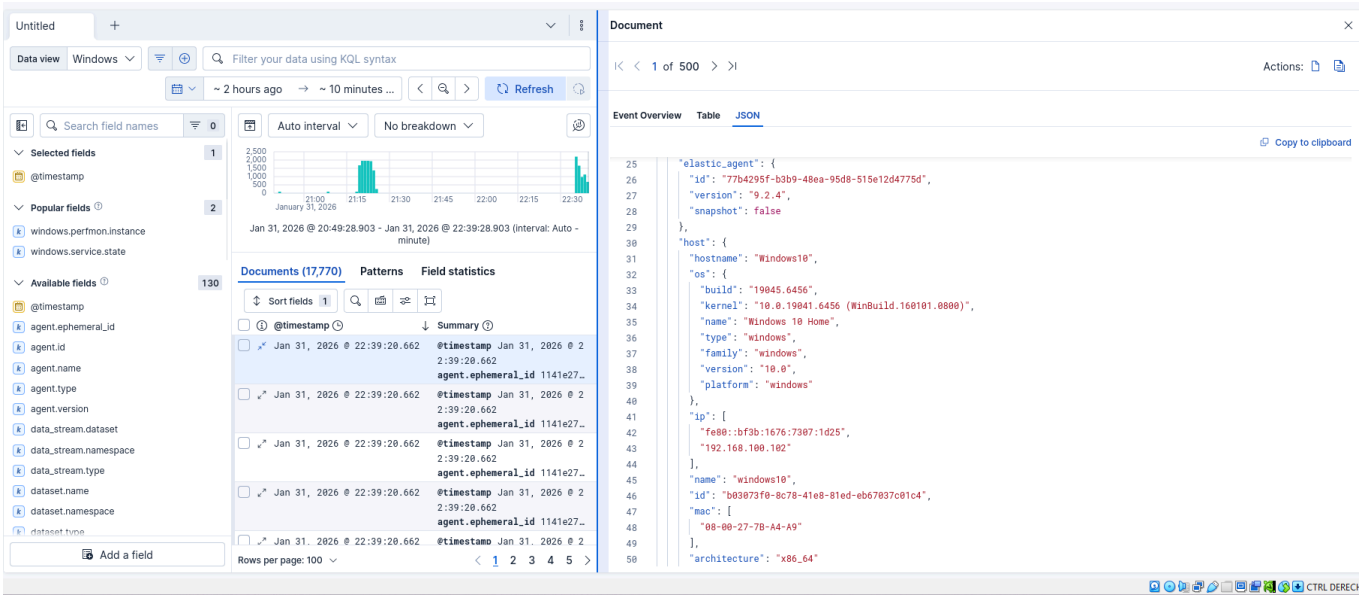
- monitorizar el estado del sistema interno,
- identificar eventos relevantes,
- validar la visibilidad del entorno Windows dentro del SIEM.

Evidencias de recepción de datos en Elastic

Se verificó la recepción de los datos generados por el sistema Windows en **Elastic SIEM**, observándose documentos con las siguientes características:

- IP de origen perteneciente a la red **LAN**.
- Dataset asociado a métricas y logs de Windows.
- Marca temporal coherente con la actividad del sistema.





Ejemplo de documento recibido (JSON)

A continuación se muestra un ejemplo de documento JSON correspondiente a datos enviados desde el sistema Windows:


```
{ "host": {"hostname": "Windows10","os": {"build": "19045.6456","kernel":
"10.0.19041.6456 (WinBuild.160101.0800)","name": "Windows 10 Home","type":
"windows","family": "windows","version": "10.0","platform": "windows" "ip":
["fe80::bf3b:1676:7307:1d25","192.168.100.102" }
```


Este documento permite identificar claramente:

- el sistema operativo de origen,
- el segmento de red (LAN),
- el tipo de información recogida.

(El JSON completo se incluye en el anexo.)

Validación global de la ingesta de logs

Una vez desplegadas todas las fuentes de logs y configuradas las políticas de agentes, se realizó una validación conjunta para confirmar que:

- cada fuente genera logs desde el **segmento de red correcto**,
- Elastic SIEM recibe y almacena dichos logs,
- es posible diferenciar claramente el origen de los eventos mediante IP y dataset.

Verificación por segmento de red

La validación se realizó mediante consultas en **Discover**, filtrando por IP de origen y dataset.

Segmento	Fuente	Tipo de logs	Evidencia
LAN	Windows 10	Métricas / logs	`192.168.100.102` LAN
DMZ	Honeypot	Logs de servicios	`192.168.200.99` DMZ
DMZ2	Suricata	Alertas / eventos	`192.168.250.100` DMZ2

Esta validación confirma que los logs no solo llegan a Elastic, sino que lo hacen desde la **red correcta**, cumpliendo los requisitos obligatorios expuestos para el ejercicio.

Incidencias encontradas durante el despliegue

Durante la implementación del laboratorio se detectaron varias incidencias técnicas, propias de un despliegue real, que requirieron análisis y ajustes de configuración. Dado que lo empecé de 0, todo limpio y borrado anterior, me encontré con varios problemas en varios puntos.

Incidencia 1 — Discrepancia de IP entre el agente y Elastic

Descripción:
En algunos casos, la IP mostrada en la interfaz web de Elastic no coincidía inicialmente con la IP configurada en la máquina origen.

Causa:

- Uso de múltiples interfaces de red.
- Resolución prioritaria de direcciones IPv6 o interfaces secundarias por parte del agente.

Resolución:

- Verificación del campo `host.ip` en los documentos JSON.
- Confirmación del segmento de red correcto mediante rangos IP.
- Uso de filtros explícitos en Discover para validar la procedencia real de los logs.

Aprendizaje:
La IP mostrada en la interfaz no siempre refleja de forma directa la red lógica; la validación debe realizarse siempre sobre los campos del documento.

Incidencia 2 — Problemas de visualización de logs

Descripción:
Inicialmente, algunos logs no aparecían en Discover pese a estar siendo ingeridos correctamente.

Causa:

- Uso de Data Views personalizados que no incluían todos los índices (`logs-*`).
- Diferencia entre métricas y logs (`metrics-*` vs `logs-*`).

Resolución:

- Revisión y ajuste de los Data Views.
- Uso del Data View genérico `logs-*` para validación.
- Confirmación de la ingesta mediante campos `event.ingested` y `@timestamp`.

Aprendizaje:
La ausencia visual de logs no implica necesariamente un fallo de ingesta, sino a menudo un problema de filtrado o visualización.

Incidencia 3 — Retraso en la aparición de eventos

Descripción:
Se observó un retardo entre la generación de un evento y su aparición en Elastic.

Causa:

- Procesos de batching y refresco propios de Elastic Agent y Elastic Cloud.

Resolución:

- Comparación entre `@timestamp` y `event.ingested`.
- Confirmación de que el retardo observado era normal (segundos).

Aprendizaje:
Elastic SIEM no trabaja en tiempo real estricto, y es importante distinguir entre generación e ingesta del evento.

Incidencia 4 — Gestión de logs en honeypot desplegado en Docker

Descripción:
Durante el despliegue del honeypot **Cowrie**, ejecutado dentro de un contenedor Docker, se detectó que los logs generados por el servicio no se almacenaban ni persistían en la ubicación esperada.

Causa:

- El honeypot se ejecuta dentro de un **contenedor Docker**.
- Por defecto, los logs quedaban:
    - en rutas internas del contenedor, o

    - enviados únicamente a `stdout/stderr`,
      sin persistencia en el sistema de ficheros del host.
- Elastic Agent, al ejecutarse en el host, no podía acceder a dichos logs sin una configuración adicional.

Resolución:

- Análisis de la configuración del contenedor y del servicio del honeypot.
- Identificación de la ruta interna donde el servicio generaba los logs.
- Modificación de la configuración para:
    - redirigir los logs a una ruta persistente,
    - o montar un **volumen Docker** que expusiera los logs al host.
- Ajuste de los parámetros necesarios para asegurar que los logs quedaran almacenados en una ubicación accesible por Elastic Agent.

Tras estos cambios, los logs del honeypot comenzaron a generarse y almacenarse correctamente, permitiendo su recolección y visualización en Elastic SIEM.

Aprendizaje:
El uso de contenedores introduce una capa adicional de abstracción que debe tenerse en cuenta a la hora de recolectar logs. Es fundamental comprender **dónde se generan realmente los logs** y cómo exponerlos de forma persistente para su integración con un SIEM.

Conclusiones finales

Esta práctica me ha permitido diseñar e implementar una **infraestructura Blue Team completa**, integrando segmentación de red, control de tráfico y centralización de logs en un entorno realista.

A través del uso de **pfSense** como elemento central de seguridad, se ha conseguido separar correctamente los distintos segmentos (**LAN, DMZ y DMZ2**), aplicando reglas de firewall coherentes con el rol de cada red y validando su correcto funcionamiento mediante pruebas de conectividad.

La integración de **Elastic SIEM** como sistema de monitorización ha permitido centralizar logs y métricas procedentes de **fuentes heterogéneas**, incluyendo:

- un sistema interno **Windows** en la red LAN,
- un **honeypot** expuesto de forma controlada en la DMZ,
- un sensor **Suricata** en una red aislada (DMZ2).

Esta diversidad de fuentes ha facilitado la comparación entre distintos tipos de eventos y sistemas operativos, reforzando la comprensión de cómo se generan y analizan los logs en

función del contexto y del segmento de red.

Durante el desarrollo del laboratorio se han presentado incidencias reales relacionadas con la visualización y validación de los logs, cuya resolución ha contribuido a afianzar conceptos clave como:

- la interpretación de campos relevantes en documentos JSON,
- la diferencia entre generación e ingesta de eventos,
- la importancia de los Data Views en Elastic.

En conjunto, la práctica me ha aportado una visión práctica y realista del trabajo Blue Team, acercándose a escenarios habituales en entornos profesionales de monitorización y respuesta ante incidentes.

En definitiva, este laboratorio ha servido como una base sólida para comprender la **importancia de la segmentación, la observabilidad y la gestión centralizada de logs** en la defensa de infraestructuras modernas.

Anexos

Anexo A - Logs DMZ2 (Suricata)

Anexo B - Logs DMZ (Honeypot)

Anexo C - Logs LAN (Windows)

- ## Anexo A


```
{
"_index": ".ds-logs-suricata.eve-default-2026.01.28-000001",
"_id": "AZwUhfsIDU7d4tanJF-o",
"_version": 1,
"_ignored": [
"event.type",
"suricata.eve.direction",
"suricata.eve.files",
```

```
"suricata.eve.flow.dest_ip",

"suricata.eve.flow.dest_port",

"suricata.eve.flow.src_ip",

"suricata.eve.flow.src_port",

"suricata.eve.tc_progress",

"suricata.eve.ts_progress"

],

"_source": {

"@timestamp": "2026-01-31T14:47:08.271Z",

"agent": {

"ephemeral_id": "9d9388aa-1a43-4ca2-9a39-29a682c50256",

"id": "f8a1090c-222a-4827-b2cb-fc75503d0771",

"name": "BootCamp-Kali",

"type": "filebeat",

"version": "9.2.4"

},

"data_stream": {

"dataset": "suricata.eve",

"namespace": "default",

"type": "logs"

},

"destination": {

"address": "192.168.250.100",
```

```
"bytes": 46806,
"domain": "lsi2.ugr.es",
"ip": "192.168.250.100",
"packets": 33,
"port": 39742
},
"ecs": {
"version": "8.17.0"
},
"elastic_agent": {
"id": "f8a1090c-222a-4827-b2cb-fc75503d0771",
"snapshot": false,
"version": "9.2.4"
},
"event": {
"agent_id_status": "verified",
"category": [
"network",
"intrusion_detection"
],
"created": "2026-01-31T14:47:08.410Z",
"dataset": "suricata.eve",
"ingested": "2026-01-31T14:47:09Z",
```

```
"kind": "alert",
"severity": 2,
"start": "2026-01-31T14:47:08.148Z",
"type": [
"allowed"
]
},
"http": {
"request": {
"method": "GET"
},
"response": {
"body": {
"bytes": 43207
},
"status_code": 200
}
},
"input": {
"type": "log"
},
"log": {
"file": {
```

```
"path": "/var/log/suricata/eve.json"

},

"offset": 13827286

},

"message": "Descarga de archivo detectado",

"network": {

"bytes": 48659,

"community_id": "1:/dvYz5hObC+CM7gmlkIBiDkWz+I=",

"packets": 64,

"protocol": "http",

"transport": "tcp"

},

"observer": {

"hostname": "BootCamp-Kali",

"ip": [

"192.168.250.100",

"fe80::a00:27ff:fec2:f035",

"172.17.0.1"

],

"mac": [

"02-42-61-7F-E0-A9",

"08-00-27-C2-F0-35"

],
```

```
"product": "Suricata",
"type": "ids",
"vendor": "OISF"
},
"related": {
"hosts": [
"lsi2.ugr.es"
],
"ip": [
"150.214.190.179",
"192.168.250.100"
]
},
"rule": {
"category": "Descarga de archivo detectado",
"id": "3",
"name": "Archivo PDF Detectado"
},
"source": {
"address": "150.214.190.179",
"as": {
"number": 198096,
"organization": {
```

```
"name": "Junta de Andalucia"

}

},

"bytes": 1853,

"geo": {

"city_name": "Seville",

"continent_name": "Europe",

"country_iso_code": "ES",

"country_name": "Spain",

"location": {

"lat": 37.41069998592138,

"lon": -5.964300027117133

},

"region_iso_code": "ES-SE",

"region_name": "Seville"

},

"ip": "150.214.190.179",

"packets": 31,

"port": 80

},

"suricata": {

"eve": {

"alert": {
```

```
"category": "Descarga de archivo detectado",
"gid": 1,
"rev": 0,
"signature": "Archivo PDF Detectado",
"signature_id": 3
},
"direction": "to_client",
"event_type": "alert",
"files": [
{
"filename": "/~jmguirao/isln/archivos/http1.pdf",
"size": 43207,
"stored": false,
"state": "UNKNOWN",
"tx_id": 0,
"gaps": false,
"storing": true,
"sid": [
3
]
}
],
"flow": {
```

```
"dest_ip": "150.214.190.179",

"dest_port": 80,

"src_ip": "192.168.250.100",

"src_port": 39742

},

"flow_id": "1200375218362962",

"http": {

"http_content_type": "application/pdf",

"protocol": "HTTP/1.1"

},

"in_iface": "eth0",

"ip_v": 4,

"pkt_src": "wire/pcap",

"tc_progress": "response_body",

"ts_progress": "request_complete",

"tx_id": 0

}

},

"tags": [

"forwarded",

"suricata-eve"

],

"url": {
```

```
"domain": "lsi2.ugr.es",
"original": "/~jmguirao/isln/archivos/http1.pdf",
"path": "/~jmguirao/isln/archivos/http1.pdf"
},
"user_agent": {
"device": {
"name": "Other"
},
"name": "Wget",
"original": "Wget/1.25.0",
"version": "1.25.0"
}
},
"fields": {
"rule.id": [
"3"
],
"elastic_agent.version": [
"9.2.4"
],
"event.category": [
"network",
"intrusion_detection"
```

```
],
"suricata.eve.files.tx_id": [
0
],
"suricata.eve.tx_id": [
0
],
"user_agent.original.text": [
"Wget/1.25.0"
],
"suricata.eve.flow.dest_ip": [
"150.214.190.179"
],
"observer.vendor": [
"OISF"
],
"source.geo.region_name": [
"Seville"
],
"suricata.eve.alert.signature": [
"Archivo PDF Detectado"
],
"suricata.eve.http.protocol": [
```

```
"HTTP/1.1"

],

"source.ip": [

"150.214.190.179"

],

"agent.name": [

"BootCamp-Kali"

],

"destination.address": [

"192.168.250.100"

],

"suricata.eve.event_type": [

"alert"

],

"network.community_id": [

"1:/dvYz5hObC+CM7gmlkIBiDkWz+I="

],

"event.agent_id_status": [

"verified"

],

"http.response.status_code": [

200

],
```

```
"suricata.eve.flow_id": [
"1200375218362962"
],
"source.geo.city_name": [
"Seville"
],
"user_agent.original": [
"Wget/1.25.0"
],
"event.severity": [
2
],
"source.packets": [
31
],
"input.type": [
"log"
],
"suricata.eve.in_iface": [
"eth0"
],
"tags": [
"forwarded",
```

```
"suricata-eve"

],

"url.path": [

"/~jmguirao/isln/archivos/http1.pdf"

],

"agent.id": [

"f8a1090c-222a-4827-b2cb-fc75503d0771"

],

"source.port": [

80

],

"suricata.eve.direction": [

"to_client"

],

"destination.bytes": [

46806

],

"event.start": [

"2026-01-31T14:47:08.148Z"

],

"source.as.number": [

198096

],
```

```
"destination.port": [
39742
],
"suricata.eve.files.size": [
43207
],
"destination.packets": [
33
],
"suricata.eve.alert.category": [
"Descarga de archivo detectado"
],
"agent.type": [
"filebeat"
],
"related.ip": [
"150.214.190.179",
"192.168.250.100"
],
"suricata.eve.files.storing": [
true
],
"observer.product": [
```

```
"Suricata"

],

"elastic_agent.snapshot": [

false

],

"suricata.eve.pkt_src": [

"wire/pcap"

],

"suricata.eve.flow.src_port": [

39742

],

"elastic_agent.id": [

"f8a1090c-222a-4827-b2cb-fc75503d0771"

],

"suricata.eve.ip_v": [

4

],

"destination.ip": [

"192.168.250.100"

],

"observer.hostname": [

"BootCamp-Kali"

],
```

```
"event.ingested": [
"2026-01-31T14:47:09.000Z"
],
"@timestamp": [
"2026-01-31T14:47:08.271Z"
],
"data_stream.dataset": [
"suricata.eve"
],
"log.file.path": [
"/var/log/suricata/eve.json"
],
"url.domain": [
"lsi2.ugr.es"
],
"suricata.eve.files.state": [
"UNKNOWN"
],
"agent.ephemeral_id": [
"9d9388aa-1a43-4ca2-9a39-29a682c50256"
],
"suricata.eve.http.http_content_type": [
"application/pdf"
```

```
],
"user_agent.device.name": [
"Other"
],
"suricata.eve.tc_progress": [
"response_body"
],
"suricata.eve.alert.rev": [
0
],
"url.original.text": [
"/~jmguirao/isln/archivos/http1.pdf"
],
"http.request.method": [
"GET"
],
"suricata.eve.flow.src_ip": [
"192.168.250.100"
],
"observer.mac": [
"02-42-61-7F-E0-A9",
"08-00-27-C2-F0-35"
],
```

```
"user_agent.version": [
"1.25.0"
],
"source.geo.region_iso_code": [
"ES-SE"
],
"suricata.eve.alert.gid": [
1
],
"event.kind": [
"alert"
],
"rule.name": [
"Archivo PDF Detectado"
],
"network.packets": [
64
],
"log.offset": [
13827286
],
"user_agent.name": [
"Wget"
```

```
],
"destination.domain": [
"lsi2.ugr.es"
],
"data_stream.type": [
"logs"
],
"ecs.version": [
"8.17.0"
],
"observer.type": [
"ids"
],
"event.created": [
"2026-01-31T14:47:08.410Z"
],
"agent.version": [
"9.2.4"
],
"related.hosts": [
"lsi2.ugr.es"
],
"observer.ip": [
```

```
"192.168.250.100",
"fe80::a00:27ff:fec2:f035",
"172.17.0.1"
],
"suricata.eve.flow.dest_port": [
80
],
"source.geo.location": [
{
"coordinates": [
-5.964300027117133,
37.41069998592138
],
"type": "Point"
}
],
"source.address": [
"150.214.190.179"
],
"suricata.eve.alert.signature_id": [
3
],
"event.module": [
```

```
"suricata"

],

"network.protocol": [

"http"

],

"suricata.eve.files.filename": [

"/~jmguirao/isln/archivos/http1.pdf"

],

"source.geo.country_iso_code": [

"ES"

],

"network.bytes": [

48659

],

"source.bytes": [

1853

],

"suricata.eve.files.sid": [

3

],

"source.as.organization.name.text": [

"Junta de Andalucia"

],
```

```
"data_stream.namespace": [
"default"
],
"suricata.eve.files.stored": [
false
],
"source.as.organization.name": [
"Junta de Andalucia"
],
"source.geo.continent_name": [
"Europe"
],
"message": [
"Descarga de archivo detectado"
],
"http.response.body.bytes": [
43207
],
"network.transport": [
"tcp"
],
"url.original": [
"/~jmguirao/isln/archivos/http1.pdf"
```

```
],
"suricata.eve.files.gaps": [
false
],
"suricata.eve.ts_progress": [
"request_complete"
],
"event.type": [
"allowed"
],
"source.geo.country_name": [
"Spain"
],
"rule.category": [
"Descarga de archivo detectado"
],
"event.dataset": [
"suricata.eve"
]
}
}
```


- ## Anexo B

```
{
"_index": ".ds-logs-filestream.generic-default-2026.01.31-000001",
"_id": "AZwVgPsIDU7d4tZHTbKz",
"_version": 1,
"_source": {
"@timestamp": "2026-01-31T19:20:37.843Z",
"agent": {
"ephemeral_id": "d41d0327-ee68-49c4-af13-06fbca18fb00",
"id": "e6483af3-7f70-4388-b8ce-f0b6f2cd438f",
"name": "kali",
"type": "filebeat",
"version": "9.2.4"
},
"data_stream": {
"dataset": "filestream.generic",
"namespace": "default",
"type": "logs"
},
"ecs": {
"version": "8.0.0"
},
"elastic_agent": {
"id": "e6483af3-7f70-4388-b8ce-f0b6f2cd438f",
```

```
"snapshot": false,
"version": "9.2.4"
},
"event": {
"agent_id_status": "verified",
"dataset": "filestream.generic",
"ingested": "2026-01-31T19:20:54Z"
},
"host": {
"architecture": "x86_64",
"containerized": false,
"hostname": "kali",
"id": "c5d27ebb4ca744e497135785bb7974ff",
"ip": [
"192.168.200.99",
"fe80::a00:27ff:fe17:8c03",
"172.17.0.1",
"fe80::42:35ff:fe4e:8592",
"fe80::841a:6aff:fef1:ffb1"
],
"mac": [
"02-42-35-4E-85-92",
"08-00-27-17-8C-03",
```

```
"86-1A-6A-F1-FF-B1"

],

"name": "kali",

"os": {

"codename": "kali-rolling",

"family": "debian",

"kernel": "6.18.5+kali-amd64",

"name": "Kali GNU/Linux",

"platform": "kali",

"type": "linux",

"version": "2025.4"

}

},

"input": {

"type": "filestream"

},

"log": {

"file": {

"device_id": "2049",

"fingerprint":
"cb4984eccbbb0ad4defc48f08063e90bba97d5a1bfdbba5d184a8b622df51e76",

"inode": "1837840",

"path": "/home/kali2/cowrie-logs/cowrie.json"

},
```

```
"offset": "23020"

},

"message": "{\"eventid\":\"cowrie.command.input\",\"input\":\"ls -
la\",\"message\":\"CMD: ls -
la\",\"sensor\":\"3b7205ed57a5\",\"uuid\":\"440efca0-fc58-11f0-849c-
7aba4c68aaec\",\"timestamp\":\"2026-01-
31T19:20:36.186002Z\",\"src_ip\":\"192.168.0.17\",\"session\":\"e2bf9ec3d684\"
,\"protocol\":\"ssh\"}"

},

"fields": {

"elastic_agent.version": [

"9.2.4"

],

"host.os.name.text": [

"Kali GNU/Linux"

],

"host.hostname": [

"kali"

],

"host.mac": [

"02-42-35-4E-85-92",

"08-00-27-17-8C-03",

"86-1A-6A-F1-FF-B1"

],

"host.ip": [
```

```
"192.168.200.99",
"fe80::a00:27ff:fe17:8c03",
"172.17.0.1",
"fe80::42:35ff:fe4e:8592",
"fe80::841a:6aff:fef1:ffb1"
],
"agent.type": [
"filebeat"
],
"event.module": [
"filestream"
],
"host.os.version": [
"2025.4"
],
"host.os.kernel": [
"6.18.5+kali-amd64"
],
"log.file.device_id": [
"2049"
],
"host.os.name": [
"Kali GNU/Linux"
```

```
],
"agent.name": [
"kali"
],
"elastic_agent.snapshot": [
false
],
"host.name": [
"kali"
],
"event.agent_id_status": [
"verified"
],
"host.id": [
"c5d27ebb4ca744e497135785bb7974ff"
],
"host.os.type": [
"linux"
],
"elastic_agent.id": [
"e6483af3-7f70-4388-b8ce-f0b6f2cd438f"
],
"data_stream.namespace": [
```

```
"default"

],

"host.os.codename": [

"kali-rolling"

],

"input.type": [

"filestream"

],

"log.offset": [

"23020"

],

"message": [

"{\"eventid\":\"cowrie.command.input\",\"input\":\"ls -la\",\"message\":\"CMD:
ls -la\",\"sensor\":\"3b7205ed57a5\",\"uuid\":\"440efca0-fc58-11f0-849c-
7aba4c68aaec\",\"timestamp\":\"2026-01-
31T19:20:36.186002Z\",\"src_ip\":\"192.168.0.17\",\"session\":\"e2bf9ec3d684\"
,\"protocol\":\"ssh\"}"

],

"data_stream.type": [

"logs"

],

"host.architecture": [

"x86_64"

],

"event.ingested": [
```

```
"2026-01-31T19:20:54.000Z"

],

"@timestamp": [

"2026-01-31T19:20:37.843Z"

],

"agent.id": [

"e6483af3-7f70-4388-b8ce-f0b6f2cd438f"

],

"ecs.version": [

"8.0.0"

],

"host.containerized": [

false

],

"host.os.platform": [

"kali"

],

"log.file.inode": [

"1837840"

],

"data_stream.dataset": [

"filestream.generic"

],
```

```
"log.file.path": [
"/home/kali2/cowrie-logs/cowrie.json"
],
"agent.ephemeral_id": [
"d41d0327-ee68-49c4-af13-06fbca18fb00"
],
"agent.version": [
"9.2.4"
],
"log.file.fingerprint": [
"cb4984eccbbb0ad4defc48f08063e90bba97d5a1bfdbba5d184a8b622df51e76"
],
"host.os.family": [
"debian"
],
"event.dataset": [
"filestream.generic"
]
}
}
```


- ## Anexo C


```
{
"_index": ".ds-metrics-windows.perfmon-default-2026.01.31-000001",
```

```
"_id": "-Nj_FZwBs1_PuHA4MBPM",
"_version": 1,
"_source": {
"agent": {
"name": "Windows10",
"id": "77b4295f-b3b9-48ea-95d8-515e12d4775d",
"type": "metricbeat",
"ephemeral_id": "1141e279-4030-433c-ac1b-8cafed94d3df",
"version": "9.2.4"
},
"@timestamp": "2026-01-31T21:39:20.662Z",
"ecs": {
"version": "8.0.0"
},
"data_stream": {
"namespace": "default",
"type": "metrics",
"dataset": "windows.perfmon"
},
"service": {
"type": "windows"
},
"elastic_agent": {
```

```
"id": "77b4295f-b3b9-48ea-95d8-515e12d4775d",
"version": "9.2.4",
"snapshot": false
},
"host": {
"hostname": "Windows10",
"os": {
"build": "19045.6456",
"kernel": "10.0.19041.6456 (WinBuild.160101.0800)",
"name": "Windows 10 Home",
"type": "windows",
"family": "windows",
"version": "10.0",
"platform": "windows"
},
"ip": [
"fe80::bf3b:1676:7307:1d25",
"192.168.100.102"
],
"name": "windows10",
"id": "b03073f0-8c78-41e8-81ed-eb67037c01c4",
"mac": [
"08-00-27-7B-A4-A9"
```

```
],
"architecture": "x86_64"
},
"metricset": {
"period": 10000,
"name": "perfmon"
},
"windows": {
"perfmon": {
"instance": "svchost",
"metrics": {
"cpu_perc": 0
},
"object": "Process"
}
},
"event": {
"duration": 1030525300,
"agent_id_status": "verified",
"ingested": "2026-01-31T21:39:31Z",
"module": "windows",
"dataset": "windows.perfmon"
}
```

```
},
"fields": {
"elastic_agent.version": [
"9.2.4"
],
"host.os.name.text": [
"Windows 10 Home"
],
"host.hostname": [
"Windows10"
],
"host.mac": [
"08-00-27-7B-A4-A9"
],
"host.os.build": [
"19045.6456"
],
"service.type": [
"windows"
],
"host.ip": [
"fe80::bf3b:1676:7307:1d25",
"192.168.100.102"
```

```
],
"agent.type": [
"metricbeat"
],
"event.module": [
"windows"
],
"agent.name.text": [
"Windows10"
],
"host.os.version": [
"10.0"
],
"windows.perfmon.metrics.cpu_perc": [
0
],
"host.os.kernel": [
"10.0.19041.6456 (WinBuild.160101.0800)"
],
"host.os.name": [
"Windows 10 Home"
],
"agent.name": [
```

```
"Windows10"

],

"elastic_agent.snapshot": [

false

],

"host.name": [

"windows10"

],

"event.agent_id_status": [

"verified"

],

"host.id": [

"b03073f0-8c78-41e8-81ed-eb67037c01c4"

],

"metricset.name.text": [

"perfmon"

],

"host.os.type": [

"windows"

],

"elastic_agent.id": [

"77b4295f-b3b9-48ea-95d8-515e12d4775d"

],
```

```
"windows.perfmon.object": [
"Process"
],
"data_stream.namespace": [
"default"
],
"windows.perfmon.instance": [
"svchost"
],
"metricset.period": [
10000
],
"data_stream.type": [
"metrics"
],
"event.duration": [
1030525300
],
"host.architecture": [
"x86_64"
],
"metricset.name": [
"perfmon"
```

```
],
"event.ingested": [
"2026-01-31T21:39:31.000Z"
],
"@timestamp": [
"2026-01-31T21:39:20.662Z"
],
"agent.id": [
"77b4295f-b3b9-48ea-95d8-515e12d4775d"
],
"ecs.version": [
"8.0.0"
],
"host.os.platform": [
"windows"
],
"data_stream.dataset": [
"windows.perfmon"
],
"agent.ephemeral_id": [
"1141e279-4030-433c-ac1b-8cafed94d3df"
],
"agent.version": [
```

```
"9.2.4"
],
"host.os.family": [
"windows"
],
"event.dataset": [
"windows.perfmon"
]
}
}
```