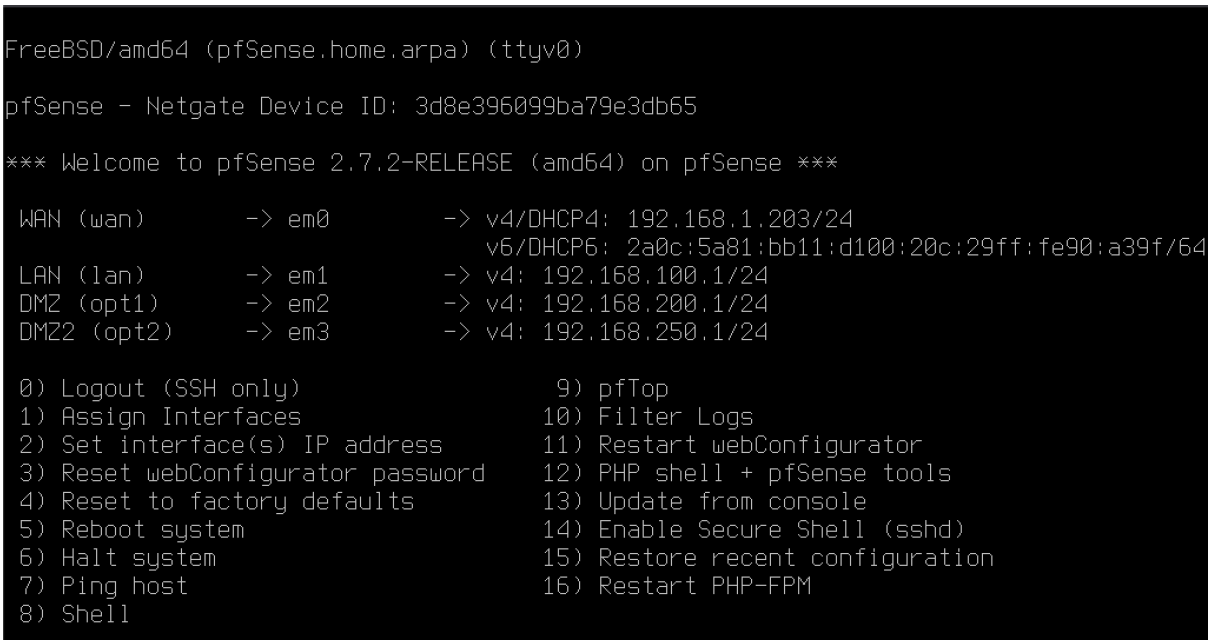
Home Lab

Firewall and SIEM Configuration

In this lab, a **network infrastructure** was designed and deployed with a focus on **system defense and security monitoring**.
The environment includes a **firewall and router based on pfSense**, which manage the routing of external traffic toward internal devices, while enforcing filtering policies to ensure both security and proper network functionality.



The network topology was divided into three main segments:

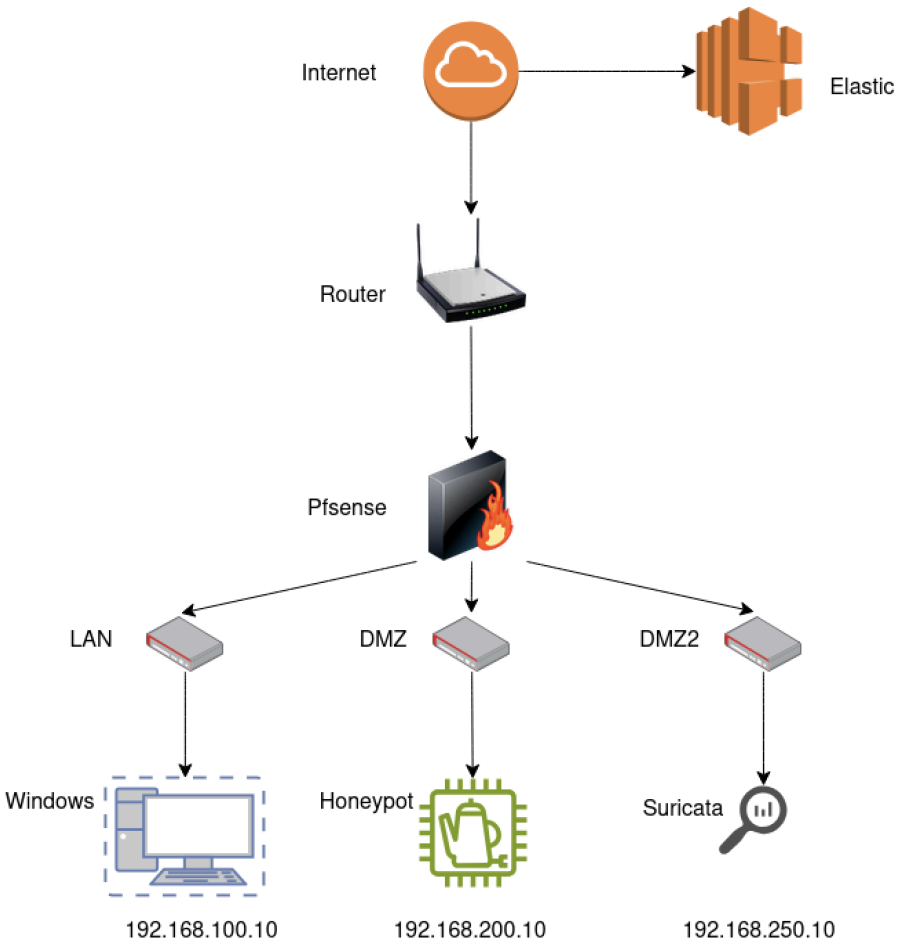
- **LAN Subnet (192.168.100.0/24):** contains a single Windows workstation configured with the static IP address **192.168.100.10**. This subnet represents the secure, internal corporate environment.
- **DMZ Subnet (192.168.200.0/24):** hosts a **honeypot** designed to collect data from potential attackers. The system, with IP **192.168.200.10**, is intentionally exposed to external connections, simulating a vulnerable service.
- **DMZ2 Subnet (192.168.250.0/24):** includes a Linux-based machine with IP **192.168.250.10**, running both an **Apache2 web server** and a **Suricata IDS**, configured to generate alerts upon detecting suspicious traffic or behavior.

All systems in the lab have an **Elastic Stack (ELK) agent** installed, enabling real-time

monitoring and data collection.
Through this integration, event logs are centralized, allowing for anomaly detection and the identification of attack patterns across the network segments.
For the **SIEM component**, **Kibana** was used as the main visualization and analysis interface.
Each host's Elastic Agent was configured with its own log collection policy and operational parameters, facilitating **event tracking and correlation** between devices and network zones.

pfSense Firewall Configuration

The **pfSense firewall** serves as the central control point for managing traffic between the internal subnets and the external network (WAN).
Its configuration aims to ensure proper **network segmentation**, **isolation of exposed services**, and **protection of internal assets** from potential threats.

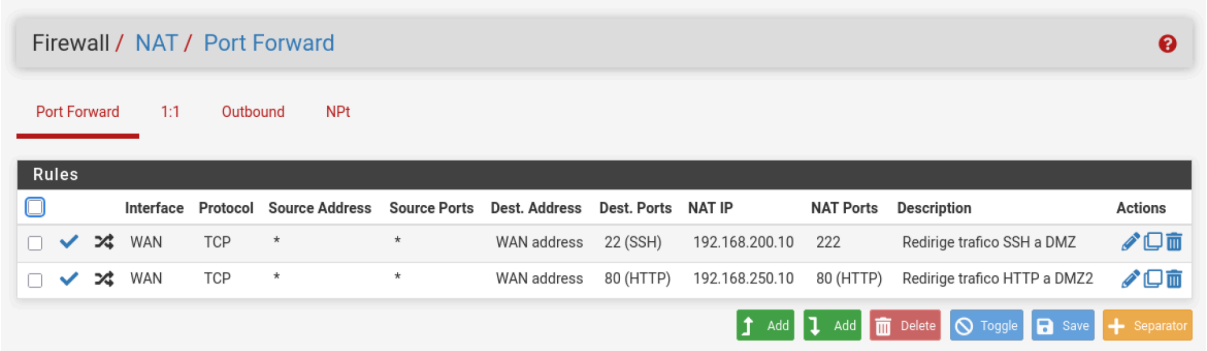


NAT Rules and Traffic Redirection

Two **NAT (Network Address Translation)** rules were created to redirect incoming traffic from the **WAN interface** to the appropriate devices within the DMZ zones:

- **HTTP Traffic (port 80):** redirected from the WAN interface to **192.168.250.10** (the Apache web server located in DMZ2).
  This rule allows external users to access the web service hosted in the lab environment.
- **SSH Traffic (external port 22 → internal port 222):** redirected from the WAN interface to **192.168.200.10** (the honeypot in DMZ).
  This setup simulates an intentionally exposed vulnerable service, allowing attackers to interact with the honeypot and generate behavioral data for later analysis.

The NAT rules automatically generate the corresponding rules on the **WAN interface**, ensuring that connections are processed according to the defined security policies.

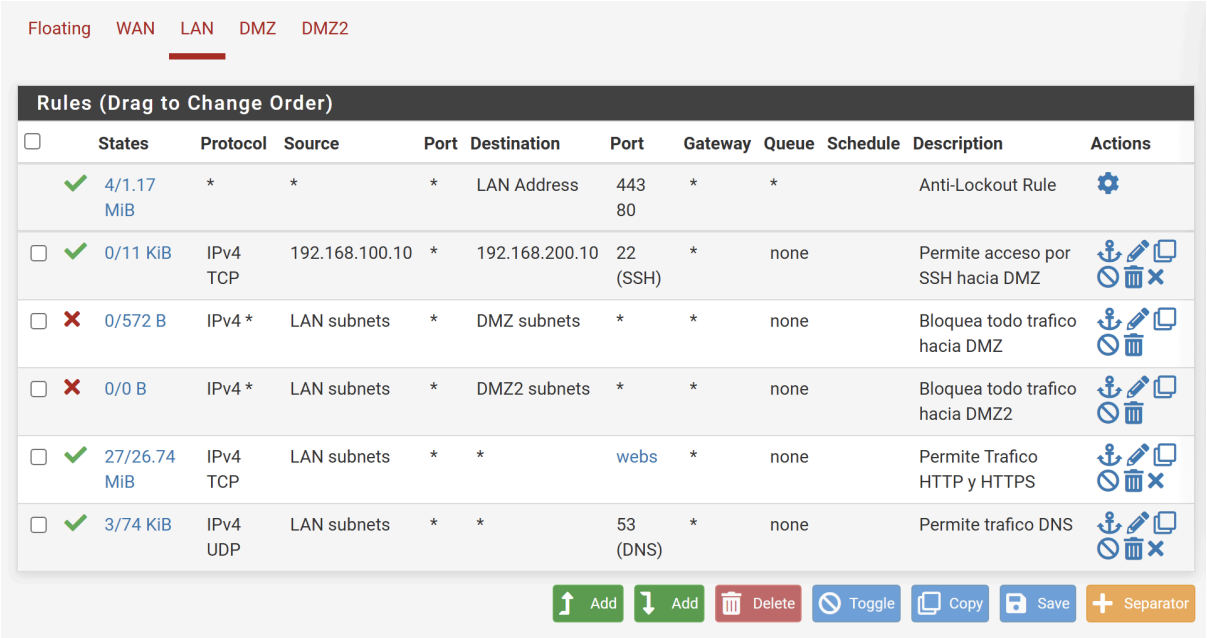


Rules per Subnet

LAN Subnet (192.168.100.0/24)

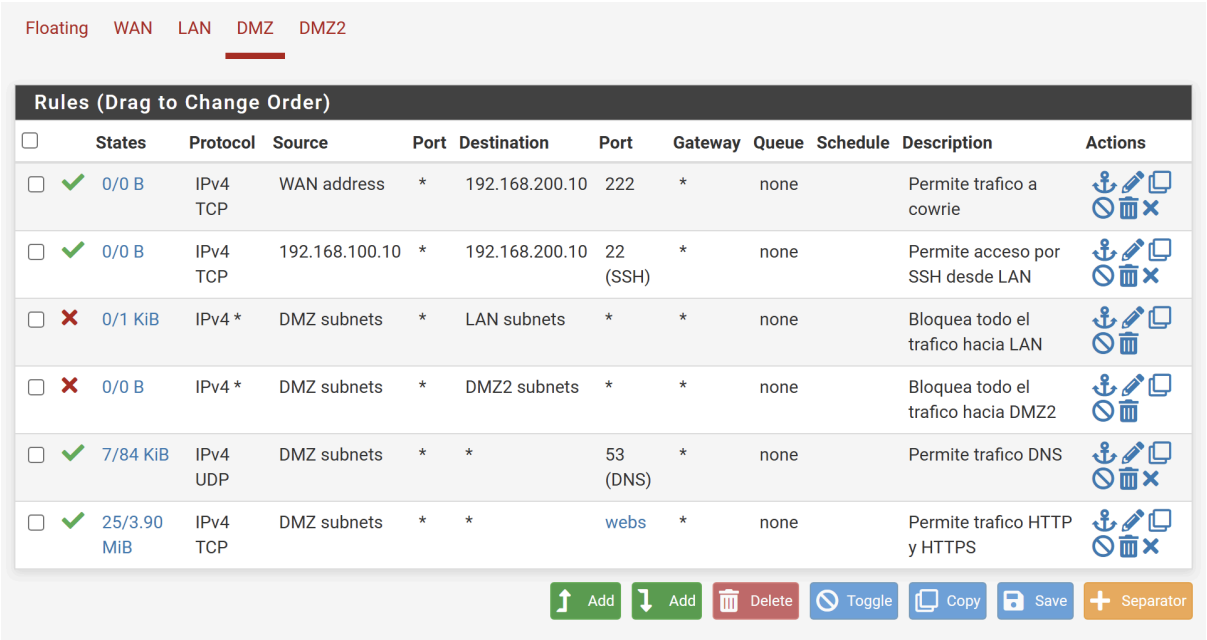
- Allow **SSH traffic** to DMZ (192.168.200.10) for honeypot administration.
- Block all other traffic to DMZ and DMZ2 to prevent direct communication between segments.
- Allow outbound **DNS (UDP 53)**, **HTTP (TCP 80)**, and **HTTPS (TCP 443)** traffic.

These rules are placed at the end of the list since pfSense applies firewall policies from top to bottom.



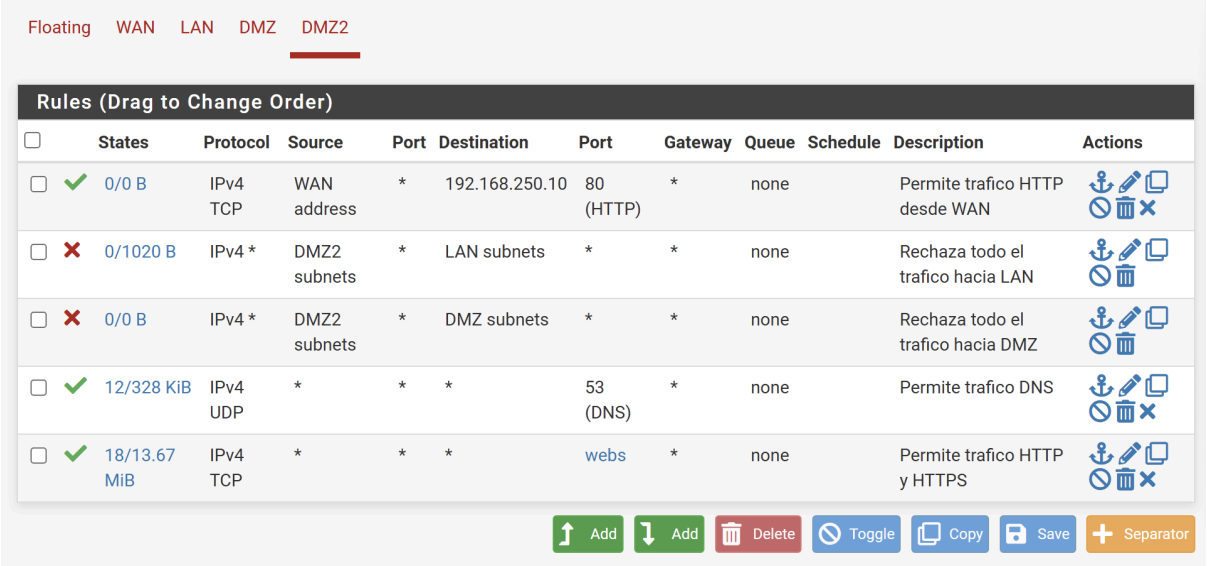
DMZ Subnet (192.168.200.0/24)

- Allow incoming traffic from the **WAN address** (pfSense public IP) to **192.168.200.10** on port **222/TCP**.
- Allow SSH access from the **LAN** (192.168.100.10) exclusively for honeypot management.
- Block all traffic to the **LAN** and **DMZ2** networks, maintaining isolation between zones.
- Allow **DNS (UDP)**, **HTTP (TCP)**, and **HTTPS (TCP)** traffic at the end of the rule set.



DMZ2 Subnet (192.168.250.0/24)

- Allow traffic from the **WAN address** to **192.168.250.10** (the Apache web server).
- Block all traffic to the **LAN** and **DMZ** networks to ensure complete separation between environments.
- Allow **DNS (UDP)**, **HTTP (TCP)**, and **HTTPS (TCP)** traffic as the final rule in the chain.



Result

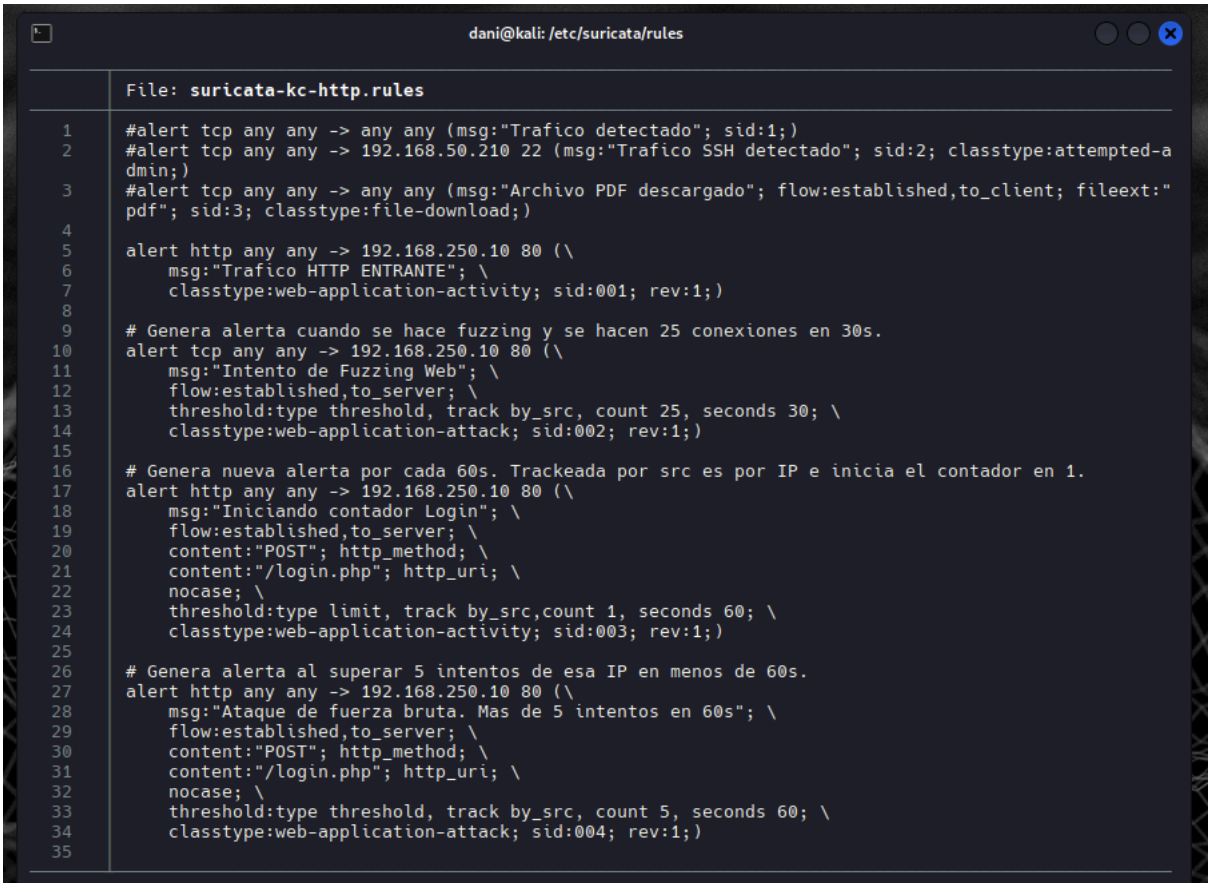
This configuration ensures:

- A **controlled and segmented traffic flow** between networks.
- Only **necessary services** (web and honeypot) are accessible from the outside.
- **Internal hosts remain isolated and protected** from external access.

By applying these policies, the network architecture follows the principles of **defense in depth** and **network segmentation**, both essential concepts within Blue Team operations.

Suricata

In the DMZ2 subnet, where the Apache web server is hosted, the intrusion detection system **Suricata** was deployed. The IDS was configured in **passive mode (detection only)** so that it generates alerts when detecting anomalous behavior or attack patterns without blocking network traffic.



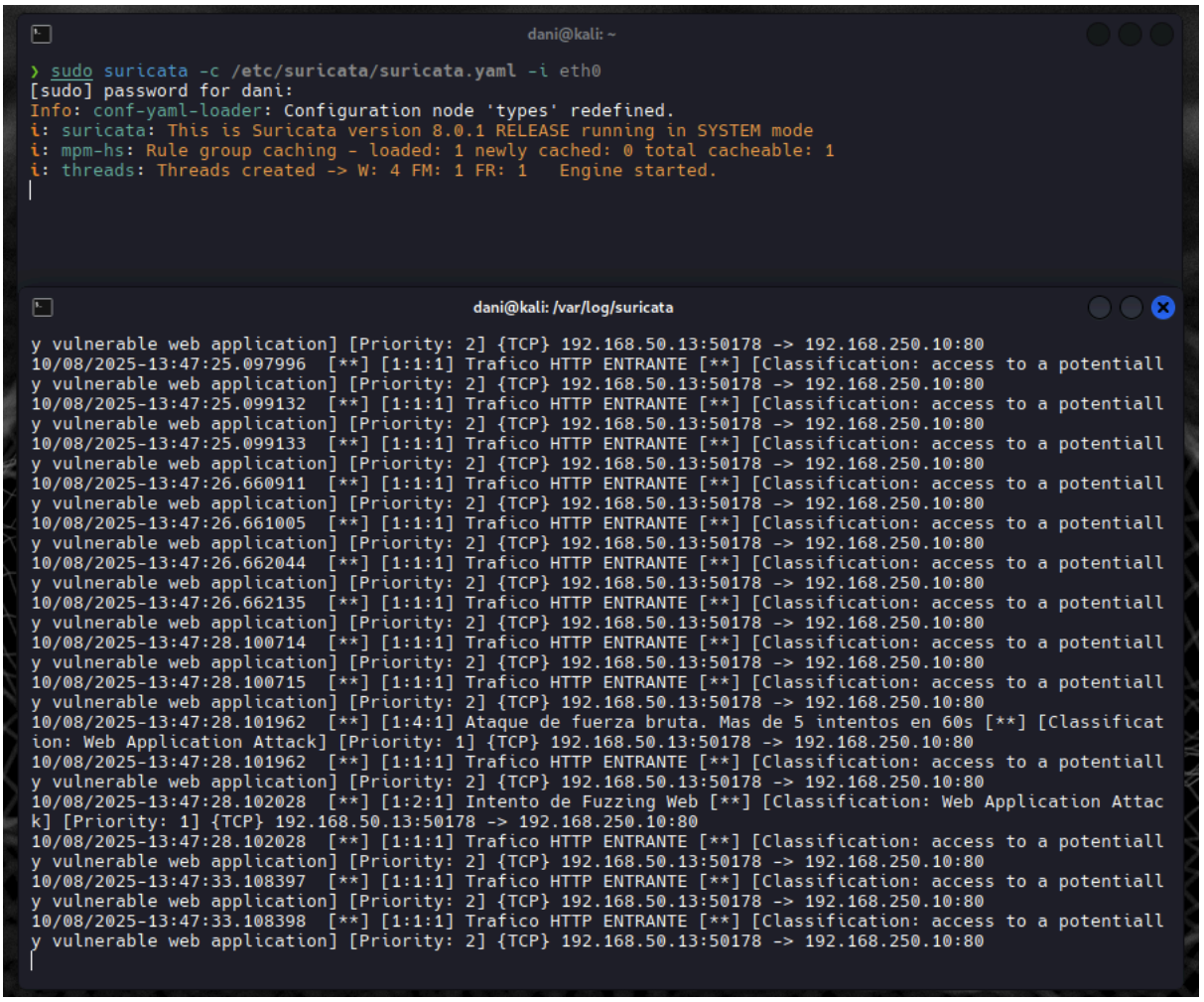
The main objective of this implementation is to detect possible attack attempts against the web service and record this activity for later analysis in the SIEM. Suricata uses a set of custom rules designed to cover different web attack scenarios and suspicious activity patterns.

Configured rules

A specific set of detection rules was created with the following criteria:

- **HTTP traffic:** an alert is triggered each time an HTTP connection to the Apache server is detected. This allows maintaining traceability of every incoming request.
- **Authentication attempts:** when a login event is detected on the web application, a variable is defined to act as a counter linked to the source IP address.
- **Brute-force detection:** if the counter reaches five consecutive failed login attempts, an alert is generated, indicating a possible brute-force or credential stuffing attack.
- **Fuzzing or scanning detection:** when an abnormal number of requests are detected within a short time period, Suricata triggers an alert indicating a

possible fuzzing or enumeration attempt against the service.



These rules are stored in the corresponding configuration file and managed through the policies defined in Kibana for centralized forwarding and visualization. The generated alerts are logged in **EVE JSON format**, allowing direct integration with the **Elastic Agent** for delivery to the ELK stack.

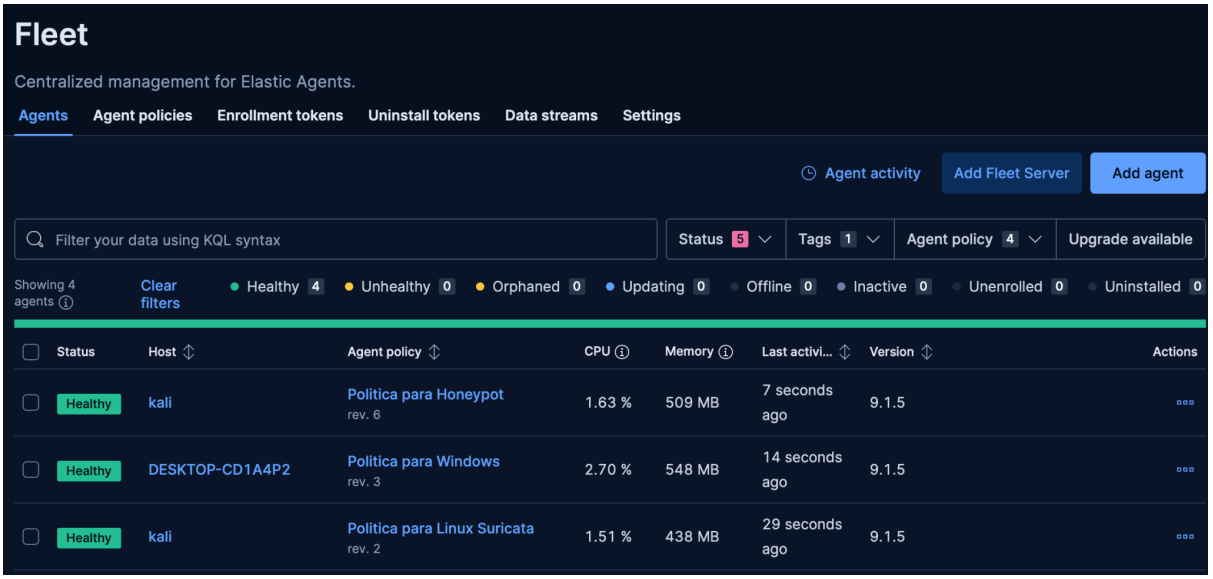
Expected results

Proper configuration of Suricata allows for:

- Continuous monitoring of HTTP traffic directed to the web server.
- Identification of intrusion attempts or anomalous behaviors in real time.
- Event correlation within the SIEM to detect broader attack patterns.
- Improved defensive capabilities of the environment through early threat detection.

Kibana

For centralized management and analysis of security events, **Kibana** was used as the main interface of the **SIEM based on the ELK Stack (Elastic Stack)**. From Kibana, all logs sent by the different **Elastic Agents** installed on the lab devices can be visualized, allowing analysts to correlate events, build dashboards, and proactively detect anomalies.



Agent policies and integrations

Three custom agent policies were created, each one tailored to the type of device and its specific role within the network. Every policy includes integrations designed to collect and forward the right set of logs for each system:

- **Suricata Policy:** collects and forwards alerts generated by the IDS deployed in DMZ2, including event fields such as source, destination, and detected signatures.
- **Windows Policy:** gathers logs from the Windows operating system (security, system events, PowerShell executions, etc.) located in the LAN subnet.
- **Honeypot Policy (FileStream):** captures executed commands and recorded activity from the honeypot.

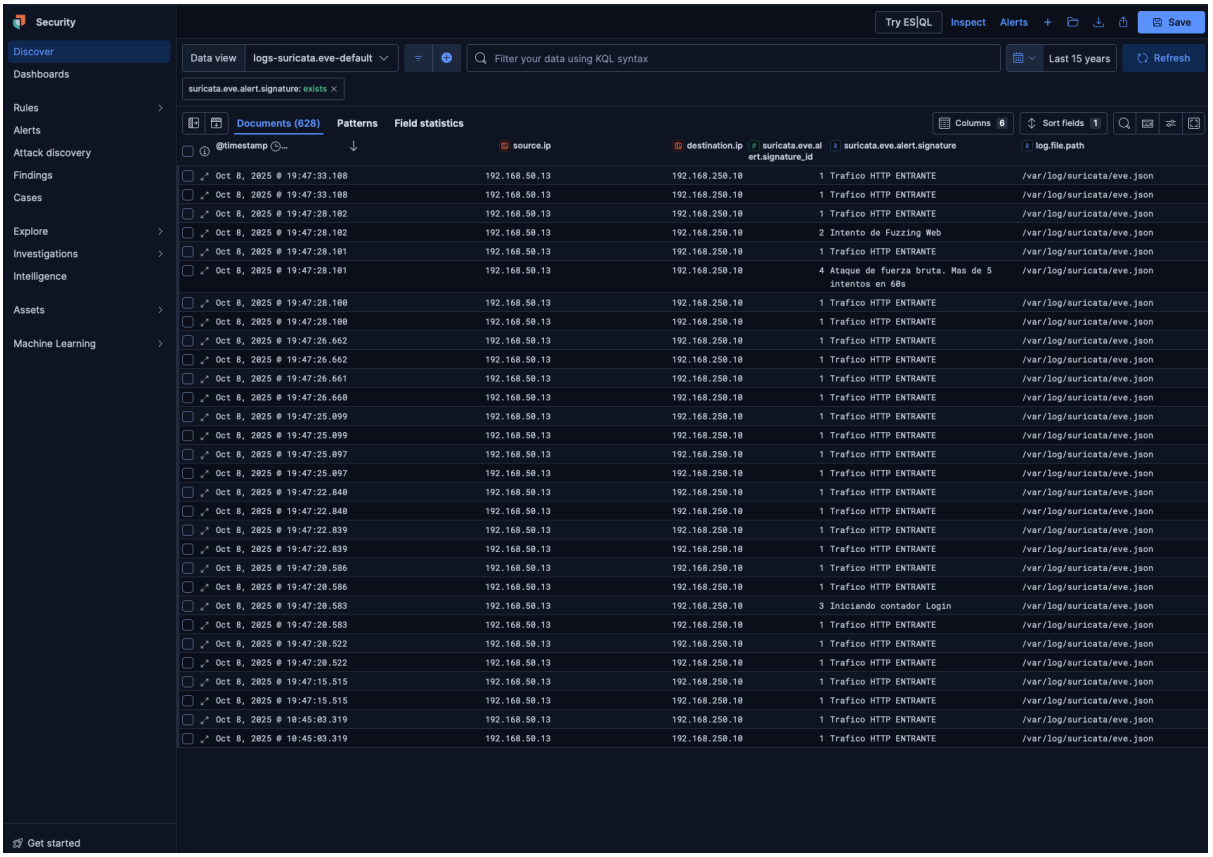
Once created, the policies were assigned to their respective agents. After installation, each agent started sending data continuously to Elasticsearch for indexing and correlation.

Data discovery and analysis

Using Kibana's **Discover** tab, an initial review of generated events was performed. Custom **data views** were configured to simplify log reading and analysis based on each agent's origin.

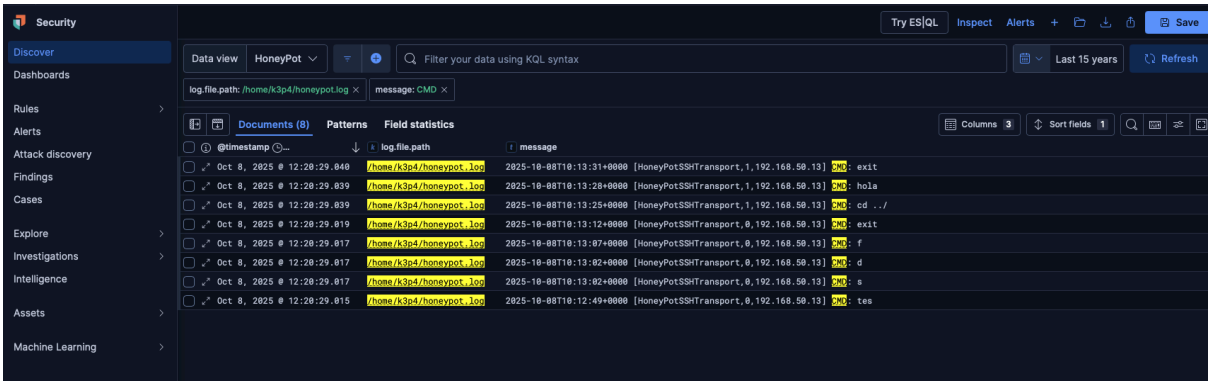
- **Suricata data view.** Includes the following fields:
  - `@timestamp`
  - `source.ip`
  - `destination.ip`
  - `suricata.eve.alert.signature_id`
  - `suricata.eve.alert.signature`
  - `log.file.path`

These fields allow identifying which alerts were triggered, the originating IP address, and the time they occurred. By filtering on `suricata.eve.alert.signature`, only actual Suricata detections are displayed.

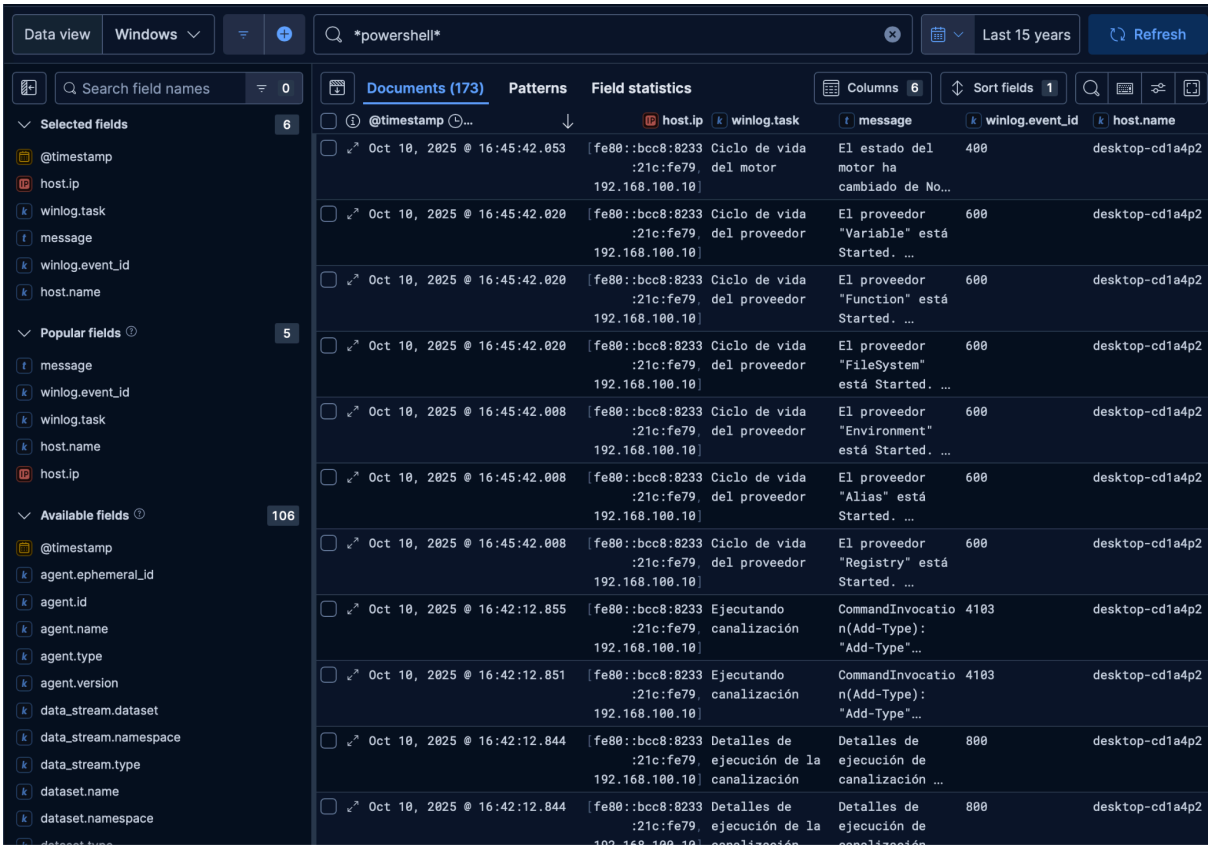


- **Honeypot data view.** Includes:
  - `@timestamp`
  - `log.file.path`
  - `message`

By filtering the keyword `CMD` in the *message* field, it is possible to see which commands were executed by attackers inside the honeypot, helping analyze their techniques and behavior.



- **Windows data view.** Allows tracking activities executed on the Windows machine within the LAN. In this example, events related to PowerShell were filtered to detect potentially suspicious or automated operations.

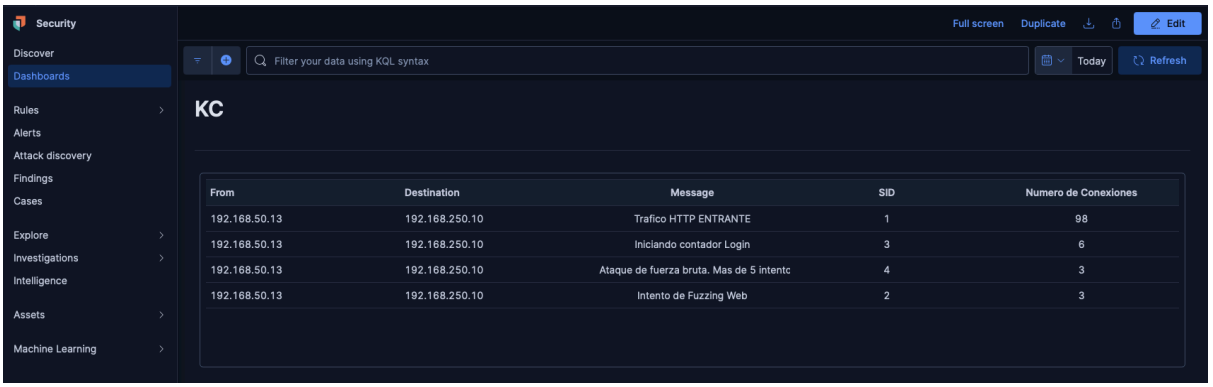


Dashboards and visualization

A **custom dashboard** was created in Kibana to visually represent the most relevant security events. Among the configured metrics are:

- The total count of Suricata-generated alerts.
- Top 10 source and destination IP addresses.
- Time-based evolution of events by detection signature.

These visualizations help correlate activities across subnets, identify attack patterns, and assess the effectiveness of both IDS and firewall rules.



Results and conclusions

Thanks to the integration of **Elastic** + **Suricata** + **pfSense**, the lab setup enables:

- Real-time detection and analysis of security incidents.
- Centralization of security information across all subnets.
- Observation and correlation of attacker interactions captured by the honeypot with IDS alerts.
- A global, unified view of the environment through dynamic dashboards.

Together, this deployment represents a **functional and educational Blue Team environment**, ideal for understanding how defensive components interact within a modern infrastructure.

This lab allowed me to gain a deeper understanding of how a **stateful firewall** operates and the importance of **rule hierarchy** for effective traffic control. It also helped me recognize the significance of properly configured security policies and how **event correlation between the firewall, IDS, and SIEM** is essential for effective incident detection and analysis within a corporate network.