
This is the **published version** of the bachelor thesis:

Sasot Martorell, Lluc. *Disseny i desplegament d'un Centre d'Operacions de Seguretat (SOC). Monitorització, detecció i resposta a incidents a la xarxa del DEIC de la UAB.* Treball de Final de Grau (Universitat Autònoma de Barcelona), 2026 (Enginyeria Informàtica)

This version is available at <https://ddd.uab.cat/record/334364>

under the terms of the  license.

Disseny i desplegament d'un Centre d'Operacions de Seguretat (SOC)

Monitorització, detecció i resposta a incidents a la xarxa del DEIC de la UAB

Lluc Sasot Martorell

30 de juny de 2026

Resum— Aquest treball dissenya, implementa, valida i desplega una solució SOC (*Security Operations Center*) de tres capes per a la xarxa del DEIC de la UAB, motivada per l'exposició creixent de les xarxes acadèmiques i per un precedent de *ransomware* a la universitat. Integra un NIDS (Suricata), un SIEM (Splunk) i un SOAR (Splunk SOAR) sobre una captura passiva per *port mirroring*, i es valida en un laboratori GNS3 amb la segmentació en VLAN del departament. Es detecten de manera completa dos escenaris d'atac: una execució remota de codi i una força bruta SSH. La latència de detecció és de l'ordre dels mil·lisegons. Les principals contribucions són la detecció del moviment lateral inter-VLAN, només visible en un punt d'agregació del trànsit entre segments, i una resposta automatitzada de dos nivells segons la gravetat. Com a prova pilot sobre maquinari real, ha revelat possibles exposicions crítiques de credencials i ha aportat visibilitat del trànsit. La solució ofereix, a més, un patró replicable i de baix impacte per a xarxes departamentals acadèmiques.

Paraules clau— NIDS, SIEM, SOAR, Suricata, Splunk, detecció d'intrusions, port mirroring, seguretat de xarxa, GNS3, SOC.

Abstract— This work designs, implements, validates and deploys a three-layer SOC (*Security Operations Center*) solution for the network of the DEIC at the UAB, motivated by the growing exposure of academic networks and by a past ransomware case at the university. It integrates a NIDS (Suricata), a SIEM (Splunk) and a SOAR (Splunk SOAR) on top of a passive port-mirroring capture, and is validated in a GNS3 laboratory with the department's VLAN segmentation. Two attack scenarios are detected fully: a remote code execution and an SSH brute-force. Detection latency is on the order of milliseconds. The main contributions are the detection of inter-VLAN lateral movement, only visible at a point that aggregates inter-segment traffic, and a two-tier automated response based on threat severity. As a pilot on real hardware, it revealed possible critical credential exposures and provided traffic visibility. The solution also offers a replicable, low-impact blueprint for academic departmental networks.

Keywords— NIDS, SIEM, SOAR, Suricata, Splunk, intrusion detection, port mirroring, network security, GNS3, SOC.



1 INTRODUCCIÓ

EN els darrers anys, les xarxes corporatives i acadèmiques han incrementat la seva exposició a incidents de seguretat a causa de l'alta connectivitat, la diversitat de dispositius i la coexistència de serveis crítics amb entorns oberts i de recerca. En aquest context, tenir visibilitat sobre el trànsit i capacitats de detecció i resposta

és clau per identificar incidents i reduir-ne l'impacte.

Un dels factors que ha reforçat aquesta necessitat és l'historial d'incidents de seguretat a la Universitat Autònoma de Barcelona (UAB), entre els quals destaca un atac de *ransomware*. En aquest cas, documentat el 2021, l'incident va paralitzar serveis interns i va obligar a desconnectar sistemes informàtics com a mesura de contenció.

Aquest mateix risc afecta la xarxa interna del Departament d'Enginyeria de la Informació i de les Comunicacions (DEIC), un dels departaments de la UAB. Aquesta xarxa està segmentada en laboratoris, despatxos i servidors, inter-

- E-mail de contacte: 1673590@uab.cat
- Menció: Tecnologies de la Informació
- Treball tutoritzat per: Sergi Robles Martínez (DEIC)
- Curs 2025/26

connectats a través del router central del departament, anomenat *Triki*, tal com il·lustra la Figura 2. Els laboratoris tenen commutació pròpia i un enllaç cap al departament, de manera que les pràctiques de l'estudiantat no afecten el funcionament general de la xarxa de l'Escola. El segment de despatxos concentra els equips del professorat i de suport a la docència i la recerca. Aquests equips, però, es troben en una VLAN (*Virtual Local Area Network*) integrada dins la infraestructura general de l'Escola, de manera que, tot i la seva ubicació física al departament, no pertanyen plenament a la xarxa gestionada pel DEIC. El segment de servidors allotja bona part dels serveis departamentals: suport a la docència, la recerca i les còpies de seguretat. Aquesta configuració dona lloc a un entorn heterogeni, amb requisits de seguretat diferents segons el segment, fet que dificulta el control homogeni i la detecció centralitzada d'incidències. Aquesta heterogeneïtat evidencia la necessitat de mecanismes de monitorització que identifiquin comportaments anòmals amb el mínim impacte sobre la infraestructura existent.

Aquest Treball de Fi de Grau (TFG) es planteja com una contribució aplicada a la seguretat de la xarxa del DEIC, centrada en la monitorització i l'anàlisi de trànsit per generar evidències útils en la detecció i gestió d'incidències.

El treball proposa la selecció, implementació i implantació d'una solució SOC (*Security Operations Center*) basada en tres components que conformen un cicle de detecció i resposta, tal com mostra la Figura 1. En primer lloc, un NIDS (*Network Intrusion Detection System*) per analitzar el trànsit de xarxa a partir d'una còpia obtinguda per *port mirroring* (SPAN, *Switched Port Analyzer*). En segon lloc, un SIEM (*Security Information and Event Management*) que rep els esdeveniments del NIDS a través d'un UF (*Universal Forwarder*), els centralitza i els explota mitjançant consultes i quadres de comandament (*dashboards*). Finalment, un SOAR (*Security Orchestration, Automation and Response*) que defineix respostes automatitzades en forma de *playbooks* per a casos d'ús prioritzats.

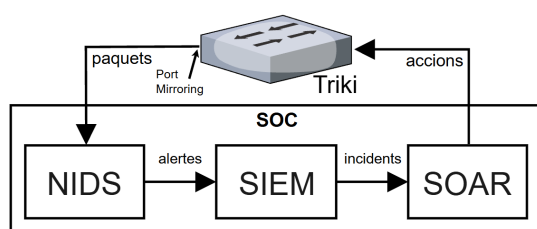


Fig. 1: Flux de la solució SOC: captura passiva (NIDS), centralització (SIEM) i resposta automatitzada (SOAR).

Per reduir el risc operatiu, la solució es valida prèviament en un laboratori emulat amb GNS3 (*Graphical Network Simulator-3*), cosa que permet provar la integració entre components, reproduir escenaris de trànsit controlats i ajustar configuracions abans d'afectar la infraestructura real.

La resta del document s'organitza així: objectius, metodologia i planificació, estat de l'art, desenvolupament de l'arquitectura SOC, proves i resultats, i conclusions amb possibles línies de continuació.

2 OBJECTIUS

L'objectiu general consisteix a desenvolupar una solució de monitorització, detecció i resposta per a la xarxa del DEIC i desplegar-la sobre la infraestructura real del departament. Aquesta solució ha de millorar la visibilitat del trànsit, l'anàlisi d'activitat sospitosa i la generació d'evidències útils per a la detecció i gestió d'incidències de seguretat. La construcció i la validació prèvia en un laboratori emulat són el mitjà per assolir aquest desplegament amb el mínim impacte sobre els serveis en producció. Aquest objectiu s'alinea amb l'Objectiu de Desenvolupament Sostenible 9 (indústria, innovació i infraestructura), ja que reforça la resiliència de la infraestructura digital del departament i hi incorpora capacitats de detecció i resposta.

2.1 Objectius específics

A partir de l'objectiu general, es defineixen objectius específics que divideixen el projecte en blocs de treball concrets i verificables, prioritzats i ordenats segons les dependències tècniques:

- **O1.** Analitzar l'arquitectura de la xarxa del DEIC.
- **O2.** Construir un laboratori emulat amb GNS3, representatiu de la xarxa del DEIC, que serveixi d'entorn segur per desplegar-hi i validar el SOC abans del desplegament real.
- **O3.** Desplegar i configurar un sensor NIDS capaç de generar alertes i esdeveniments estructurats a partir del trànsit observat, aplicant un conjunt de regles de detecció predefinides.
- **O4.** Integrar els esdeveniments del NIDS en un SIEM per centralitzar-ne la ingesta, consulta i visualització mitjançant quadres de comandament.
- **O5.** Definir i implementar respostes automatitzades amb SOAR que, davant alertes del SIEM, executin *playbooks* de contenció per als casos d'ús prioritzats.
- **O6.** Validar la traçabilitat *end-to-end* entre captura, detecció, ingestió, consulta i resposta, mitjançant escenaris d'atac repetibles que exercitin tota la cadena del SOC.
- **O7.** Validar els recursos, les llicències i els requisits necessaris per al desplegament a la xarxa real.
- **O8.** Desplegar la solució a la xarxa real del DEIC en mode passiu i avaluar-ne el comportament operatiu en termes d'estabilitat, volum d'esdeveniments i consum de recursos.

Aquests objectius orienten les decisions de disseny i implementació, descrites a la metodologia següent.

3 METODOLOGIA I PLANIFICACIÓ

Aquest capítol presenta la metodologia seguida per construir el SOC i la planificació temporal del projecte. La naturalesa iterativa del treball i la necessitat de validar cada component en entorn controlat abans del desplegament real fan del model en espiral [1] la metodologia més adequada: permet validar i integrar cada component de manera iterativa i reduir el risc tècnic a cada pas.

3.1 Principis metodològics

El model en espiral es selecciona enfront d'alternatives seqüencials com el model en cascada. Mentre que la cascada exigeix tancar cada fase abans de començar la següent,

l'espiral avança per cicles successius d'anàlisi, disseny, implementació i proves, i cada cicle lliura una versió funcional del sistema amb un increment controlat de complexitat.

Aquesta tria és especialment rellevant en seguretat de xarxa, on petites modificacions de configuració poden alterar el comportament del sistema (generació massiva d'alertes, sobrecàrrega del sistema de logs), efectes que la validació de cada cicle en entorn emulat permet detectar abans del desplegament real. Es segueixen quatre principis:

- **Validació incremental:** el sistema es valida a cada cicle, a mesura que s'hi afegeixen i s'integren nous components.
- **Reducció de risc:** les configuracions crítiques (SPAN, regles NIDS, ingestió SIEM) es validen en entorn emulat abans de producció.
- **Traçabilitat tècnica:** cada cicle genera evidències verificables (logs, alertes, quadres de comandament).
- **Evolució controlada:** el sistema evoluciona progressivament des d'una arquitectura inicial fins a una plataforma SOC completa.

3.2 Planificació per cicles

El projecte s'estructura en cicles seqüencials entre el 16/02/2026 i el 28/06/2026, cadascun amb fases d'anàlisi, disseny, implementació i proves, i un increment progressiu de funcionalitats. La Taula 1 en resumeix el focus, el calendari i els increments principals.

4 ESTAT DE L'ART

La monitorització de la seguretat en xarxa compta amb un conjunt consolidat d'eines i marcs de referència, articulats al voltant del concepte de SOC: un model organitzatiu que combina persones, processos i tecnologia per detectar, analitzar i respondre a incidents de manera centralitzada [2]. Tecnològicament, aquest model es construeix sobre tres famílies d'eines complementàries: el NIDS com a capa de detecció, el SIEM com a capa de centralització i anàlisi, i el SOAR com a capa de resposta automatitzada. Aquest apartat revisa les alternatives més representatives de cada família, els entorns de validació previs al desplegament i un cas d'aplicació universitari, per fonamentar la selecció d'eines adoptada.

4.1 Sistemes de detecció d'intrusions

Els sistemes NIDS constitueixen la primera capa de monitorització, encarregada d'analitzar el trànsit de xarxa per identificar patrons coneguts d'atac o comportaments anòmals

[3]. En la detecció basada en signatures, la referència històrica és Snort [4], que va popularitzar el llenguatge de regles convertit avui en estàndard *de facto* de la indústria. Suricata [5] manté la compatibilitat amb aquest llenguatge, però hi afegeix una arquitectura *multithread* i una sortida estructurada en JSON (*eve.json*) orientada a la integració amb sistemes externs. L'estudi comparatiu de White et al. [6] mostra que, gràcies a aquest paral·lelisme, una sola instància de Suricata supera Snort fins i tot en configuracions multiinstància. Amb independència del motor, aquests sensors es despleguen habitualment en mode passiu mitjançant *port mirroring* (SPAN), sense interferir en la comunicació original.

4.2 Plataformes SIEM

Els sistemes SIEM centralitzen els esdeveniments dels sensors i correlacionen informació de múltiples fonts mitjançant consultes, agregacions i quadres de comandament, convertint grans volums de logs en informació accionable. Entre les solucions més implantades destaquen Splunk, plataforma comercial amb gran capacitat d'ingesta i consulta, i Elastic Stack (ELK), l'alternativa *open source* formada per Elasticsearch, Logstash i Kibana [7].

4.3 Sistemes SOAR

Els sistemes SOAR afegeixen una capa d'automatització sobre el SIEM: davant d'una alerta determinada, executen fluxos de resposta predefinits (*playbooks*) que redueixen el temps de reacció i estandarditzen els processos operatius del SOC [2]. Dues solucions representatives són Splunk SOAR, integrat de manera nativa amb l'ecosistema Splunk [8], i Cortex XSOAR [9], orientat a l'orquestració de processos amb eines de tercers.

4.4 Validació en entorns emulats i casos d'aplicació

Abans del desplegament en producció, és habitual validar configuracions i integracions en laboratoris emulats. Eines com GNS3 [10] permeten reproduir topologies de xarxa i escenaris de trànsit controlats, fet que facilita provar els mecanismes de captació, les regles de detecció i el flux complet d'esdeveniments sense afectar la infraestructura real.

Pel que fa a l'àmbit universitari, un cas representatiu és el de la Louisiana State University, que opera un SOC basat en Splunk SIEM i Splunk SOAR on els estudiants participen en tasques reals de monitorització i resposta a incidents [11], fet que mostra la viabilitat d'arquitectures SOC amb tecnologies estàndard de la indústria en entorns acadèmics

TAULA 1: PLANIFICACIÓ PER CICLES DEL PROJECTE

Cicle	Focus	Període	Increment principal
Cicle 0	Arquitectura i selecció d'eines	16/02–02/03	Anàlisi de la xarxa del DEIC, definició de l'arquitectura del SOC i dels fluxos entre NIDS, SIEM i SOAR, selecció de les tecnologies i preparació de l'entorn de laboratori GNS3.
Cicle 1	Components bàsics	03/03–12/04	Desplegament del NIDS i el SIEM al laboratori, configuració de la ingestió inicial i validació de la connectivitat i del flux bàsic d'esdeveniments <i>end-to-end</i> .
Cicle 2	Detecció i alertes	13/04–26/04	Definició i ajust de les regles de detecció del NIDS, estructura d'alertes i quadres de comandament al SIEM, i validació de la cobertura dels escenaris definits.
Cicle 3	Resposta automàtica	27/04–03/06	Disseny i configuració dels <i>playbooks</i> de resposta al SOAR, integració amb el SIEM i validació de les respostes automatitzades davant esdeveniments simulats.
Cicle 4	Desplegament real	04/06–28/06	Migració de la solució validada cap a la infraestructura real del departament, desplegament en mode passiu i validació del <i>pipeline</i> complet amb trànsit de producció.

reals. Cada una d'aquestes capes compta amb solucions provades per separat. Els treballs publicats, però, es concentren en desplegaments perimetral o de gran escala i amb prou feines documenten la integració completa de NIDS, SIEM i SOAR per detectar el moviment lateral inter-VLAN (el desplaçament d'un atacant entre segments interns un cop superat el perímetre) en una xarxa departamental, sovint sense caracteritzar-la quantitativament. Aquest treball cobreix aquesta manca amb una arquitectura SOC reproducible, centrada en la visibilitat interna entre segments i validada sobre un laboratori que replica la segmentació real del DEIC.

5 DESENVOLUPAMENT

Es descriu el desenvolupament de la solució seguint l'ordre d'implantació: anàlisi de la xarxa, laboratori GNS3, integració dels components del SOC i desplegament real.

5.1 Anàlisi de la xarxa

La Figura 2 presenta l'arquitectura de la xarxa del DEIC, amb els principals segments, els fluxos de trànsit i els punts de monitorització identificats en l'estudi inicial. Donada la criticitat dels serveis i la voluntat de no impactar la infraestructura existent, s'ha descartat un desplegament *inline* del NIDS i s'ha optat per un enfocament passiu basat en *port mirroring*. Aquest mecanisme obté una còpia del trànsit sense interferir en la comunicació original, i fa que una eventual aturada del sensor no afecti la disponibilitat dels serveis monitoritzats.

El *Triki* concentra el volum més elevat de trànsit i s'ha seleccionat com a punt de captura únic. Concentrar la captura en un sol node de distribució, en lloc de desplegar múltiples sensors, redueix la complexitat operativa i el cost computacional, ja que un únic port de sortida dedicat replica cap al sensor el trànsit entrant i sortint dels segments rellevants. La decisió respon a un compromís entre cobertura i rendiment: agrega el trànsit inter-VLAN més significatiu per als casos d'ús previstos sense saturar el NIDS. Amb la identificació dels segments, els fluxos i la selecció del punt de captura òptim, queda assolit l'objectiu O1.

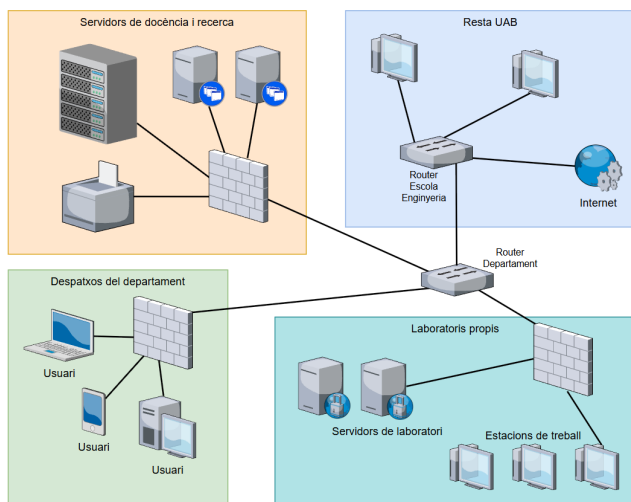


Fig. 2: Arquitectura de xarxa del DEIC i punts de monitorització

5.2 Anàlisi de requisits

De l'anàlisi de la xarxa i dels objectius del projecte es deriven els requisits que la solució ha de satisfer. Com a requisits funcionals, el SOC ha de capturar el trànsit de manera passiva, detectar patrons d'atac i generar alertes estructurades, centralitzar-les i consultar-les, i automatitzar respostes de contenció per als casos d'ús prioritzats. Com a requisits no funcionals, ha de tenir un impacte nul sobre els serveis en producció (captura passiva i tolerant a l'aturada del sensor), aïllar el pla de gestió de les xarxes de dades, contenir el cost reaprofitant maquinari existent i components de codi obert o amb llicència acadèmica, i ser reproducible, validant cada component en un entorn emulat abans del desplegament real.

Un cop tancada l'anàlisi, s'ha procedit al muntatge de la solució. Seguint la metodologia en espiral, la implementació s'ha realitzat primer en un entorn emulat amb GNS3, on s'han validat la topologia, el mecanisme de captació i la integració dels components, abans de plantejar-ne el desplegament sobre la xarxa real.

5.3 Creació de l'entorn simulat a GNS3

S'ha implementat un laboratori emulat que reproduïx de manera simplificada però representativa la xarxa del DEIC (Figura 8, a l'apèndix), per validar el *pipeline* complet (captura, detecció, ingestió, visualització i resposta) abans de qualsevol intervenció en producció. L'entorn permet executar escenaris d'atac de manera repetible i ajustar configuracions sense risc operatiu.

5.3.1 Desplegament del laboratori

El laboratori s'ha desplegat sobre un únic equip amfitrió amb tecnologies de virtualització heterogènies, escollides en cada cas segons els requisits del component: les solucions més lleugeres allà on n'hi ha prou i la virtualització completa només per als sistemes que ho requereixen. La Taula 4, a l'apèndix, resumeix on s'executa cada element.

L'entorn s'ha consolidat sobre un únic ordinador situat al DEIC amb el sistema operatiu Ubuntu 24.04, accessible des de la VPN (*Virtual Private Network*) WireGuard [12], on s'executa el servei `gns3-server` com a *daemon* `systemd`. Tots els nodes s'instancien sobre aquesta màquina mitjançant KVM (*Kernel-based Virtual Machine*) natiu, sense virtualització imbricada. La GUI (*Graphical User Interface*) de GNS3 s'executa al PC de l'autor i s'hi connecta com a *remote main server*, de manera que el laboratori es manté operatiu de forma contínua, independentment de l'estat d'aquell PC.

Quant als rols dins del laboratori, *Sergi* allotja el servei Marimo, que és l'objectiu dels escenaris d'atac, *LabPC Estudiants* reproduceix l'equip de laboratori des del qual s'executen els *exploits*, i *SPAN-TAP-Triki* implementa el *port mirroring* sobre Open vSwitch (OVS).

5.3.2 Implementació de la topologia

La topologia s'ha estructurat com una arquitectura jeràrquica de nivell 3 que segmenta el trànsit del DEIC en quatre VLAN funcionals i el connecta amb la xarxa de l'Escola i Internet mitjançant una cadena de tres routers VyOS. Per a més detalls, vegeu la Figura 8 a l'apèndix. El disseny reproduceix la separació de plans habitual en xarxes departamentals universitàries i permet aplicar polítiques de

filtrat i reescriptura d'adreces (DNAT/SNAT, *Destination/Source Network Address Translation*) sense modificar la pila de monitorització. La Taula 2 resumeix la segmentació amb els rangs reals del DEIC.

TAULA 2: SEGMENTACIÓ EN VLAN A GNS3

Segment	VLAN	Subxarxa
Despatxos	20	158.109.79.128/25
Servidors	30	158.109.79.0/26
Laboratoris	40	158.109.79.64/26
Gestió (SOC)	99	192.168.99.0/24

Sobre aquesta segmentació s'han distribuït els nodes en les seves VLAN, i destaca la VLAN 99 de gestió, que aïlla els tres components del SOC (*Suricata*, *Splunk SIEM* i *Splunk SOAR*) en un pla de control privat. Les adreces concretes es recullen a la Taula 5, a l'apèndix.

El node *SPAN-TAP-Triki* (OVS) actua com a punt únic d'agregació: rep les quatre VLAN des de *RouterDepartament-Triki*, les reparteix cap als segments d'accés i conserva el port de *mirrortap* cap al sensor. La cadena de routers VyOS connecta el DEIC amb l'Escola i Internet, i *RouterEscolaEnginyeria* aplica Proxy ARP (*Proxy Address Resolution Protocol*) per integrar el rang 158.109.79.0/24 en l'adreçament heretat de l'Escola. Aquest ús de Proxy ARP conserva l'adreçament original del DEIC sense renumerar la xarxa real, fet clau per a un eventual desplegament en producció.

5.3.3 Configuració del port mirroring

Sobre el node *SPAN-TAP-Triki* s'ha configurat un *mirrortap* d'OVS que replica el trànsit dels ports de les VLAN 20, 30, 40 i 99 cap a un port de sortida dedicat (*eth2*), on es connecta la interfície del NIDS (*ens4*, en mode promiscu). La rèplica inclou paquets entrants i sortints de cada port, de manera que el NIDS disposa d'una visió bidireccional de tots els segments. El *mirrortap* persisteix entre reinicis gràcies a la base de dades *conf.db* d'OVS, dins del sistema de fitxers persistent del contenidor. Així es completa el mecanisme de captació passiu, sense impacte sobre la infraestructura monitoritzada. Amb el laboratori emulat desplegat, queda assolit l'objectiu O2: l'entorn de proves està a punt per desplegar-hi el SOC, i es dona per completat el Cicle 0.

5.3.4 Integració del NIDS

Un cop validada la connectivitat i el funcionament del *mirrortap*, s'ha desplegat *Suricata* com a sensor NIDS connectat a la interfície de captura. La selecció de *Suricata*, enfront de *Snort* i *Zeek*, respon al suport natiu de sortida estructurada (*eve.json*), a la compatibilitat amb el llenguatge de regles estàndard de la indústria i a una arquitectura *multithread* que li permet escalar a trànsit d'alta velocitat [5, 6].

La configuració principal s'ha realitzat sobre el fitxer *suricata.yaml*, on s'ha definit la interfície d'entrada en mode promiscu, la variable *HOME_NET* amb la xarxa del DEIC (158.109.79.0/24) i el conjunt de regles personalitzat *my.rules*.

La Figura 3 recull les tres regles del conjunt *my.rules*. La primera (SID 1000001), definida per a la validació inicial del sensor, detecta connexions TCP entrants al port 80 del servidor de docència (158.109.79.10), on s'executa un servei HTTP. Aquesta regla, deliberadament trivial, només pretén confirmar que *Suricata* processa correctament

el trànsit capturat per la rèplica i genera alertes estructurades davant d'activitat real sobre la xarxa. Les altres dues (SID 1000002 i 1000003) corresponen als escenaris d'atac, descrits a la secció de resultats.

```

alert tcp any any -> 158.109.79.10 80 (msg:"DEIC batec
del pipeline (TCP SYN cap a ServidorDocencia:80)";
flags:S; sid:1000001; rev:1; classtype:web-
application-activity;)
alert http any any -> $HOME_NET any (msg:"CVE-2026-39987
Marimo RCE via WebSocket /terminal/ws sense
autenticar"; flow:established,to_server; http.
method; content:"GET"; http.uri; content:"/terminal
/ws"; http.header; content:"Upgrade"; nocase;
content:"websocket"; nocase; sid:1000002; rev:2;
classtype:web-application-attack;)
alert tcp any any -> [158.109.79.0/24] 22 (msg:"DEIC for
ça bruta SSH per llindar (tota la xarxa interna)";
flags:S; threshold:type both,track by_src,count 5,
seconds 60; sid:1000003; rev:1; classtype:attempted
-admin;)

```

Fig. 3: Conjunt de regles personalitzat *my.rules* (SID 1000001, 1000002 i 1000003).

Les alertes es registren en dos formats: *fast.log*, lleigible per humans per a la inspecció ràpida durant el desenvolupament de regles, i *eve.json*, estructurat en JSON i font primària d'ingestió cap al SIEM (exemples a l'apèndix, Figura 12 i Figura 13).

El desplegament de *Suricata* valida l'objectiu O3: el sensor genera alertes estructurades a partir del trànsit observat, en format lleigible (*fast.log*) i en format integrable amb sistemes externs (*eve.json*).

5.3.5 Integració del SIEM

A continuació, s'ha integrat *Splunk Enterprise 10.2.3* com a SIEM centralitzat sobre una màquina dedicada dins la VLAN 99 de gestió (192.168.99.200). La ingestió es fa a partir del fitxer *eve.json* generat per *Suricata*, mitjançant un agent lleuger *Splunk Universal Forwarder* [13] instal·lat sobre la màquina del NIDS (192.168.99.201), que reenvia les noves entrades a l'*indexer* de *Splunk* en temps gairebé real.

La selecció de *Splunk* com a SIEM, enfront de l'alternativa oberta Elastic Stack (ELK), respon a dos criteris. En primer lloc, la integració nativa amb *Splunk SOAR* comunica l'alerta amb el *playbook* de resposta sense desenvolupament addicional, mentre que ELK no inclou capa SOAR. En segon lloc, autogestionar Elasticsearch exigeix experiència de manteniment del clúster, un cost operatiu alt per a un equip reduït, que *Splunk* evita amb una ingesta i un llenguatge de consulta (SPL, *Search Processing Language*) llestos per a l'ús. Per a un SOC universitari, malgrat el menor cost per volum d'ELK, les llicències acadèmiques i el valor formatiu d'una eina estàndard inclinen la balança, com mostra el SOC de la Louisiana State University [11].

La transmissió usa el protocol *Splunk-to-Splunk (S2S)* sobre una connexió TCP persistent al port 9997, que desacobla la generació de logs de la ingestió: *Suricata* continua operant encara que el SIEM no estigui disponible.

Un cop ingerits, els esdeveniments es normalitzen i s'indexen en un índex dedicat (*suricata*), de manera que els camps rellevants (adreces IP d'origen i destinació, i protocol, entre d'altres) queden disponibles per a consultes posteriors mitjançant el llenguatge SPL. La Figura 15 de l'apèndix en mostra un exemple.

A partir d'aquesta informació s'han construït quadres de comandament operatius amb tres vistes: volum d'alertes per franja temporal, adreces IP amb més activitat sospitosa, i distribució d'alertes per protocol i regla disparada.

Per verificar la integració completa, s'ha generat trànsit HTTP des de l'estació *Sergi* cap al *Servidor Docència* (port 80), activant la regla `SID 1000001` (Figura 3).

L'alerta generada per Suricata es va reflectir correctament a l'índex *suricata* del SIEM en temps gairebé real i va quedar visible al *dashboard* operatiu, confirmant la traçabilitat completa i donant per assolit l'objectiu O4. La Figura 9 de l'apèndix en mostra un fragment. Amb el NIDS i el SIEM connectats i el flux bàsic d'esdeveniments validat *end-to-end*, es completa el Cicle 1.

5.3.6 Integració del SOAR

La darrera capa del SOC és el SOAR, encarregat d'automatitzar la resposta davant les alertes centralitzades pel SIEM. Tot i que l'automatització correspon a un cicle posterior, la plataforma s'integra aquí com a tercer component, i la seva lògica de resposta s'exercita més endavant amb els escenaris d'atac. S'ha desplegat Splunk SOAR On-Prem 8.5.0, escollit per la seva integració nativa amb Splunk (enfront d'alternatives com Cortex XSOAR), sobre una màquina Rocky Linux 9 dins la VLAN 99 de gestió (192.168.99.202), compartint el pla de control aïllat amb el NIDS i el SIEM [8].

La ingestió d'esdeveniments es fa per consulta periòdica a Splunk: el SOAR llança cerques filtrades per identificador de regla (SID) i, per cada alerta nova, crea un *container* amb els camps rellevants normalitzats en format CEF (*Common Event Format*), com ara les adreces IP d'origen i de destinació. Aquests camps són els que els *playbooks* fan servir per decidir i executar la resposta.

La resposta s'aplica fora de banda. El SOAR obre una connexió SSH cap a *RouterDepartament-Triki* a través de la VLAN 99, sobre la qual el servei SSH del router escolta de manera exclusiva, sense exposar-se a les VLAN de dades ni al WAN (*Wide Area Network*), preservant l'aïllament del pla de gestió. Sobre aquesta connexió, el *playbook* aplica una regla de *firewall* VyOS que descarta el trànsit de l'adreça origen. Cada bloqueig és una regla independent per IP, amb un número de regla *N* determinista a partir de l'últim octet de l'adreça ($N = 100 + \text{últim_octet}$), cosa que la fa idempotent: re-bloquejar una mateixa IP reescriu la mateixa regla sense duplicar-la, i la justificació del bloqueig (SID, IP i marca de temps) queda registrada al propi router.

Sobre aquesta base s'han definit dos nivells de resposta segons la gravetat de l'amenaça: un bloqueig permanent amb notificació per correu, que requereix intervenció humana per alliberar-lo, i un bloqueig temporal silenciós que s'allibera automàticament en exhaurir-se un temporitzador. La concreció d'aquests *playbooks* per a cada cas d'ús prioritzat es descriu a la secció d'escenaris d'atac.

Amb el desplegament integrat del NIDS, el SIEM i el SOAR sobre la VLAN de gestió, queda completa l'arquitectura del SOC presentada a la Figura 1, amb la cadena de detecció i resposta implementada *end-to-end*:

Trànsit → *mirror* OVS → Suricata → Splunk UF →
Splunk SIEM → Splunk SOAR

6 PRESENTACIÓ I DISCUSSIÓ DE RESULTATS

Aquesta secció presenta i discuteix els resultats de la validació en l'entorn emulat i en producció.

6.1 Escenaris d'atac avaluats al simulador

La validació es recolza en dos escenaris d'atac que reproduïen vectors de risc concrets per al DEIC i exerciten les dues capes del SOC, detecció i resposta, amb mecanismes complementaris. El primer, una execució remota de codi (RCE) sobre el servei Marimo, prova la detecció per coincidència de contingut i una resposta de bloqueig permanent amb notificació. El segon, una força bruta SSH per túnel VPN, exercita la detecció per llinard de freqüència i un bloqueig temporal i silenciós. Per a cada escenari es descriuen el model d'amenaça, la cadena d'atac i les evidències al SIEM (Figura 9). La Taula 3 en resumeix la comparativa.

6.1.1 Escenari 1: Exploit CVE-2026-39987 Marimo RCE

El model d'amenaça simula una persona aliena a la UAB que entra físicament en un laboratori de pràctiques i executa un *exploit* des d'un equip d'estudiant. La IP origen (158.109.79.66, VLAN 40) és una adreça UAB legítima que cap control perimetral marcaria com a sospitosa, i el trànsit entre VLAN no és visible des del perímetre. Només el NIDS situat sobre el *mirror* inter-VLAN pot detectar-lo, fet que justifica empíricament l'arquitectura adoptada davant del moviment lateral intern.

La vulnerabilitat explotada, descoberta recentment, és la **CVE-2026-39987**, una execució remota de codi pre-autenticació al servidor Marimo $\leq 0.20.4$ [14], instal·lat a la màquina *Sergi* (158.109.79.150, VLAN 20). El servei escolta sense autenticació i exposa un *shell* per WebSocket a l'*endpoint* `/terminal/ws`, de manera que l'explotació és una petició HTTP en text pla, completament visible per a Suricata. La regla de detecció (`SID 1000002`) hi casa per tres condicions de contingut, el mètode GET, l'URI `/terminal/ws` (exclusiva de Marimo) i la capçalera `Upgrade: websocket`, amb una taxa de falsos positius pràcticament nul·la.

La resposta automatitzada és un bloqueig permanent amb notificació. El *playbook* de SOAR aplica una regla de descart sobre la IP origen a *RouterDepartament-Triki* i envia un correu amb el detall de l'incident i la comanda per revertir-lo. La Figura 14 de l'apèndix en mostra l'estructura. El bloqueig no s'allibera sol, ja que un RCE no autenticat sobre una estació del professorat exigeix confirmació humana abans de restablir l'accés. En cada intent d'explotació, la regla `SID 1000002` s'ha activat de manera repetible i tota la cadena (detecció, ingestió, bloqueig permanent i correu) s'ha exercitat correctament.

6.1.2 Escenari 2: Força bruta SSH

L'atac simula un adversari extern que accedeix a la xarxa interna per un túnel VPN legítim: una màquina Kali (203.0.113.50) s'autentica com a *pe-per* de WireGuard [12] contra el concentrador *Heimdall* (158.109.79.130), que aplica *masquerading* de manera que els intents d'autenticació SSH amb *hydra* apareixen des de la seva adreça interna. Com que SSH és xifrat, la detecció es basa en freqüència: la regla (`SID 1000003`)

alerta quan un mateix origen supera cinc intents en seixanta segons [15], i `track by_src` detecta també escombrats horitzontals. La resposta és un bloqueig temporal silenciós (sense correu): la força bruta és soroll d'alta freqüència i notificar-lo provocaria fatiga d'alertes, a més de compartir origen amb usuaris legítims de la VPN. El descart s'allibera automàticament via temporitzador. Aquest *playbook* assoleix l'objectiu O5.

Tots dos escenaris s'han validat *end-to-end* i de manera repetible, cobrint la traçabilitat completa entre captura, detecció, ingestió, consulta i resposta, fet que assoleix l'objectiu O6. Amb la generació d'alertes davant els dos vectors i la resposta automatitzada de tots dos escenaris, es completen el Cicle 2, de detecció, i el Cicle 3, d'automatització de respostes.

TAULA 3: COMPARATIVA DELS DOS ESCENARIS D'ATAC AVALUATS

	Escenari 1	Escenari 2
Vector	RCE pre-auth (Marimo)	Força bruta SSH
Origen	<i>Insider</i> físic (VLAN 40)	Túnel VPN
Detecció	Coincidència de contingut	Llindar de freqüència
Regla	SID 1000002	SID 1000003
Resposta	Bloqueig permanent + correu	Bloqueig temporal automàtic

6.2 PoC: Desplegament a la xarxa real

Els resultats anteriors validen la solució com a prova de concepte en l'entorn emulat: la cadena de detecció, centralització i resposta funciona de manera repetible i amb un cost de recursos baix. Sobre aquesta base s'ha dut a terme el desplegament de l'objectiu O8 (Cicle 4) sobre maquinari real del DEIC, partint d'un risc reduït. El pressupost de recursos, llicències i punts de connexió ja s'havia validat al laboratori (objectiu O7), i la VLAN de gestió actua com a frontera de portabilitat que minimitza els canvis entre laboratori i producció.

El SOC s'ha desplegat en tres màquines virtuals sobre un únic amfitrió físic del departament (158.109.79.21, Debian 13), connectat a un *mirror* de la VLAN de despatxos (Figura 2) i operant només en detecció per descartar qualsevol impacte sobre els serveis. El sensor de producció carrega el conjunt de regles obert ET Open [16], amb més de 50.000 signatures.

L'operació sobre trànsit real ha requerit un ajust: les signatures informatives de reensamblat TCP generen un volum molt elevat d'esdeveniments sobre fluxos capturats de manera asimètrica al *mirror*. S'han silenciat de manera selectiva perquè no emmascarin les alertes d'intrusió al *dashboard* de producció, i aquest ajust mostra la diferència entre el laboratori, amb un conjunt mínim de regles, i la xarxa real, amb el conjunt complet. La Figura 10 de l'apèndix en mostra la versió de producció.

L'anàlisi del trànsit real ha revelat, a més, un desequilibri entre severitat i soroll que fonamenta la política de resposta del SOAR. La severitat d'una alerta reflecteix la prioritat del *classtype* de la regla que la dispara, no la certesa que s'hagi produït un atac. Sota aquest criteri, en un dia típic de producció es compten 618 alertes de prioritat alta (severitat 1) i 295 de prioritat mitjana (severitat 2), davant de 23.653 de prioritat baixa (severitat 3): el 96,3% del volum és soroll informatiu, com actualitzacions de paquets (`apt`), API de serveis SaaS o trànsit QUIC xifrat. Aquestes xifres s'han ob-

tingut amb una consulta al SIEM de producció que agrupa per severitat les alertes d'un matí laborable, la franja de més activitat de la xarxa del DEIC. El resultat d'aquesta consulta es mostra a la Figura 11 de l'apèndix. Aquest desequilibri justifica una resposta diferenciada segons la gravetat: de moment, com que el desplegament és encara prematur, no s'apliquen modificacions automàtiques al *firewall* de *Triki* sobre trànsit de producció, sinó que el SOAR només envia un correu a través del *relay* de correu intern del departament (*skynet*), al qual es connecta des d'un domini propi autoritzat per entregar correu als comptes `@uab.cat`, gestionats amb Outlook. La condició de notificació creua severitat i *classtype* precisament perquè una prioritat alta no implica un atac real: en producció, el 100% de les alertes de severitat 1 corresponen a *policy-violation*, de manera que el filtre exclou aquest *classtype* i deixa només les categories d'atac veritable. Aquesta notificació s'ha aplicat de moment només a l'escenari de Marimo (SID 1000002) per validar-ne el funcionament, però el mecanisme està preparat per estendre's a qualsevol alerta de severitat 1 que no sigui *policy-violation*. La modificació automàtica del *firewall* sobre trànsit real es deixa com a línia futura.

6.3 Comportament de la detecció sota càrrega

La prova central avalua si el sensor manté la detecció quan els paquets de l'atac queden amagats en un volum creixent de trànsit de fons, tant en l'entorn emulat com en producció. La Figura 4 i la Figura 5 resumeixen els resultats en ambdós entorns, que es discuteixen a continuació.

6.3.1 Entorn emulat

La càrrega oferta es llegeix del comptador real de la interfície de captura, no de la taxa enviada. La mesura es va instrumentar per garantir-ne la repetibilitat: trànsit de fons generat amb Python des de *LabPC* cap a la VLAN 30, CPU de Suricata mesurat via `/proc/[pid]/stat`, pèrdua comparant el comptador de la interfície amb `kernel_packets` del `stats.log` (soroll ~3%), i latència de detecció lligada al `eve.json` sobre 240 execucions de l'*exploit* de Marimo distribuïdes pels règims de càrrega.

La Figura 4 mostra tres mètriques simultànies: el consum de CPU de Suricata (eix esquerre, blava), la pèrdua de captura (eix esquerre, verda) i la latència de detecció mediana de l'*exploit* de Marimo (eix dret, vermella). La CPU puja de manera lineal de 0,33% en *idle* fins al 3,5% a 37.500 pps, amb un increment d'aproximadament 0,08% per cada 1.000 pps addicionals. Aquesta tendència lineal és coherent amb el model de Lukaseider et al. [17], que observen que la CPU per nucli de Suricata es manté constant quan el sensor opera a saturació (descarta l'excés de paquets), però puja linealment quan està per sota del límit de processament, que és exactament el nostre cas. La pèrdua de captura es manté dins del soroll de la mesura ($\leq 3\%$) en tot el rang, confirmant que el sensor captura el trànsit de forma neta. La latència de detecció es manté plana entre 7 i 14 ms fins a 30.000 pps, i només augmenta fins a uns 31 ms al sostre d'emulació, on conviuen la saturació de l'OVS i l'inici de pèrdues de paquet.

L'*exploit* de Marimo es detecta correctament en tots els règims de càrrega, fins al sostre que l'emulació pot generar.

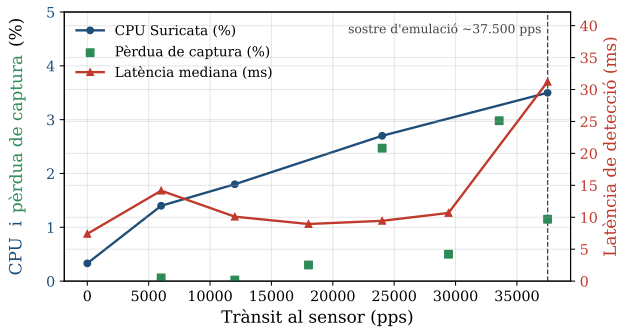


Fig. 4: Consum de CPU, pèrdua de captura i latència de detecció de Marimo en funció del trànsit ofert al laboratori.

La detecció per llindar de la força bruta SSH, que depèn del nombre d'intents per origen i no del volum de fons, es manté igualment. A més, durant una finestra de 60 segons amb 234.867 paquets de fons sense cap atac, cap signatura es va activar, i per tant no es van observar falsos positius en aquesta finestra, amb el conjunt mínim de tres regles del laboratori. En una prova complementària, la força bruta distribuïda des d'una fins a quatre màquines simultànies es va identificar de manera independent gràcies a `track_by_src`.

6.3.2 Producció

Les mètriques d'aquest apartat són pròpies, mesurades directament sobre el sensor de producció (158.109.79.21, Debian 13), i no fons externes. Per aïllar l'efecte de la càrrega del trànsit de fons, la prova es va executar de matinada, cap a les 02.00, quan el trànsit orgànic del DEIC és mínim, injectant trànsit sintètic generat amb Python fins a saturar l'enllaç de captura d'1 Gb/s. La mesura és la mateixa que a l'entorn emulat.

En aquestes condicions, Suricata processa el trànsit fins a saturar la interfície, al voltant de 93.000 paquets per segon, amb una pèrdua de captura nul·la, un consum de CPU del 62% d'un nucli i una latència de detecció que es manté entre 2 i 16 ms (Figura 5). Aquest resultat confirma quantitativament la predicció del laboratori: el coll d'ampolla és la targeta de captura i no el motor de detecció, que en maquinari real processa diversos gigabits per segon [17].

La diferència de consum entre els dos entorns és coherent amb l'augment simultani del cabal i de la mida del conjunt de regles: al laboratori, el consum es manté per sota del 4% amb tres regles a 37.500 paquets per segon; en producció arriba al 62% amb més de 50.000 signatures ET Open [16] a 93.000 paquets per segon. Tots dos factors, un conjunt de regles més gran i un cabal superior, eleven la càrrega de treball del motor de detecció, que l'arquitectura *multithread* de Suricata absorbeix repartint-la entre més fils a mesura que la demanda creix. Això explica el salt del 4% al 62% sense que el sensor arribi a cap límit de processament. En règim de producció, el *mirror* lliura un trànsit net, sense pèrdua ni duplicació de paquets i amb una taxa de falsos positius nul·la per a les regles d'atac avaluades. La pèrdua només apareix sota la prova de saturació amb ràfegues, que desborden momentàniament la memòria intermèdia de captura. Fins i tot llavors la detecció es manté, ja que l'*exploit* de prova es va detectar de manera repetida durant tota la prova.

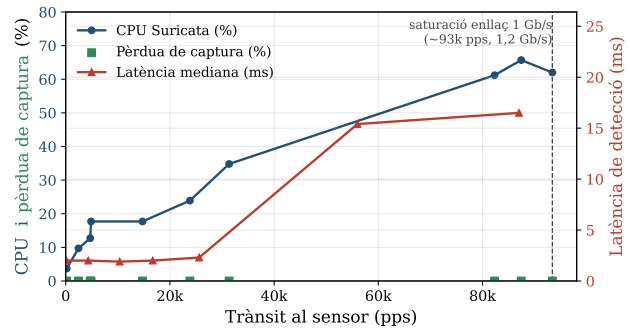


Fig. 5: Consum de CPU, pèrdua de captura i latència de detecció en funció del trànsit ofert sobre el trànsit real del DEIC.

6.4 Temps mitjà de detecció i resposta

En seguretat, el *Mean Time To Detect* (MTTD) i el *Mean Time To Respond* (MTTR) són les mètriques clau per mesurar l'eficàcia d'un SOC [2]. La Figura 6 desglossa el MTTR de notificació (detecció → correu) en quatre etapes successives: MTTD, indexació SIEM, consulta periòdica del SOAR i *playbook* + SMTP. Aquesta descomposició permet identificar on es consumeix el temps i on es poden fer millores.

El MTTR de notificació total se situa de l'ordre de mig minut en ambdós entorns. Desglossant-lo per etapes, la detecció és més ràpida en maquinari real (latències al voltant de 2 ms, enfront dels 7 a 14 ms del laboratori), probablement per la major potència de processament de l'amfitrió físic. La indexació SIEM és més lenta en producció (2,5 vs 1 s), probablement per la major volumetria d'esdeveniments de fons. El terme dominant en ambdós casos és l'espera de la consulta periòdica del SOAR (configurable a 60 s, mitjana observada ~30 s), seguit del *playbook* i SMTP (~5 s). El coll d'ampolla és, doncs, la consulta periòdica del SOAR, un paràmetre configurable, no el sensor ni el SIEM.

6.5 Troballes de seguretat sobre trànsit real

El resultat de més valor del desplegament no prové dels escenaris dissenyats sinó del trànsit orgànic del departament. En operació, el sensor ha detectat dues exposicions de credencials en clar per HTTP Basic, que il·lustren l'abast de la monitorització passiva.

6.5.1 Credencials de servei de tercer

L'equip 158.109.79.233 s'autentica contra un node Bitcoin Core (158.109.79.13, port 8332) mitjançant HTTP Basic sobre HTTP pla, de manera que la capçalera d'autorització viatja en clar i és visible per a qualsevol equip del segment. Amb aquesta credencial es controla el node i la cartera per RPC (*Remote Procedure Call*), des de consultar el saldo fins a exportar claus privades. El servei rep 184.839 peticions HTTP, sent l'equip més actiu de la xarxa, la qual cosa amplifica la gravetat de l'exposició. Un adversari que captés aquest trànsit podria transferir fons, modificar adreces de destinatari o esborrar el registre de transaccions per amagar el rastre, tot des de qualsevol màquina del segment sense necessitat de vulnerabilitat prèvia.

6.5.2 Credencials de la infraestructura pròpia

L'API REST del servidor GNS3 (158.109.79.156, port 3080), emprat per construir el laboratori d'aquest

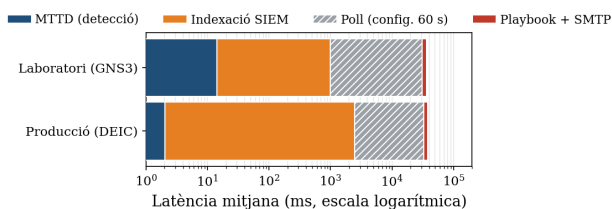


Fig. 6: Temps de notificació *end-to-end* (detecció → correu) en ambdós entorns.

mateix treball, es serveix sobre HTTP pla, i un client (158.109.79.130) hi envia el testimoni d'administració en cada petició, també com a HTTP Basic en clar. El NIDS ha captat, doncs, el seu propi pla de gestió circulant sense xifrar, amb accés total a l'entorn de simulació. Un adversari amb aquest testimoni podria aturar el sensor NIDS, esborrar les alertes del SIEM, modificar la topologia del laboratori o obrir consoles remotes a qualsevol màquina, compromentent la integritat de tota la validació del projecte.

Totes dues troballes, sobre un mateix punt de captura, mostren que un sol protocol en clar al segment lliura les credencials a qualsevol observador. Aquest resultat fonamenta el valor del NIDS més enllà dels dos escenaris d'atac, ja que detecta exposicions reals i posa en pràctica un cas d'ús de detecció de robatori de credencials.

6.6 Limitacions de la validació

La validació presenta tres limitacions principals. En primer lloc, el sensor no s'ha pogut portar al seu propi límit de saturació en cap dels dos entorns: al laboratori emulat el sostre de trànsit (uns 37.500 paquets per segon) el fixa l'entorn GNS3 i no Suricata, i en el desplegament real l'enllaç de captura d'1 Gb/s satura abans (cap als 93.000 paquets per segon) que el motor de detecció. El cabal màxim real del sensor resta, doncs, per caracteritzar sobre un enllaç més ràpid. En segon lloc, el *mirror* agrega el trànsit inter-VLAN, però el trànsit intra-VLAN no travessa el punt de captura i constitueix un punt cec actual de la monitorització. En tercer lloc, la caracterització sota càrrega s'ha fet amb trànsit sintètic i l'observació sobre trànsit real de producció és encara de curta durada, de manera que el volum d'esdeveniments i el comportament del sensor a llarg termini només es podran establir del tot amb un període d'operació més extens.

7 CONCLUSIONS

Aquest treball ha dissenyat, implementat, validat i desplegat una solució SOC de tres capes per a la xarxa del DEIC, que la dota de visibilitat sobre el trànsit i de capacitats de detecció i resposta. Integra Suricata com a NIDS, Splunk com a SIEM i Splunk SOAR com a capa de resposta, sobre una captura passiva per *port mirroring* que no altera la infraestructura. S'ha construït i validat en un laboratori GNS3 que reproduïx la segmentació en VLAN del departament, amb una metodologia en espiral que valida cada component abans d'integrar-lo.

Els resultats confirmen l'eficàcia de l'arquitectura escollida: la detecció dels dos escenaris d'atac, una execució remota de codi pre-autenticació (CVE-2026-39987) i una força bruta SSH, s'ha validat *end-to-end* i de manera repe-

tible, des de la captura fins a la visualització al SIEM i la resposta automatitzada al SOAR.

La prova de càrrega aporta l'evidència quantitativa més rellevant: Suricata manté la detecció íntegra dels dos atacs encara que els seus paquets quedin amagats entre desenes de milers de datagrames de fons, cosa que valida la viabilitat del desplegament passiu i descarta el sensor com a coll d'ampolla. L'anàlisi del volum d'esdeveniments ha mostrat, a més, que el cost del SIEM el determina el nombre d'alertes i no l'ample de banda, una lliçó que ha orientat l'acotament de les regles.

Amb aquests resultats, els objectius O1 a O7 queden assolits, i també l'objectiu O8, la culminació del projecte: el desplegament i la validació de la solució sobre la xarxa real del departament, amb la notificació per correu a través del *relay* intern del DEIC (*skynet*) ja operativa i el bloqueig automàtic validat al laboratori. A més, el desplegament sobre trànsit real ha demostrat el valor operatiu de la solució, en detectar exposicions de credencials en clar fins aleshores no vigilades. Aquestes aportacions s'integren en una arquitectura SOC reproduïble, validada en laboratori i desplegada com a prova pilot sobre la xarxa real del departament.

Més enllà de l'eficàcia tècnica, convé valorar-ne el cost, les llicències i l'escalabilitat. El desplegament reaprofit el maquinari existent del departament i es recolza en components de codi obert, com Suricata i el conjunt de regles ET Open, i en la llicència Community de Splunk SOAR, gratuïta i limitada a cent accions diàries, suficient per al volum del desplegament. El component amb un cost potencial és Splunk: el treball ha emprat una llicència de desenvolupament, però, com a institució pública sense ànim de lucre, la UAB pot optar al programa de donació de Splunk per a entitats d'aquesta mena, que ofereix una llicència gratuïta i renovable anualment de 10 GB diaris, suficient per al volum mesurat. Altrament, un desplegament permanent requeriria una llicència comercial o una alternativa de codi obert com ELK. Pel que fa a l'escalabilitat, la VLAN de gestió actua com a frontera de portabilitat que permet replicar l'arquitectura amb canvis mínims fora del DEIC, com en altres departaments o al CPD (Centre de Processament de Dades) de l'Escola, on un únic punt d'agregació del trànsit donaria visibilitat a un conjunt més ampli de serveis compartits.

7.1 Contribucions del treball

En conjunt, el treball aporta tres contribucions. En primer lloc, demostra que el moviment lateral inter-VLAN des d'adreces internes legítimes és un punt cec dels controls perimetral, i només es fa visible en un punt de captura que agregui el trànsit entre segments, com el *mirror* central. En segon lloc, defineix una política de resposta SOAR de dos nivells segons la gravetat de l'amenaça. En tercer lloc, aporta un mecanisme de bloqueig fora de banda, idempotent i determinista.

7.2 Limitacions

L'abast d'aquests resultats, però, té límits que convé precisar. Les limitacions tècniques de la validació, detallades a la secció de resultats, es resumeixen en tres punts: el motor de detecció no s'ha pogut portar al seu límit de saturació en cap dels dos entorns, la cobertura es restringeix de moment al trànsit inter-VLAN i deixa l'intra-VLAN com a punt cec, i l'observació sobre producció és encara de curta durada.

Pel que fa als objectius, tots queden assolits, però l'objectiu O5 només ho fa plenament al laboratori. En el desplegament real, la resposta automatitzada s'ha limitat a la notificació per correu, mentre que la modificació automàtica del *firewall* sobre trànsit de producció s'ha posposat de manera deliberada, per prudència davant d'un entorn en servei. Aquesta contenció automàtica, validada al laboratori, queda doncs pendent d'activar sobre la xarxa real. Aquestes limitacions delimiten l'abast del que s'hi conclou i orienten directament les línies de continuació.

7.3 Línies futures

Un cop desplegada la solució a la xarxa real, es plantegen cinc línies de continuació:

- **Ampliació i depuració de signatures:** ampliar el conjunt de regles del NIDS per cobrir nous vectors, afinar-lo amb un període d'observació més llarg sobre el trànsit real per reduir-ne els falsos positius, i suprimir les signatures que aporten informació redundant o no accionable.
- **Nous playbooks de resposta:** afegir nous fluxos de resposta per a les deteccions que aquestes regles aportin, com la modificació automàtica del *firewall* sobre trànsit de producció.
- **Enriquiment amb RAG:** incorporar un assistent d'IA basat en RAG (*Retrieval-Augmented Generation*) sobre el SOAR que enriqueixi cada alerta amb context i en suggereixi la resposta, a partir de fonts com la documentació interna i l'històric d'incidents, per reduir el temps d'anàlisi durant la contenció. Aquest assistent s'executaria sobre un model de llenguatge local, ja desplegat a Zeus (un equip del DEIC amb una GPU RTX 5090) per a aquesta finalitat però encara no integrat, de manera que les dades sensibles dels incidents no surten de la xarxa del departament.
- **Sensors distribuïts:** desplegar sensors addicionals als commutadors d'accés per capturar el trànsit intra-VLAN, que actualment no travessa el *mirror* central, i eliminar així el punt cec de la monitorització.
- **Notificació en temps real:** substituir la consulta periòdica (*polling*) per l'enviament directe d'alertes del SIEM al SOAR, que elimina el coll d'ampolla del MTTR de notificació per a accions urgents.

ÚS DE LA IA GENERATIVA

En l'elaboració d'aquest treball s'han emprat eines d'IA generativa, sempre sota la supervisió i la revisió crítica de l'autor. El seu ús s'ha limitat a la millora de la redacció i la correcció lingüística de la memòria i a la depuració de codi auxiliar. Tots els aspectes avaluable del treball són obra íntegra de l'autor.

AGRAÏMENTS

Vull agrair al tutor, el Sergi Robles Martínez, la seva guia i el seu seguiment al llarg del projecte, i al Departament d'Enginyeria de la Informació i de les Comunicacions (DEIC) el suport i els recursos cedits en el marc de la beca de col·laboració [18]. Finalment, agraeixo a la família i als amics el seu suport durant la realització del treball.

REFERÈNCIES

- [1] B. W. Boehm, "A spiral model of software development and enhancement," *Computer*, vol. 21, núm. 5, pàg. 61–72, 1988. DOI: 10.1109/2.59.
- [2] M. Vielberth, F. Böhm, I. Fichtinger i G. Pernul, "Security Operations Center: A systematic study and open challenges," *IEEE Access*, vol. 8, pàg. 227756–227779, 2020. DOI: 10.1109/ACCESS.2020.3045514.
- [3] K. Scarfone i P. Mell, "Guide to Intrusion Detection and Prevention Systems (IDPS)," NIST Special Publication 800-94, 2007. DOI: 10.6028/NIST.SP.800-94.
- [4] M. Roesch, "Snort: Lightweight intrusion detection for networks," a *Proc. 13th USENIX Conf. System Administration (LISA)*, 1999, pàg. 229–238.
- [5] Open Information Security Foundation (OISF), "What is Suricata," Documentació de Suricata, 2026. [En línia]. Disponible a: <https://docs.suricata.io/en/latest/what-is-suricata.html>. [Consultat: 12 de març de 2026].
- [6] J. S. White, T. Fitzsimmons i J. N. Matthews, "Quantitative analysis of intrusion detection systems: Snort and Suricata," a *Proc. SPIE 8757, Cyber Sensing*, 2013. DOI: 10.1117/12.2015616.
- [7] G. González-Granadillo, S. González-Zarzosa i R. Díaz, "Security Information and Event Management (SIEM): Analysis, trends, and usage in critical infrastructures," *Sensors*, vol. 21, núm. 14, art. 4759, 2021. DOI: 10.3390/s21144759.
- [8] Splunk, "Introduction to Splunk SOAR Playbooks," Documentació de Splunk, 2025. [En línia]. Disponible a: <https://help.splunk.com/en/splunk-soar/soar-on-premises/build-playbooks/6.4.1/introduction-to-splunk-soar-on-premises-playbooks>. [Consultat: 13 de maig de 2026].
- [9] Palo Alto Networks, "Cortex XSOAR: Security orchestration, automation and response," documentació del producte, 2025. [En línia]. Disponible a: <https://www.paloaltonetworks.com/cortex/cortex-xsoar>. [Consultat: 13 de maig de 2026].
- [10] J. C. Neumann, *The Book of GNS3: Build Virtual Network Labs Using Cisco, Juniper, and More*. San Francisco, CA, EUA: No Starch Press, 2015. ISBN 978-1-59327-554-9.
- [11] Splunk, "Louisiana State University builds a next-generation SOC," 2023. [En línia]. Disponible a: https://www.splunk.com/en_us/customers/success-stories/lsu.html. [Consultat: 17 de febrer de 2026].
- [12] J. A. Donenfeld, "WireGuard: Next generation kernel network tunnel," a *Proc. Network and Distributed System Security Symposium (NDSS)*, 2017. DOI: 10.14722/ndss.2017.23160.
- [13] Splunk, "About the Universal Forwarder," Documentació de Splunk, v10.2.3, 2025. [En línia]. Disponible a: <https://docs.splunk.com/Documentation/Forwarder/10.2.3/Forwarder/Abouttheuniversalforwarder>. [Consultat: 8 d'abril de 2026].
- [14] NIST, "CVE-2026-39987 Detail," *National Vulnerability Database*, 2026. [En línia]. Disponible a: <https://nvd.nist.gov/vuln/detail/CVE-2026-39987>. [Consultat: 11 de juny de 2026].
- [15] Open Information Security Foundation (OISF), "Thresholding keywords," Documentació de Suricata, 2026. [En línia]. Disponible a: <https://docs.suricata.io/en/latest/rules/thresholding.html>. [Consultat: 14 de març de 2026].
- [16] Proofpoint Emerging Threats, "Emerging Threats Open Ruleset (ET Open)," conjunt de regles obert per a Suricata, llicència BSD, 2026. [En línia]. Disponible a: <https://rules.emergingthreats.net/open/>. [Consultat: 17 de juny de 2026].
- [17] T. Lukaseder, J. Fiedler i F. Kargl, "Performance evaluation in high-speed networks by the example of intrusion detection," *arXiv preprint arXiv:1805.11407*, 2018.
- [18] Agència de Gestió d'Ajuts Universitaris i de Recerca (AGAUR), "Beques de col·laboració en departaments universitaris (COLAB)," Generalitat de Catalunya. [En línia]. Disponible a: <https://agaur.gencat.cat/es/beques-i-ajuts/convocatories-per-temes/Beques-de-collaboracio-en-departaments-universitaris-per-al-curs-2021-2022-COLAB-2021>. [Consultat: 30 de juny de 2026].

APÈNDIX

A.1 Diagrama de Gantt de la planificació

La Figura 7 recull la planificació temporal del projecte, organitzada en els cicles descrits a la Taula 1, amb la distribució de les fites principals al llarg del curs.

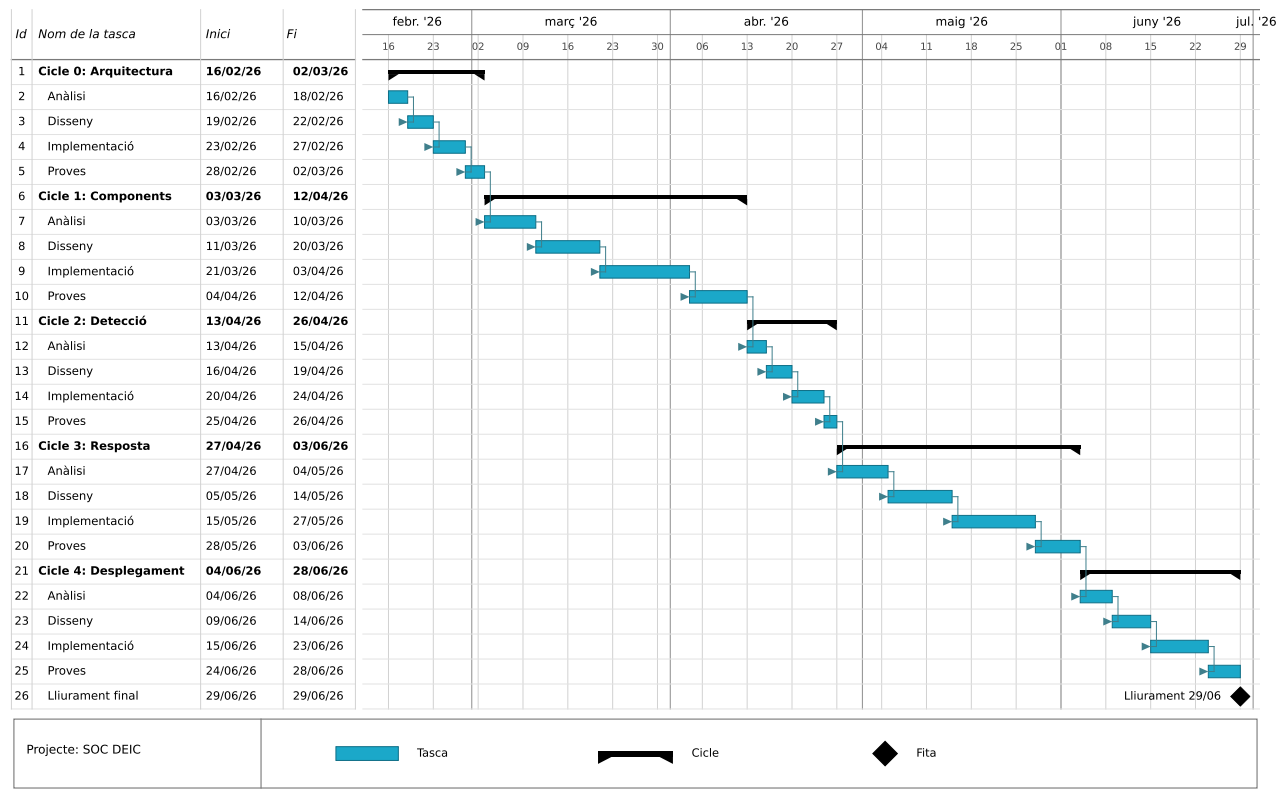


Fig. 7: Diagrama de Gantt amb la planificació del projecte per cicles.

A.2 Topologia del laboratori a GNS3

La Figura 8 mostra la topologia completa del laboratori emulat: la cadena de tres routers VyOS, la segmentació en quatre VLAN i el punt de *mirror* cap al sensor. Reprodueix de manera simplificada però representativa l'arquitectura real del DEIC analitzada a la Figura 2.

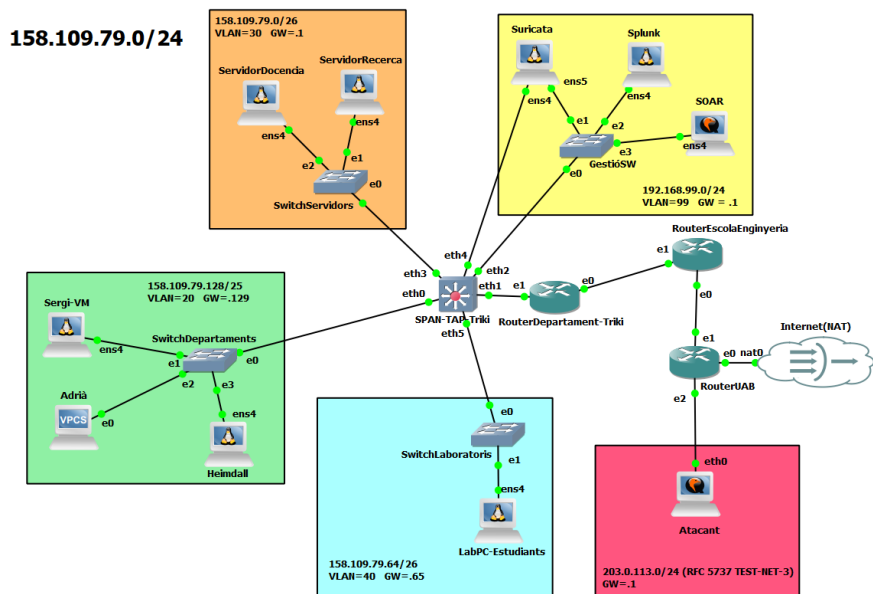


Fig. 8: Topologia del laboratori GNS3: cadena de tres routers VyOS, segmentació en quatre VLAN i punt de *mirror* cap al sensor.

A.3 Dashboard operatiu del SIEM

El *dashboard* construït sobre Splunk explota la informació indexada del NIDS per validar els escenaris d'atac. Se'n recullen dues instàncies: la del laboratori emulat (Figura 9) i la del desplegament real al DEIC (Figura 10).

A Dashboard del simulador

La Figura 9 mostra el *dashboard* del laboratori emulat, amb què s'exerciten i validen els escenaris d'atac.

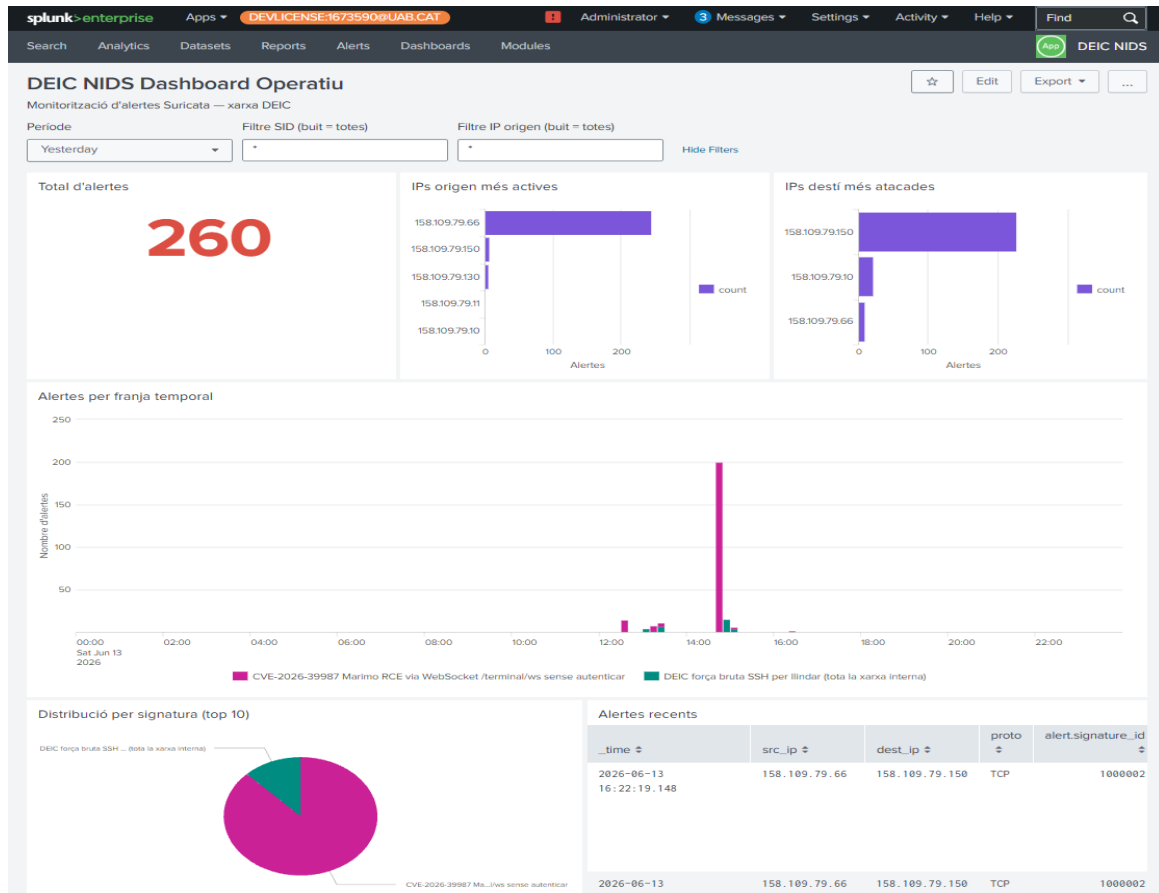


Fig. 9: *Dashboard* operatiu de Splunk al laboratori, alimentat per trànsit simulat.

B Dashboard de producció

La Figura 10 mostra el mateix *dashboard* sobre el desplegament real al DEIC, alimentat amb trànsit de producció.

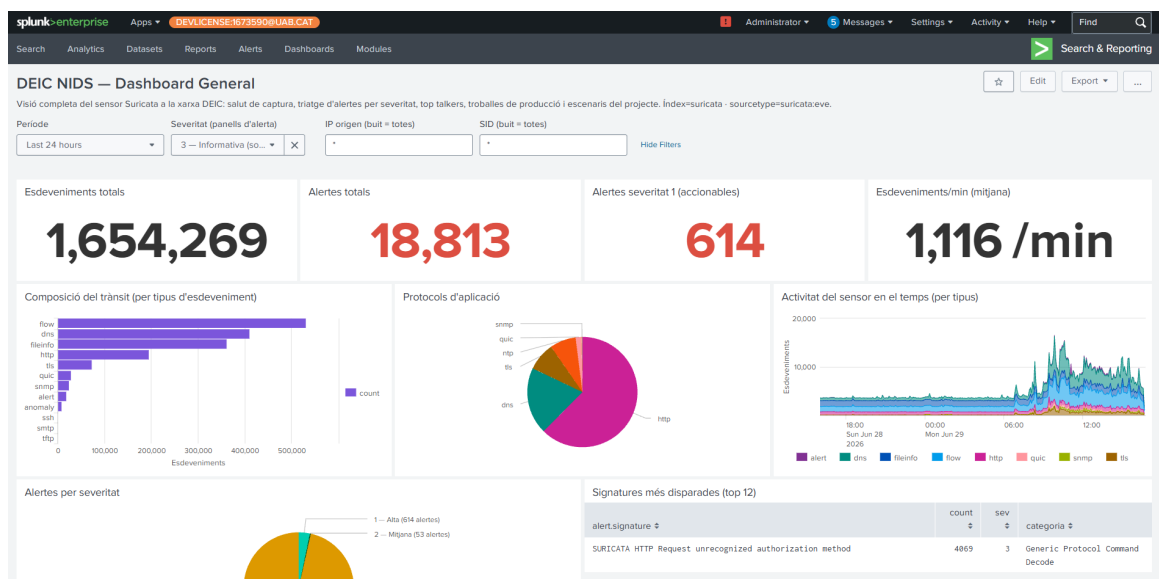


Fig. 10: *Dashboard* operatiu de Splunk en producció, alimentat amb el trànsit real captat a la xarxa del DEIC.

A.8 Cerca d'exemple al SIEM

Les dades del NIDS s'exploten al SIEM amb cerques en llenguatge SPL (*Search Processing Language*). La Figura 15 mostra una cerca a Splunk sobre l'índex `suricata` i el seu resultat: filtra les alertes per identificador de regla i en tabula els camps rellevants (marca de temps, adreces d'origen i destinació i firma disparada).

New Search

index=suricata event_type=alert alert.signature_id=1000002
| table _time, src_ip, dest_ip, alert.signature, alert.severity
| sort - _time

11 events (23/06/2026 00:00:00.000 to 30/06/2026 00:43:13.000) No Event Sampling

_time	src_ip	dest_ip	alert.signature	alert.severity
2026-06-26 19:23:45.983	158.109.79.156	158.109.79.21	CVE-2026-39987 Marimo RCE via WebSocket /terminal/ws sense autenticar	1
2026-06-26 19:23:43.466	158.109.79.156	158.109.79.21	CVE-2026-39987 Marimo RCE via WebSocket /terminal/ws sense autenticar	1
2026-06-26 19:23:48.954	158.109.79.156	158.109.79.21	CVE-2026-39987 Marimo RCE via WebSocket /terminal/ws sense autenticar	1
2026-06-26 19:23:38.435	158.109.79.156	158.109.79.21	CVE-2026-39987 Marimo RCE via WebSocket /terminal/ws sense autenticar	1
2026-06-26 19:23:35.916	158.109.79.156	158.109.79.21	CVE-2026-39987 Marimo RCE via WebSocket /terminal/ws sense autenticar	1
2026-06-26 19:23:33.397	158.109.79.156	158.109.79.21	CVE-2026-39987 Marimo RCE via WebSocket /terminal/ws sense autenticar	1
2026-06-26 19:23:38.878	158.109.79.156	158.109.79.21	CVE-2026-39987 Marimo RCE via WebSocket /terminal/ws sense autenticar	1
2026-06-26 19:23:28.357	158.109.79.156	158.109.79.21	CVE-2026-39987 Marimo RCE via WebSocket /terminal/ws sense autenticar	1
2026-06-26 19:23:25.839	158.109.79.156	158.109.79.21	CVE-2026-39987 Marimo RCE via WebSocket /terminal/ws sense autenticar	1
2026-06-26 19:23:23.321	158.109.79.156	158.109.79.21	CVE-2026-39987 Marimo RCE via WebSocket /terminal/ws sense autenticar	1
2026-06-26 19:23:28.804	158.109.79.156	158.109.79.21	CVE-2026-39987 Marimo RCE via WebSocket /terminal/ws sense autenticar	1

Fig. 15: Cerca SPL a Splunk sobre l'índex `suricata` amb el resultat tabulat (alertes de l'escenari Marimo, SID 1000002).

A.9 Exemple del format de les regles

Una regla de Suricata es compon d'una capçalera amb l'acció, el protocol, les adreces i ports d'origen i destinació i el sentit del trànsit, seguida d'un conjunt d'opcions entre parèntesis com el missatge descriptiu (`msg`), l'identificador de regla (`sid`), el número de revisió (`rev`) i les paraules clau de coincidència (per exemple `itype` o `content`). La Figura 16 en mostra dos exemples al fitxer `my.rules`.

```
debian@debian:/etc/suricata/rules$ cat my.rules
alert icmp any any -> 158.109.79.10 any (msg:"ICMP echo a
  ServidorDocencia"; itype:8; sid:1000001; rev:2;)
alert http any any -> 158.109.79.10 any (msg:"HTTP GET a
  ServidorDocencia"; flow:established,to_server; http.metho
d; content:"GET"; sid:1000002; rev:2;)
```

Fig. 16: Exemple del format de les regles de Suricata al fitxer `my.rules`: una regla ICMP i una regla HTTP amb la capçalera i les opcions de detecció.