UNIVERSITAT D'ANDORRA

TREBALL FINAL de CARRERA

Curs 2024 2025

Per optar al títol de

Bàtxelor en informàtica

SIEM-AI: Proactive threat detection system with artificial intelligence techniques

Autor/a *Adrià Ferran Fernández Laxio*

Tutors/es *Amadeu Albós Raya*

Sant Julià de Lòria, 20 de juny 2025

Dedicatòria i agraïments

Aquest treball és fruit de mesos d'esforç, dedicació i aprenentatge, però també d'acompanyament, suport i molta paciència per part de les persones que m'envolten.

En primer lloc, vull dedicar aquest treball a la meva família i parella. Gràcies per ser-hi sempre, per animar-me a continuar endavant fins i tot en els moments més difícils. Sense la vostra comprensió i suport incondicional, aquest camí hauria estat molt més difícil.

També vull agrair als meus amics i amigues, aquells que m'han ajudat i animat en moments personals i d'estudi. Gràcies per fer-me riure quan més ho necessitava i per fer més lleugeres les etapes més dures.

Vull expressar també el meu agraïment als professors i professores que m'han guiat durant la realització d'aquest projecte. Gràcies per la vostra dedicació, pel vostre criteri i per oferir-me sempre una mirada crítica i constructiva.

Fitxa del treball final

Títol

SIEM-AI: Proactive threat detection system with artificial intelligence techniques

Autor/a

Adrià Ferran Fernández Laxio

Tutors

Amadeu Albós Raya

Àrea del treball

Cybersecurity

Paraules clau

Detection, artificial intelligence, security.

Resum

This project consists of the development of an advanced SIEM (Security Information and Event Management) system integrated with artificial intelligence technologies for the proactive detection of security threats. The main objective is to create a platform capable of collecting, processing and analyzing security logs from multiple sources to identify patterns of malicious activity and generate real-time alerts.

The SIEM core processes the data through a system of specialized parsers that normalize logs from different formats to a unified schema. Subsequently, an artificial intelligence engine analyzes each event using machine learning algorithms such as Isolation Monte for anomaly detection and neural networks implemented with PyTorch for threat classification. This hybrid approach allows detecting both known attacks and unknown anomalous behavior.

An event correlation component applies rule and pattern based logic to identify complex attack chains following models such as the Cyber Kill Chain. This allows detecting advanced persistent threats that would not be identifiable through individual analysis of logs.

Storage is managed by MongoDB as the main engine and an in-memory fallback system for development environments. The system provides multiple endpoints for queries, statistics and results analysis, facilitating integration with external tools and monitoring dashboards.

The main innovations of the project include the combination of supervised and unsupervised algorithms for more accurate detection, the modular architecture that facilitates scalability, and the ability to process large volumes of data in real time while maintaining high performance.

Índex de continguts

Índex de figures

1 Pla de treball

1.1 Context i justificació

En l'actual context digital, la seguretat informàtica ha esdevingut una prioritat estratègica per a les organitzacions tant públiques com privades. L'increment constant dels ciberatacs representa un risc creixent, amb un impacte directe sobre la confidencialitat, la integritat i la disponibilitat de la informació. Segons dades de *l'Agència Nacional de Ciberseguretat d'Andorra* (ANC-AD), durant el 2024 es van registrar més de 1.000 incidents de ciberseguretat, fet que suposa un augment del 15% al 18% respecte a l'any anterior (Cano, 2025). A més, les previsions per al 2025 apunten a un increment encara més accentuat, impulsat per l'ús de tècniques més sofisticades potenciades per la intel·ligència artificial (Revistabyte, 2025).

Aquestes amenaces afecten especialment aquelles xarxes i sistemes que gestionen dades sensibles o operacions crítiques, com les d'empreses del sector financer, sanitari o administracions públiques. En molts casos, aquestes entitats no implementen els estàndards internacionals recomanats, com l'ISO 27001, o incompleixen bones pràctiques bàsiques de seguretat (ISN, 2024), deixant obertes múltiples vulnerabilitats. A més, s'ha detectat un augment significatiu d'atacs dins de xarxes internes, com els de tipus "zero-day", que aprofiten vulnerabilitats desconegudes per les quals no existeix encara una solució (Akamai, 2023).

Davant d'aquest panorama, es fa evident la necessitat de reforçar les capacitats de detecció i resposta davant d'incidents. Per aquest motiu, el projecte es centra en el disseny i la implementació d'un sistema intel·ligent basat en IA, capaç d'analitzar, monitorar i alertar en temps real sobre activitats sospitoses dins d'una xarxa, mitjançant tècniques d'aprenentatge automàtic i anàlisi de comportament. Aquest sistema permetrà millorar significativament el temps de detecció i resposta davant de ciberatacs, reduint l'impacte potencial i augmentant la resiliència de la infraestructura tecnològica.

Malgrat que al mercat ja existeixen eines de ciberseguretat, moltes d'elles operen de manera reactiva o no estan adaptades al context específic de les organitzacions. Integrar un sistema intel·ligent suposa una millora significativa, ja que permet adaptar-se a patrons de comportament no previsibles, detectar anomalies sense necessitat de signatures prèvies i evolucionar contínuament davant de noves amenaces.

Així doncs, aquest projecte s'enfoca a la creació d'una solució de ciberseguretat basada en intel·ligència artificial orientada a la detecció proactiva, amb la finalitat de prevenir ciberatacs, optimitzar la gestió de riscos i reforçar la protecció de la informació crítica.

1.2 Objectius

Aquest projecte neix amb la voluntat de combinar l'interès personal i professional:

- L'objectiu principal del projecte és una proposta d'interès personal i professional, per desenvolupar els meus coneixements de ciberseguretat. El projecte permetrà aplicar i ampliar els meus coneixements per desenvolupar una solució final enriquidora.
- Una de les claus del projecte consistirà en aprendre com monitoritzar un entorn real dins de possibles casos d'atacs, analitzant diferents tipus d'amenaces.
- Integrar i treballar amb diferents tecnologies actuals que permetin tindre una visió del funcionament sobre solucions de seguretat.
- Aprendre i implementar mecanismes que pugin detectar atacs dins d'un entorn real.
- Explorar l'ús de tècniques d'intel·ligència artificial per detectar comportaments anòmals.

1.3 Objectius de Desenvolupament Sostenible

- Impacte positiu:

- **ODS 8 - Fomentar el creixement econòmic i el treball digne:** La ciberseguretat és un sector en auge i aquest projecte pot millorar la seguretat informàtica en entorns empresarials, fomentant la innovació en seguretat.
- **ODS 9 - Desenvolupar infraestructures i fomentar la innovació:** La implementació d'aquest sistema, millora la resiliència de les infraestructures digitals i promou la innovació en automatitzacions intel·ligents aplicades a la ciberseguretat.
- **ODS 11 - Ciutats i assentaments més segurs:** Una millor seguretat en organitzacions protegeix infraestructures sensibles com serveis públics, informació critica, governs...
- **ODS 16 - Societats pacífiques i inclusives, justícia i institucions sòlides:** El projecte contribueix a reduir delictes cibernètics i millorar la seguretat digital, protegint a tots els usuaris i la seva informació.

- Impacte negatiu:

- **ODS 10 - Reduir la desigualtat entre països:** El producte final, necessita recursos tècnics i econòmics, fet que pot afavorir empreses amb més capital i deixar enrere petites organitzacions que no poden costejar aquest serveis.
- **ODS 12 - Consum i producció sostenibles:** El desplegament del sistema en servidors, xarxes i infraestructures digitals pot augmentar el consum d'energia i recursos, afectant la sostenibilitat.

- Impacte neutre:

- **ODS 1: Erradicació de la pobresa:** Aquest treball no te relació amb el augment o disminució de la pobresa al mon.
- **ODS 2: Seguretat alimentària i agricultura:** L'alimentació no es veu afectada per cap camp relacionat amb el projecte.
- **ODS 3: Salut i benestar:** Directament, no afecta al sector de la salut, però, podem veure casos que afecten a la seguretat i privacitat del sector.
- **ODS 4: Educació de qualitat:** No esta enfocat a un aprenentatge acadèmic, sinó a casos del mon laboral.
- **ODS 5: Igualtat de gènere:** No influeix en la reducció de la bretxa de gènere ni en polítiques d'igualtat.
- **ODS 6: Aigua neta i sanejament:** No afecta a cap procés relacionat amb l'aigua, es a dir que no estan relacionats.
- **ODS 7: Energies sostenibles:** Encara que l'ús eficient de recursos informàtics podria tenir un petit impacte, el projecte no està orientat a la promoció d'energies netes.
- **ODS 13: Canvi climàtic:** No es relaciona amb polítiques de reducció d'emissions ni accions per combatre el canvi climàtic.
- **ODS 14: Protecció dels oceans:** No es relaciona amb cap ecosistema marítim.
- **ODS 15: Protecció dels ecosistemes terrestres:** No te un impacte en cap ecosistema terrestre.
- **ODS 17: Aliances per al desenvolupament sostenible:** El projecte no està centrat en la cooperació internacional per al desenvolupament sostenible.

1.4 Enfocament i mètode

L'estratègia per resoldre la problemàtica plantejada en aquest projecte, es basa en el desenvolupament d'un producte funcional, mitjançant la integració de tecnologies existents amb components dissenyats específicament per adaptar-se a les necessitats del sistema. Gracies a aquest enfocament, permetrà obtenir una base solida amb solucions ja creades com a infraestructura per crear un producte final.

Per dur a terme, es farà ús de la metodologia en cascada, adient per a projectes amb etapes ben definides. Aquest mètode permet estructurar en fases, garantint una planificació detallada, cosa fonamental en el projecte actual.

La metodologia en cascada és un model seqüencial on s'estructura per tasques i etapes seqüencials agrupades ordenadament. Cada fase es completa abans de passar a la següent, on s'han de completar les tasques de cada etapa per poder continuar al següent pas.

Aquesta metodologia és ideal per aquest projecte, degut a la estructura clara i amb una idea final de tots els passos per realitzar un bon producte. Aquestes fases estaran distribuïdes segons les necessitats del projecte i s'organitzaran de manera endreçada, per assegurar un desenvolupament eficient:

Anàlisis: Definir els requisits i objectius del projecte.

Disseny: Planificar l'arquitectura del sistema i seleccionar les eines adequades.

Implementació: Desenvolupar les funcionalitats i integrar els components.

Proves: Validar el sistema per assegurar que compleix els requisits.

Distribució i manteniment: Llançar el sistema i garantir el seu bon funcionament amb un pla de manteniment.

1.5 Planificació

Anàlisi:
Objectiu: Entendre el problema general, recollir informació prèvia i establir la base del projecte.
Durada: 24/2/25 – 1/3/25

-Definir problemàtica
Objectiu: Redactar de forma clara quin és el problema que es vol resoldre amb el projecte.
Durada: 24/2/25 – 25/2/25

-Creació de diagrames de representació del cas
Objectiu: Elaborar esquemes visuals d'alt nivell del funcionament del sistema.
Durada: 26/2/25 – 1/3/25
Dependències: Definir problemàtica

-Establir objectius finals concrets
Objectiu: Definir amb precisió què ha de complir el sistema per considerar-se funcional.
Durada: 26/2/25 – 1/3/25
Dependències: Definir problemàtica

-Definir totes les funcionalitats del sistema
Objectiu: Llistar i descriure les funcionalitats específiques que oferirà el sistema.
Durada: 26/2/25 – 1/3/25
Dependències: Objectius finals

Disseny:
Objectiu: Plantejar l'arquitectura del sistema abans d'implementar-lo.
Durada: 5/3/25 – 10/3/25
Dependències: Anàlisi

-Representació del projecte amb diagrames d'alt nivell
Objectiu: Crear diagrames generals de l'estructura i funcionament del sistema.
Durada: 5/3/25 – 9/3/25

-Definir els components del sistema
Objectiu: Especificar quins elements formen el sistema i que funció realitzarà.
Durada: 5/3/25 – 10/3/25
Dependències: Diagrames dels components

-Definir aspectes del sistema extra per una resolució correcta
Objectiu: Tenir en compte requisits
Durada: 5/3/25 – 7/3/25
Dependències: Disseny

Lliurament primer seguiment:
Objectiu: Presentar el primer informe de seguiment al tutor per validar l'avanç.
Data: 14/3/25
Dependències: Disseny finalitzat

Implementació:

Objectiu: Desenvolupar, configurar i integrar el sistema segons el disseny previ.
Durada: 14/3/25 – 25/4/25
Dependències: Lliurament primer seguiment

Fase 1:

-Instal·lar eines i preparar un entorn de tre

Objectiu: Configurar un entorn controlat per al desenvolupament i proves.
Durada: 14/3/25 – 21/3/25

-Implementar sistemes de detecció per recopilar dades

Objectiu: Desenvolupar registres que simulin dades reals.
Durada: 14/3/25 – 21/3/25

Fase 2

-Tractar les dades generades

Objectiu: Netejar, transformar i filtrar les dades obtingudes.
Durada: 27/3/25 – 2/4/25
Dependències: Fase 1

-Classificar aquestes dades

Objectiu: Assignar etiquetes i categories a les dades segons el seu comportament i tipus.
Durada: 3/4/25 – 6/4/25
Dependències: Generar dades correctament

Fase 3

-Procesar les dades obtingudes

Objectiu: Analitzar les dades classificades per detectar anomalies.
Durada: 15/4/25 – 21/4/25
Dependències: Classificació de dades

-Establir les polítiques de seguretat

Objectiu: Definir accions que el sistema permet detectar les amenaçes.
Durada: 15/4/25 – 21/4/25
Dependències: Procés de dades

-Crear un sistema de detecció intel·ligent

Objectiu: Desenvolupar un sistema capaç de detectar atacs de forma autònoma i intel·ligent.
Durada: 15/4/25 – 25/4/25
Dependències: Polítiques definides

Proves:

Objectiu: Verificar que el sistema compleix els requisits i funciona correctament.
Durada: 3/5/25 – 9/5/25
Dependències: Implementació completa

-Crear els requisits del sistema a complir

Objectiu: Establir una llista de condicions que el sistema ha de satisfer.
Durada: 3/5/25 – 6/5/25
Dependències: Final implementació.

-Realitzar les proves pertinents

Objectiu: Executar els tests per comprovar que tot funcionament.
Durada: 5/5/25 – 7/5/25
Dependències: Requisits creats

-Avaluar els resultats obtinguts

Objectiu: Analitzar els resultats de les proves i veure si cal millorar alguna part.
Durada: 8/5/25 – 9/5/25
Dependències: Proves realitzades

-Resoldre possibles errors

Objectiu: Corregir els errors detectats durant les proves.
Durada: 8/5/25 – 9/5/25
Dependències: Avaluació de resultats

Lliurament segon seguiment

Objectiu: Presentar l'estat avançat del projecte i obtenir validació.
Data: 10/5/25
Dependències: Proves i millores

Lliurament final

Objectiu: Revisiar i millorar la documentacio amb els comentaris proporcionats
Data: 9/6/25

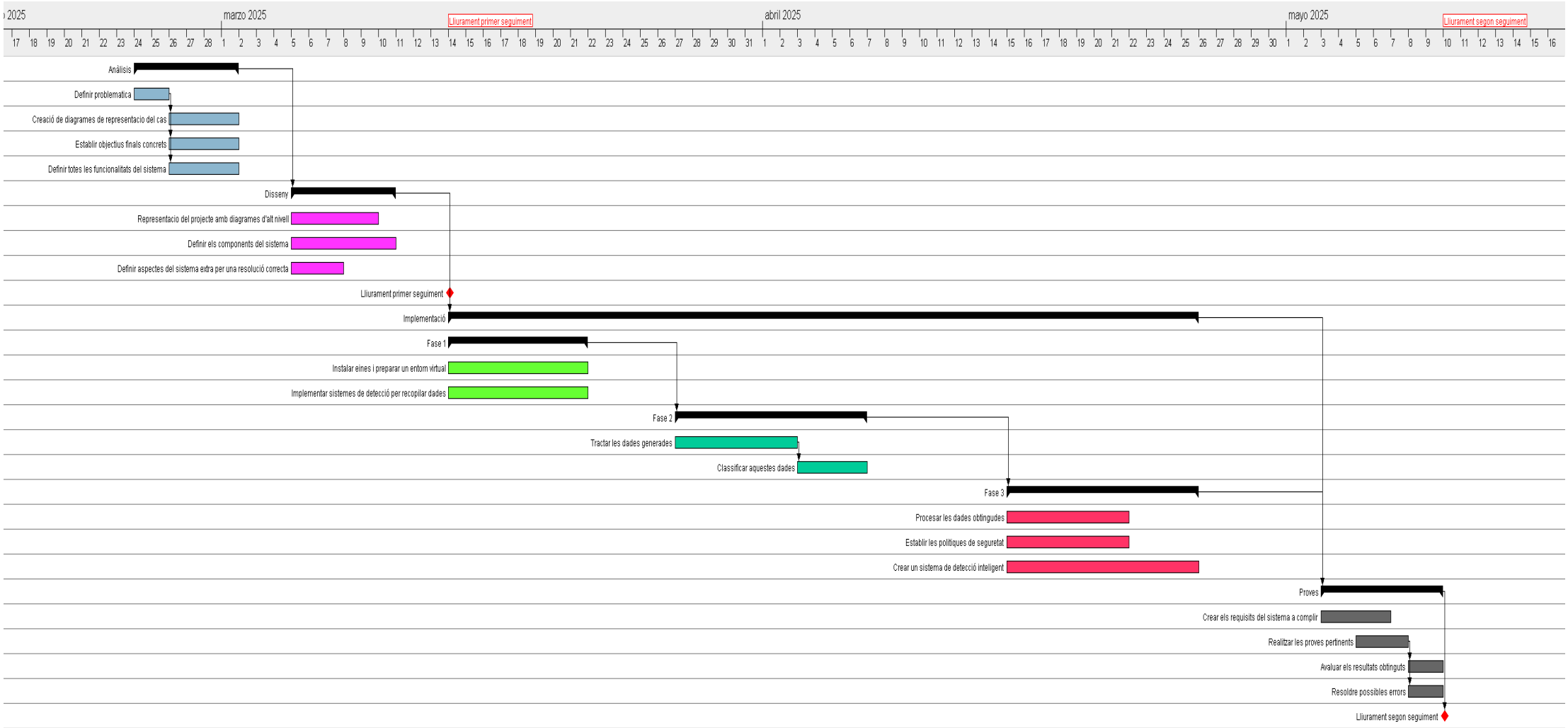


Figura 1. Diagrama GANTT

1.6 Productes finals

Dins del projecte, s'establiran dos principals productes finals que donaran una solució funcional. Aquestes consisteixen en una solució de disseny i una solució d'implementació com a producte mínim viable:

Disseny:

El producte de disseny constitueix el plànol arquitectònic i conceptual del sistema de seguretat. Defineix una solució global orientada a garantir el compliment dels requisits i objectius del sistema. Inclou l'estructura dels diferents components funcionals del sistema i detalla com aquests components s'interrelacionen. A més, s'especifiquen els processos i els fluxos de treball que descriuen el processament de les dades a dins del sistema.

Implementació:

Per un altre costat, el segon producte final és una implementació funcional del sistema, que representa un producte mínim viable . Aquesta implementació materialitza els conceptes definits en el disseny, centrant-se en les funcionalitats essencials per demostrar el funcionament de la solució proposada. L'objectiu principal d'aquest producte és poder validar, mitjançant la simulació d'escenaris, la detecció d'activitats malicioses habituals en sistemes informàtics. Aquesta implementació serveix com a base demostrativa de la capacitat del sistema per identificar i respondre a aquests tipus d'esdeveniments.

1.7 Recursos

Recursos materials:

-Ordinador per el desenvolupament: Equip amb suficient capacitat de processament per programar, provar i executar les simulacions.

-**Eines del sector:** Totes les eines que poden ser utilitzades per al treball.

-**Base de dades:** Emmagatzematge de les dades generades del sistema.

-Memòria RAM: La memòria RAM es fonamental per poder processar una quantitat d'informació ràpidament.

Recursos immaterials:

-Coneixements tècnics: Tot el aprenentatge actual i nous coneixements necessaris per assolir el treball.

-Temps de desenvolupament: La realització de les tasques durant el projecte.

-Recursos: Documentació tècnica, guies, o altres fonts de coneixements.

-Dades reals per proves: Registres de dades necessàries per estudiar casos reals.

1.8 Anàlisi de riscos

1. Riscos relacionats amb el temps de desenvolupament

- **Risc:** L'elevada complexitat tècnica pot provocar retards en la implementació.
- **Prevenció:** Planificació detallada amb un calendari realista i una metodologia per adaptar-se a possibles contratemps.
- **Correcció:** Priorització de tasques crítiques i revisió periòdica dels objectius per ajustar els terminis si és necessari.

2. Riscos relacionats amb la detecció d'atacs i falsos positius

- **Risc:** La detecció d'atacs pot generar falsos positius, afectant la precisió del sistema.
- **Prevenció:** Ús de bases de dades d'amenaces actualitzades i entrenament rigorós de models d'intel·ligència artificial.
- **Correcció:** Ajust i refinament dels algoritmes de detecció mitjançant proves contínues amb dades reals.

3. Riscos relacionats amb la compatibilitat de les eines

- **Risc:** Les eines utilitzades poden tenir problemes de compatibilitat, afectant el seu correcte funcionament.
- **Prevenció:** Selecció prèvia d'eines amb compatibilitat demostrada i realització de proves d'integració.
- **Correcció:** Substitució o ajustament de components incompatibles si es detecten problemes durant la implementació.

4. Riscos relacionats amb coneixements tècnics

- **Risc:** L'exploració d'un camp nou com la intel·ligència artificial pot dificultar el desenvolupament.
- **Prevenció:** Formació prèvia en models de detecció d'amenaces i ús d'IA aplicada a ciberseguretat.
- **Correcció:** Consulta d'experts, documentació oficial i participació en comunitats tècniques.

1.9 Estructura de la memòria

2. Anàlisi: Aquest capítol aborda en profunditat la problemàtica a la qual respon el projecte. Examina les solucions existents en l'àmbit la monitorització. A partir d'aquesta base, es defineixen els objectius i requisits que el projecte ha d'assolir.

3. Disseny: En aquest apartat, es presenta i detalla una proposta de representació del sistema. Es descriu l'arquitectura i els components que formen la solució a la problemàtica definida. A més, les interaccions i fluxos entre els diferents components que formen el sistema. També es contempla les decisions de disseny relacionades amb l'estructura general del sistema per garantir la seva estabilitat i eficiència.

4. Implementació: Aquest capítol detalla com els components del disseny s'han desenvolupat i s'han traduït en codi i components funcionals. Es detallen les eleccions clau fetes en termes de tecnologies, el seu funcionament principal i l'explicació de què resol cada tecnologia utilitzada. També s'explica com s'han integrat aquestes tecnologies dins l'arquitectura general del sistema, mostrant exemples representatius.

5. Proves: En aquesta part es descriu la metodologia emprada per verificar el funcionament del sistema desenvolupat. S'exposa l'estratègia i el pla de proves dissenyats per validar les funcionalitats clau. Finalment, es presenten els resultats obtinguts de l'execució d'aquestes proves, incloent-hi simulacions i l'avaluació final comprovant els requisits i objectius establerts en la fase d'anàlisi.

6. Distribució, manteniment i evolució: El capítol final analitza les opcions i els passos necessaris per desplegar la solució en un entorn de producció. S'aborden les consideracions de manteniment que haurà de realitzar-se dins d'aquest projecte a futur, incloent-hi la seva evolució.

2 Anàlisi

2.1 Problemàtica

En el context actual de la ciberseguretat, les organitzacions operen en un entorn altament digitalitzat i interconnectat, on els riscos i les amenaces evolucionen constantment tant en sofisticació com en freqüència. Aquesta realitat exigeix una vigilància contínua i una capacitat de resposta ràpida davant possibles incidents (IgniteTechnology, 2024).

Les infraestructures informàtiques, són una font fonamental d'informació per detectar anomalies, identificar comportaments sospitosos i controlar incidents. Aquestes infraestructures formades per sistemes i serveis, generen registres anomenats “logs”, que detecten i emmagatzemen cada moviment del sistema. Aquestes dades permeten obtindré informació rellevant de possibles incidents, però normalment les dades son massives, dificultant la detecció d’incidents (Exabeam, 2024).

Degut a aquests problemes, la manca de sistemes de monitoratge, correlació i alerta dels logs, representa una amenaça crítica per a qualsevol entorn. Per un costat, impedeix la detecció d’activitats sospitoses, la identificació de patrons d’atac i la resposta àgil davant incidents de seguretat. Per un altre costat, sense funcionalitats que solucionin el problema, els sistemes operen a cegues incrementant el risc de ser vulnerat sense saber-ho (ManageEngine, 2024).

Per poder analitzar correctament les dades, sense generar falsos positius i millorar la correcta detecció del sistema, la integració de la intel·ligència artificial en l’anàlisi de logs és essencial. Això, permet detectar anomalies en temps real, oferint una protecció més proactiva, escalable i efectiva davant les ciberatacs modernes on cada com son mes complexos (Panther, 2024).

Problemàtiques concretes generals:

1. **Fragmentació de la informació**: Els registres generats, acostumen a ser de diferents serveis o sistemes, on cadascun d’aquests components tenen formats diferents, dificultant la integració de les dades.
   Conseqüències:
   - Manca de visibilitat global sobre el sistema.
   - Dificultat per integrar dades en temps real.
   - Costos addicionals per desenvolupar connectors o adaptadors específics per cada font
2. **Manca de correlació entre esdeveniments:**
   Els esdeveniments de seguretat registrats es troben desconnectats entre si, ja que provenen de sistemes diferents sense una anàlisi centralitzada ni capacitats de correlació automatitzades.
   Conseqüències:
   - Falles en la detecció d’atacs distribuïts o persistents.
   - Falsos positius i falsos negatius, que redueixen l’eficiència del sistema de detecció.

3. **Dificultat en l'escalabilitat:**
   L’augment constant del volum de dades pot ser un problema greu per processar correctament les incidències.
   Conseqüències:
   - Incapacitat per processar grans volums de dades en temps real.
   - Problemes de rendiment en la recepció, anàlisis i emmagatzematge de registres.
   - Latència elevada que afecta negativament la capacitat de resposta ràpida davant incidents.
4. **Absència de mecanismes de priorització d'alertes:**
   Els sistemes de detecció generen grans quantitats d’alertes sense una classificació clara segons la seva gravetat o rellevància.

Conseqüències:

- Saturació d'anàlisis no prioritaris.
- Omissió d'alertes crítiques per excés d'informació trivial.

5. **Detecció tardana d'incidents:**
   Molts incidents de seguretat no es detecten a temps degut a manca de mecanismes de detecció.
   Conseqüències:
   - Pèrdua d'informació sensible o propietat intel·lectual.
   - Temps d'inactivitat operativa que pot afectar la productivitat i els ingressos.
   - Deteriorament de la imatge corporativa i pèrdua de confiança per part de clients i socis.

6. **Dependència d'expertesa humana:**
   La intervenció de personal qualificat a
   Conseqüències:
   - Dificultat per escalar els processos de seguretat davant l'augment de dades.
   - Costos elevats associats a la contractació i formació de personal.

7. **Proves i validació limitades dels sistemes de detecció:**
   La manca d'entorns controlats i escenaris simulats redueix la capacitat de les organitzacions per posar a prova la seva postura de seguretat.
   Conseqüències:
   - Dificultat per identificar punts febles en els mecanismes de detecció.
   - Limitacions a l'hora de formar i millorar sistemes d'IA o models d'aprenentatge automàtic.
8. **Dificultat en la caracterització i classificació dels atacs:**
   Un cop detectat un incident, és fonamental entendre'n la naturalesa, abast i objectius per poder-hi respondre de manera efectiva.
   Conseqüències:
   - Imprecisions en la identificació del tipus d'amenaça.
   - Dificultat per aplicar contramesures específiques i eficients.
   - Problemes per establir protocols de resposta adaptats a cada escenari.

2.2 Estat de l'art

Actualment, existeixen diverses eines SIEM al mercat, cada una amb característiques pròpies i enfocaments específics. Algunes de les plataformes més destacades són Splunk, LogRhythm, Wazuh i IBM QRadar, totes elles orientades a la monitorització, detecció i resposta davant d'amenaces en temps real (eSecurityPlanet, 2024).

Tendències actuals i futures

1. **Ús d'IA per a la ciberseguretat:** Algoritmes de machine learning milloren la detecció d'amenaces en temps real i l'automatització de respostes (Splashtop, 2024).
2. **Zero Trust Security:** Creixement del model de confiança zero, que assumeix que cap usuari o dispositiu és de confiança per defecte, exigint autenticació contínua.
3. **SIEM al núvol:** Solucions SIEM com Microsoft Sentinel i IBM QRadar Cloud permeten escalar la seguretat sense necessitat d'infraestructura pròpia (IBM, 2024).

Solucions actuals:

Wazuh: És una plataforma de seguretat de codi obert que ofereix funcions de monitoratge d'integritat, detecció d'intrusions, resposta a incidents i recopilació de registres.

- Permet l'anàlisi centralitzada de logs en entorns híbrids .
- Integra mòduls per correlacionar esdeveniments i aplicar regles de detecció personalitzades.
- Gratuïta i molt flexible, és una opció popular per organitzacions amb recursos limitats.

LogRythm: Plataforma SIEM amb capacitats avançades d'automatització i anàlisi comportamental.

- Utilitza machine learning per detectar anomalies i amenaces internes o externes.
- Inclou eines de resposta automatitzada i fluxos de treball personalitzables.
- Està dissenyada per reduir la sobrecàrrega d'alertes i millorar el temps de resposta.

Splunk: Una de les plataformes més robustes per a l'anàlisi de dades de seguretat i operatives.

- Permet ingestió massiva de registres des de múltiples fonts amb alta escalabilitat.
- Ofereix Splunk Enterprise Security per correlació avançada d'esdeveniments i detecció proactiva d'amenaces.
- S'adapta bé a entorns grans i complexos, però resultar costosa.

IBMQradar: Una de les eines SIEM mes complertes e millor valorades.

- Plataforma SIEM que recopila, analitza i correlaciona registres i dades de xarxa en temps real.
- Identifica i classifica incidents de seguretat, ajudant a reduir el temps de resposta.
- Integració amb altres eines de seguretat, incloent EDR, SOAR i Threat Intelligence.

Avaluació segons la perspectiva del projecte:

Escalabilitat

- Tecnologies actuals: Les solucions comercials com estan pensades per entorns generals.
- Projecte proposat: Es centra en entorns concrets, amb una solució més adaptada al seu volum i infraestructura.

Cost i accessibilitat

- Tecnologies actuals: Solucions costoses amb llicències i requeriments tècnics elevats.
- Projecte proposat: Implementa alternatives més accessibles per empreses de mida mitjana.

Personalització i control

- Tecnologies actuals: Ofereixen automatització i aprenentatge automàtic, però amb menys control manual directe.
- Projecte proposat: Prioritza control i gestió manual.

2.3 Objectius tècnics

Objectius mínims ha assolir:

1. El sistema ha de poder recopilar dades externes i poder tractar-les correctament classificant les dades obtingudes.
2. El sistema ha de poder detectar els tipus comuns d'atacs, que es produeixen dins d'un entorn o cas real.
3. El sistema ha de poder millorar les deteccions mitjançant algoritmes intel·ligents.
4. S'ha de poder revisar les dades obtingudes i els resultats al llarg del temps.
5. Ha de poder correlacionar events o atacs per identificar atacs mes complexos.

2.4 Requisits tècnics

- Requisits funcionals

RF1 Recopilació de registres: El sistema ha de ser capaç de recopilar registres de seguretat en temps real provinents de diferents fonts.

RF2 Detecció i alerta d'incidents: Detectar e alertar, sobre incidents de seguretat segons la situació, amb serveis automatitzats.

RF3 Anàlisi automatitzat amb IA: El sistema ha d'analitzar els registres mitjançant tècniques d'intel·ligència artificial per identificar comportaments anòmals o potencials amenaces.

RF4 Alertes: Sistema de generació d'alertes a temps.

RF5 Correlació d'esdeveniments: El sistema ha de ser capaç de correlacionar diversos esdeveniments de fonts diferents per identificar atacs complexos.

- Requisits no funcionals

RNF1 Rendiment: Ha de ser capaç de processar i analitzar una gran quantitat de logs eficientment.

RNF2 Disponibilitat: El sistema ha d'assegurar-se de poder donar persistència a les dades i alertes obtingudes.

RNF3 Escalabilitat: Ha de poder adaptar-se segons les necessitats de l'entorn i el seu creixement.

3 Disseny

3.1 Arquitectura

3.1.1 Arquitectura principal del sistema

El sistema SIEM AI està dissenyat per gestionar el cicle complet de processament d'esdeveniments de seguretat, des de la ingesta de logs fins a la generació d'alertes. La solució es compon de diversos components treballant de manera coordinada per processar les dades. A continuació, es presenta un diagrama d'alt nivell que mostra els components principals i les seves interaccions:

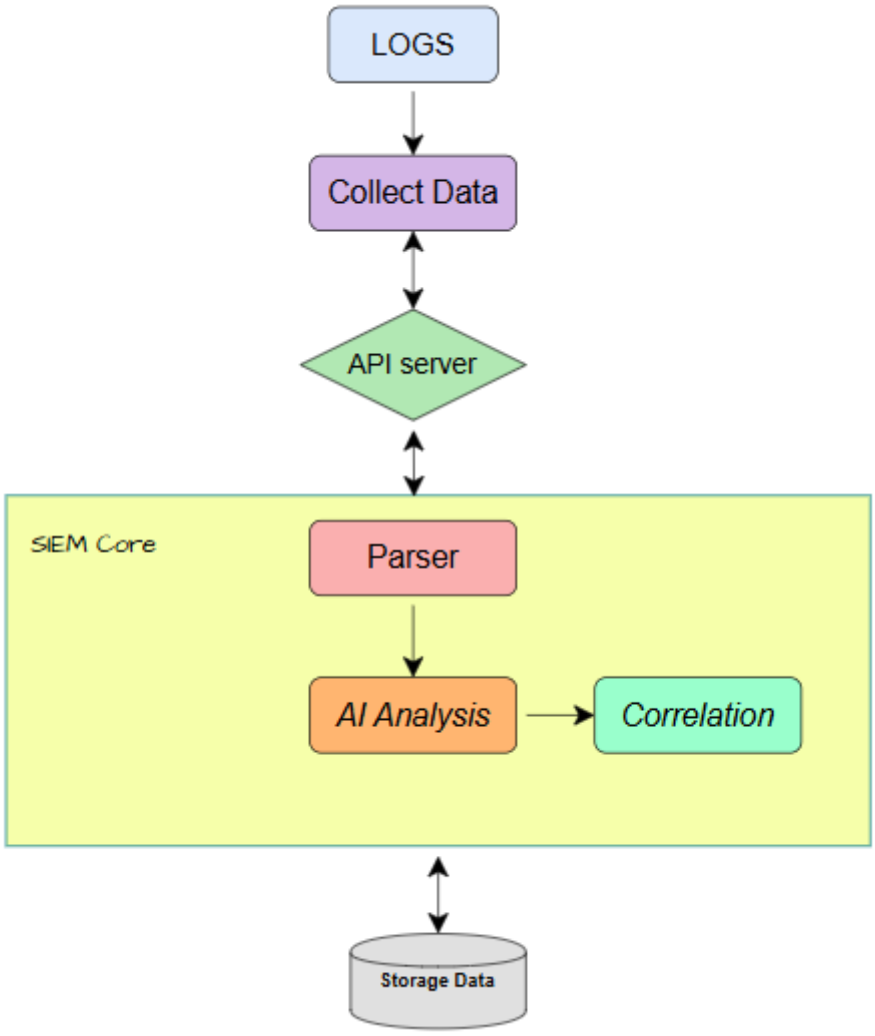


Figura 2. Diagrama de components alt nivell

Explicació del diagrama de components:

1. Generació de Logs (Registres)

Aquest component, representa els múltiples registres generats per diversos sistemes d'un possible entorn real, on proporcionen informació crucial per identificar amenaces. Aquests logs, poden provenir de sistemes i formats diferents, dificultant el seu tractament i anàlisis. Degut a aquest problema, els logs s'han de transformar dins d'un format unificat per poder tractar les dades correctament, no obstant, primer s'enviaran al component "Collect Data" on recopilarà els registres.

2. Collect Data (Recopilació de Dades)

El component de recopilació de dades, actua com el punt d'entrada dels registres generats, on abans d'analitzar-se es recopilen tots els registres unificats dins d'aquest component. Tot hi que es podrien enviar els registres directament al sistema (API), es preferible primer recopilar totes les dades a analitzar dins d'un connector que facilitarà la gestió sobre les diferents fonts de dades. Un cop s'ha recopilat les dades, serà l'encarregat de enviar tots els registres al component API comprovant que les dades son valides i s'envien correctament amb el format correcte.

3. API Server

El component API, fa la funció d'intermediari entre la part del SIEM System i els logs generats. Interactua per rebre els registres correctament i dirigir-los al sistema SIEM per processar-los de manera eficient i controlada. A mes, actuarà com a punt final (endpoint) per rebre peticions tant d'entrada com de sortida, per obtenir resultats d'anàlisis.

4. Parser (Normalitzador)

El component Parser, realitza la primera funció principal del SIEM, es a dir, la unificació de les dades dins d'un sol format. Actua com a traductor de logs per obtenir instancies ordenades i estructurades per camps, a partir de logs sense tractar. Això, permet al sistema identificar la informació rellevant que conte cada registre, fins i tot quan aquests inclouen moltes dades agrupades sense una estructura clara.

5. AI Analysis (Anàlisi d'Intel·ligència Artificial)

Aquest component és el nucli principal del sistema, ja que implementa diversos algorismes d'intel·ligència artificial per identificar possibles amenaces. Rep els registres normalitzats del component anterior, amb els camps rellevants ja identificats, i tot seguit analitza les dades mitjançant diferents mètodes i tècniques d'anàlisi.

6. Correlation (Correlació)

El component de correlació s'encarrega d'identificar patrons d'atacs complexos mitjançant la relació entre diferents esdeveniments. Gràcies a la informació proporcionada pel component d'anàlisi amb intel·ligència artificial, és capaç de detectar seqüències d'esdeveniments que poden indicar una campanya d'atac coordinada o persistents.

7. Storage Data (Emmagatzematge de Dades)

Aquest component s'encarrega de proporcionar persistència a totes les dades rellevants que genera o processa el sistema. Inclou tant els logs normalitzats com les alertes generades per l'anàlisi i la correlació. L'emmagatzematge pot fer-se en memòria (per a entorns de prova) o en bases de dades per a entorns reals, assegurant l'accés ràpid, consultes eficients.

3.1.2 Patrons principals

1. Patró Singleton

Problema que Soluciona: Evita la creació múltiple d'instàncies crítiques del sistema que podrien generar conflictes de recursos, inconsistències d'estat i duplicació de processos essencials.

Funcionament:
Garanteix que només existeixi una instància de components crítics del sistema. Quan es sol·licita una nova instància, comprova si ja existeix una i la retorna, evitant crear duplicats.

2. Patró Strategy

Problema que Soluciona:
La necessitat de processar informació de múltiples formats diferents sense crear un sistema monolític i rígid que sigui difícil de mantenir i estendre.

Funcionament:
Defineix una família d'algoritmes de processament intercanviables, encapsula cadascun d'ells i els fa intercanviables. El sistema selecciona dinàmicament el processador adequat segons el tipus d'informació detectada. Cada processador implementa la mateixa interfície però amb lògica específica per al seu domini.

3. Patró Factory Method

Problema que Soluciona:
La complexitat de crear objectes amb configuracions específiques i la necessitat d'amagar els detalls de construcció als usuaris del sistema.

Funcionament:
Proporciona una interfície per crear famílies d'objectes relacionats sense especificar les seves implementacions concretes. Centralitza la lògica de creació i permet aplicar configuracions estàndard. Els clients sol·liciten objectes a través de mètodes especialitzats sense conèixer els detalls d'implementació.

4. Patró Pipeline

Problema que Soluciona:
La necessitat de processar informació a través de múltiples etapes seqüencials de forma ordenada, modular i eficient.

Funcionament:
Organitza el processament en una cadena d'etapes on cada una realitza una tasca específica i passa el resultat a la següent. Evita acoblar directament l'inici del processament amb el final, permetent que múltiples etapes puguin gestionar la informació.

5. Patró Adapter

Problema que Soluciona:
La incompatibilitat entre diferents estructures d'informació i la necessitat de treballar amb un format unificat per a l'anàlisi i correlació.

Funcionament:
Permet que sistemes amb interfícies incompatibles treballin junts. Actua com un traductor entre l'estructura original de la informació i l'estructura esperada pel sistema. Converteix una interfície en una altra que els components principals poden entendre i processar.

3.1.3 Fluxe principal del sistema:

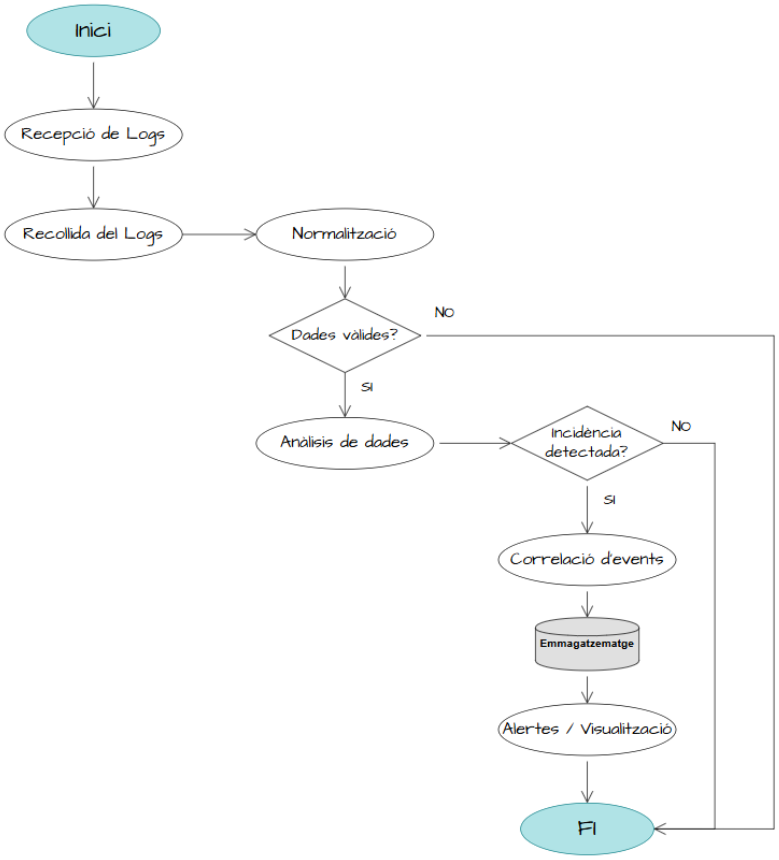


Figura 3. Diagrama de flux del sistema

1. Recepció de Logs: El sistema rep logs de múltiples fonts amb registres en diferents formats.

2. Recollida del Log: Captura i recopila els logs generats per al seu processament posterior.

3. Normalització: Conversió del log al format estàndard del sistema, independentment del format original de la font.

4. Anàlisi de Dades: Processament inicial per extraure informació rellevant i identificar patrons bàsics de seguretat.

5. Anàlisi amb IA: Aplicació d'algoritmes d'intel·ligència artificial per detectar anomalies i comportaments sospitosos.

6. Correlació d'Esdeveniments: Relacionar l'esdeveniment actual amb altres esdeveniments per identificar patrons d'atac complexos.

7. Emmagatzematge: Persistència de la informació processada i els resultats de l'anàlisi.

8. Alertes / Visualització: Generació d'alertes per a incidents crítics i presentació de la informació en dashboards per als analistes.

Condicionals:

> **Dades vàlides?** Verifica que el log conté informació completa i coherent per poder ser processat correctament.
>
> **Incidència detectada?** Decisió basada en l'anàlisi sobre si l'esdeveniment constitueix una possible amenaça de seguretat.

Relació amb els requisits identificats:

- **Automatització del processament de seguretat (RF2, RF3, RF4):** El sistema és capaç de detectar i alertar sobre incidents de seguretat de manera completament automatitzada, aplicant algoritmes d'intel·ligència artificial per analitzar els registres obtinguts.
- **Escalabilitat i adaptabilitat (RNF3):** Gràcies a que el sistema està separat en diferents mòduls amb patrons de disseny com Factory Method i Strategy, aquest pot créixer incorporant noves fonts de logs i algoritmes d'anàlisi sense afectar al funcionament general del sistema.
- **Gestió eficient de dades (RF1, RNF1, RNF2):** La integració dels components de recopilació, normalització i emmagatzematge assegura el processament eficient de grans volums de logs en temps real, garantint la disponibilitat i persistència de la informació crítica de seguretat.
- **Detecció avançada d'amenaces (RF3, RF5):** La combinació dels components d'anàlisi amb IA i correlació d'esdeveniments permet identificar tant amenaces simples com atacs complexos coordinats, oferint una capacitat de detecció superior a les solucions tradicionals basades únicament en regles.
- **Gestió unificada de logs (RF1, RF5):** L'arquitectura modular amb components Parser i Correlation ofereix un tractament unificat de logs provinents de múltiples fonts heterogènies, correlacionant esdeveniments de diferents sistemes des d'una plataforma centralitzada.
- **Flexibilitat i extensibilitat (RNF3):** Aquest disseny garanteix la compatibilitat amb diferents formats de logs i fonts de dades mitjançant els patrons Adapter i Strategy, ampliant així la cobertura del sistema per adaptar-se a diferents entorns.

3.2 Components

3.2.1 Generació de Logs

- **Descripció funcional:** Aquest component representa les múltiples fonts d'informació de seguretat que generen esdeveniments dins de l'organització. Actua com el punt d'origen de tota la informació que alimenta el sistema, proporcionant un flux d'esdeveniments de seguretat que reflecteixen l'activitat en temps real d'una infraestructura. Cada font conte informació específica del seu domini, es a dir segons el tipus de dades que recopili com pot ser un esdeveniment de xarxa.

- **Requisits específics que resol:**

  Requisits Funcionals:
  - **Recopilació de registres:** Genera els registres de seguretat en temps real que alimenten tot el sistema.

  Requisits No Funcionals:
  - **Escalabilitat:** Pot incorporar noves fonts de registres segons les necessitats.

- **Interaccions amb altres components:** Interactua directament amb el component de Recopilació de Dades on aquest rebrà la informació que genera aquest component.

3.2.2 Recopilació de Dades

- **Descripció funcional:** Component centralitzador que actua com a punt de convergència per a tots els esdeveniments generats per les diferents fonts. S'encarrega de la captura, validació inicial i preparació dels esdeveniments per al seu processament posterior, assegurant que cap informació crítica es perdi durant la transferència.

- **Requisits específics que resol:**

  Requisits Funcionals:
  - **Recopilació de registres:** Centralitza i unifica la recopilació de registres de múltiples fonts en temps real.

  Requisits No Funcionals:
  - **Rendiment:** Permet processa grans volums de dades de manera eficient.
  - **Escalabilitat:** Permet gestionar les dades mes eficientment.

- **Patrons específics del component:**
  - Buffer Pattern: Per gestionar variacions en el volum d'esdeveniments.
  - Validator Pattern: Per assegurar la qualitat de la informació rebuda.

- **Interaccions amb altres components:** Rep esdeveniments del component de Generació de Logs i els transfereix de forma controlada al component API Server, assegurant un flux ordenat i validat.

3.2.3 API Server

- **Descripció funcional:** Component que actua com a porta d'entrada i punt de control central del sistema. Gestiona totes les comunicacions entre els components externs i el nucli del sistema, proporcionant una interfície estandarditzada per a l'intercanvi d'informació i el control del flux de processament. Aquest component, a mes permet una visió a futur sobre la escalabilitat que pot tenir i les funcions de gestió que pot realitzar al sistema.

- **Requisits específics que resol:**

  Requisits Funcionals:
  - **Recopilació de registres**: Actua com a punt d'entrada controlat per als registre.

  Requisits No Funcionals:
  - **Rendiment**: Gestiona múltiples connexions simultànies de manera eficient.
  - **Disponibilitat**: Garanteix l'accés persistent als serveis del sistema.
  - **Escalabilitat**: Permet l'increment de connexions i peticions.

- **Casos d'ús principals:**
  - Recepció d'esdeveniments des de sistemes externs.
  - Consulta d'informació històrica i estadístiques.
  - Configuració de paràmetres del sistema.

- **Patrons específics del component:**
  - Facade Pattern: Simplifica l'accés als components interns del sistema.
  - Gateway Pattern: Controla i gestiona l'accés del sistema des de l'exterior.

- **Interaccions amb altres components:** Rep informació del component de Recopilació de Dades i la dirigeix cap al component de Normalització. També proporciona accés als resultats emmagatzemats per a consultes externes.

3.2.4 Normalització

- **Descripció funcional:**
  Component transformador que converteix la informació rebuda en un format estandarditzat i estructurat. Actua com a traductor, extraient la informació rellevant de cada esdeveniment i organitzant-la segons un esquema unificat que facilita el processament posterior. Aquest component, pot estar format per mes d'un tipus, es a dir , com cada tipus de font de registres, te un format diferent , s'ha de normalitzar específicament cada tipus.

- **Requisits específics que resol:**
  Requisits Funcionals:
  Recopilació de registres: Normalitza i estructura els registres per al seu processament.
  Requisits No Funcionals:
  Rendiment: Processa i transforma grans volums de dades ràpidament.
  Escalabilitat: S'adapta a diferents formats i tipus de registres.

- **Casos d'ús principals:**
  - Processament d'esdeveniments de sistemes amb formats diferents.
  - Extracció d'informació rellevant de missatges complexos.

- **Patrons específics del component:**
  - Strategy Pattern: Per seleccionar l'estratègia de processament adequada es a dir el tipus de component parser que ha d'utilitzar.
  - Template Method Pattern: Per definir el flux comú de normalització.
  - Adapter Pattern: Per adaptar formats específics al format unificat.

- **Interaccions amb altres components:** Rep informació del component API Server i proporciona esdeveniments normalitzats al component d'Anàlisi AI.

3.2.5 Anàlisi AI

- **Descripció funcional:**
  Component intel·ligent que aplica algoritmes avançats per identificar patrons anòmals i possibles amenaces. Combina múltiples tècniques d'anàlisi per proporcionar una avaluació amb diferents perspectives que puguin donar resposta sobre els riscos i amenaces.

- **Requisits específics que resol**:
  Requisits Funcionals:
  - **Detecció i alerta d'incidents**: Identifica incidents de seguretat mitjançant anàlisi intel·ligent.
  - **Anàlisi automatitzat amb IA**: Implementa tècniques d'intel·ligència artificial per detectar amenaces.
  - **Correlació d'esdeveniments**: Correlaciona esdeveniments complexos per identificar atacs sofisticats.

  Requisits No Funcionals:
  - **Rendiment**: Executa anàlisis complexos en temps real.
  - **Escalabilitat**: S'adapta a volums creixents de dades per analitzar.

- **Casos d'ús principals:**
  - Detecció d'intrusions sofisticades.
  - Identificació de comportaments anòmals.
  - Classificació automàtica de la severitat d'incidents.

- **Patrons específics del component:**
  - Observer Pattern: Per actualitzar models basant-se en nova informació.
  - Composite Pattern: Per combinar resultats de múltiples algoritmes.
  - State Pattern: Per gestionar diferents modes d'operació segons el context.

- **Interaccions amb altres components:**
  Rep esdeveniments normalitzats del component de Normalització i proporciona resultats d'anàlisi enriquits al component de Correlació.

3.2.6 Correlació

- **Descripció funcional:**
  Component analític que identifica relacions i patrons complexos entre esdeveniments aparentment independents. Permet identificar processos mes complexos d'anàlisis basant-se en mes d'una font d'informació per poder donar resposta.

- **Requisits específics que resol:**
  Requisits Funcionals:
  - **Detecció i alerta d'incidents**: Identifica incidents complexos mitjançant correlació d'esdeveniments relacionats.
  - **Correlació d'esdeveniments**: És el component principal que implementa la correlació temporal i contextual d'esdeveniments de múltiples fonts per detectar atacs avançats i campanyes coordinades.

  Requisits No Funcionals:
  - **Rendiment**: Processa correlacions en temps real mantenint un cache d'esdeveniments recents optimitzat per a consultes ràpides
  - **Escalabilitat:** Gestiona finestres de correlació adaptatives segons el tipus d'atac i s'adapta al volum creixent d'esdeveniments.

- **Casos d'ús principals:**
  - Detecció d'amenaces persistents avançades.
  - Identificació de campanyes de reconeixement i explotació.
  - Correlació d'esdeveniments entre diferents sistemes.
  - Construcció de línies temporals d'incidents complexos.

- **Patrons específics del component:**
  - Chain of Responsibility: Per processar esdeveniments a través de múltiples filtres.
  - Mediator Pattern: Per coordinar la interacció entre diferents tipus d'anàlisi.
  - Memento Pattern: Per mantenir l'estat històric necessari per a la correlació

- **Interaccions amb altres components:**
  Rep resultats d'anàlisi del component d'Anàlisi AI i proporciona informació correlacionada al component d'Emmagatzematge.

3.2.7 Emmagatzematge de Dades

- **Descripció funcional:** Component de persistència que gestiona l'emmagatzematge a llarg termini de tota la informació processada pel sistema. Proporciona accés eficient a dades històriques i manté la integritat de la informació per a anàlisis futures i requisits de compliment normatiu.

- **Requisits específics que resol:**

  Requisits Funcionals:

  - **Correlació d'esdeveniments:** Emmagatzema dades històriques necessàries per la correlació temporal.

  Requisits No Funcionals:

  - **Rendiment:** Proporciona accés ràpid a grans volums de dades emmagatzemades.
  - **Disponibilitat:** Garanteix la persistència i disponibilitat de totes les dades i alertes.
  - **Escalabilitat:** S'adapta al creixement continu de dades.

- **Casos d'ús principals:**
  - Emmagatzematge de logs processats i resultats d'anàlisi.
  - Consultes històriques per a investigacions d'incidents.
- **Patrons específics del component:**
  - Repository Pattern: Per abstraure l'accés a dades.
  - Data Access Object: Per encapsular operacions de persistència.

Interaccions amb altres components: Rep informació processada del component de Correlació i Anàlisis AI. A mes, retorna informació al sistema SIEM core per retornar els valors emmagatzemats.

4 Implementació

4.1 Arquitectura

Diagrama d'arquitectura:

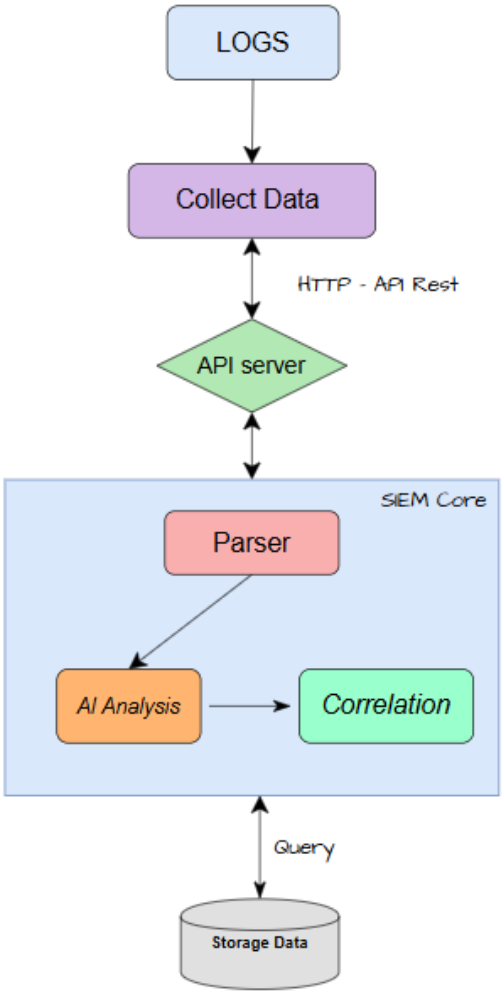


Figura 4. Diagrama d'arquitectura baix nivell

El diagrama de components descriu l'arquitectura funcional i tecnològica del sistema SIEM-AI, una plataforma avançada per a la detecció proactiva d'amenaces de ciberseguretat mitjançant intel·ligència artificial. Aquest esquema mostra com els diferents mòduls interactuen entre si, des de la recepció inicial dels logs fins a l'anàlisi, correlació i emmagatzematge final de dades.

Cada component ha estat desenvolupat amb tecnologies específiques per garantir modularitat, escalabilitat, rendiment i capacitat d'adaptació a diferents entorns. Aquesta estructura facilita tant el desplegament com el manteniment del sistema, alhora que proporciona una plataforma robusta per a la monitorització contínua i automatitzada de la seguretat.

4.1.1 Components implementats:

1. LOGS (Fonts de Dades)

Descripció: Representa les fonts externes que generen els logs de seguretat.

Tecnologies:

- PFsense: Firewall que genera logs de tràfic de xarxa.
- Suricata: Sistema IDS/IPS que detecta intrusions.
- MariaDB: Base de dades que registra activitat transaccional.
- Format: Logs en text pla, JSON, syslog segons la font.

2. Collect Data (Component de Recol·lecció)

Descripció: Client que recull fitxers de log locals i els agrupa en lots i els envia via HTTP POST al SIEM.

Tecnologies:

- Python: Llenguatge principal d'implementació.
- Requests: Llibreria per comunicació HTTP amb l'API.
- JSON: Format d'estructuració de dades per a l'enviament.

3. API Server

Descripció: Interfície REST que rep peticions externes i les dirigeix al SIEM Core.

Tecnologies:

- FastAPI: Framework web asíncron d'alt rendiment.
- Uvicorn: Servidor ASGI per executar FastAPI.

4 Parser (Normalització)

Descripció: Converteix logs de diferents formats a un esquema unificat.

Tecnologies:

- Regex : Expressions regulars per extracció de dades.
- JSON Parser: Processament de logs estructurats.

4.2 AI Analysis (Intel·ligència Artificial)

Descripció: Motor d'anàlisi intel·ligent per detectar anomalies i amenaces.

Tecnologies:

- Scikit-learn: IsolationForest, detecció d'anomalies no supervisada.
- PyTorch: Xarxa neuronal “LogClassifierNN” (64-32-16-1 neurones).

4.3 Correlation (Correlació d'Esdeveniments)

Descripció: Identifica patrons d'atac complexos enllaçant múltiples esdeveniments.

Tecnologies:

- Algoritmes basats en regles: Implementació del Cyber Kill Chain.

- Patrons temporals: Detecció de seqüències d'atac.

- Lògica heurística: Identificació d'APTs i amenaces persistents.

5. Storage Data (Emmagatzematge)

Descripció: Sistema dual d'emmagatzematge persistent i temporal.

Tecnologies:

- MongoDB: Base de dades NoSQL principal que recopila els logs i alertes.

- Memory Storage: Emmagatzematge en memòria, útil per desenvolupament i proves.

4.1.2 Relacions entre components

Collect Data - API Server:

El component Collect Data es comunica amb l'API Server mitjançant crides HTTP REST, enviant logs de seguretat agrupats en lots, i reb informació de les alertes creades. Aquesta comunicació s'implementa utilitzant el protocol HTTP POST i HTTP GET amb intercanvi de dades en format JSON.

API Server - SIEM Core:

L'API Server actua com a interfície que rep peticions HTTP externes i les dirigeix internament al SIEM Core mitjançant invocació directa de mètodes Python. S'ha optat per aquesta arquitectura encapsulada per garantir latència mínima, consistència de dades i simplicitat en el desenvolupament.

Parser - AI Analysis - Correlation:

Els components interns del SIEM Core mantenen un flux seqüencial de processament on el Parser normalitza logs i els passa directament a AI Analysis. L'AI Analysis processa cada log amb algoritmes de machine learning i deep learning, on comparteix els resultats amb Correlation per a l'anàlisi de patrons complexos. Aquesta arquitectura interna s'ha dissenyat per maximitzar l'eficiència computational i mantenir coherència en l'estat de les dades durant tot el procés d'anàlisi intel·ligent.

SIEM Core - Storage Data

El SIEM Core gestiona totes les operacions d'emmagatzematge i consulta mitjançant una connexió dual que combina MongoDB com a motor principal i emmagatzematge en memòria. S'implementa utilitzant PyMongo per a operacions CRUD sobre col·leccions especialitzades amb indexació temporal optimitzada. El emmagatzematge en memòria es utilitzat per entorns de prova i anàlisis del sistema.

4.2.3 Flux general del sistema SIEM-AI:

1. Generació de Logs:

Els sistemes de la infraestructura generen logs de seguretat de manera continuada en els seus formats natius. Cada font utilitza el seu propi esquema i estructura de dades segons la seva tecnologia específica. Els logs es guarden automàticament en fitxers locals del sistema amb dades temporals i metadades pròpies de cada servei. Aquests logs contenen informació variada sobre esdeveniments de xarxa, autenticació, accés a recursos i activitat del sistema, amb diferents nivells de detall i formats de representació.

2. Recol·lecció i Agrupació:

El component Collect Data escaneja els directoris configurats buscant nous logs per processar. Llegeix el contingut dels fitxers detectats i identifica el tipus de font basant-se en l'extensió, ubicació o patrons de contingut. Per optimitzar l'eficiència de la transmissió, agrupa múltiples logs en lots de mida configurable. Afegeix metadades de processament com camins de fitxer, números de línia i marques temporals d'ingesta, preparant els lots per a l'enviament en format JSON estructurat.

3. Transmissió via API REST:

Els lots de logs s'envien al sistema central mitjançant peticions HTTP POST a l'API Server. Aquest component, implementat amb FastAPI, rep les peticions i realitza una validació automàtica de l'estructura i format de les dades utilitzant esquemes predefinits. Un cop validats, els logs es processen individualment, extraient-los del lot i dirigint-los al SIEM Core mitjançant invocació directa de mètodes Python. L'API Server gestiona la comunicació asíncrona permetent rebre múltiples fonts simultàniament sense bloqueig.

4. Normalització Adaptativa:

El SIEM Core identifica el tipus de font de cada log i selecciona el parser específic corresponent. Cada parser aplicar lògica especialitzada per extreure informació rellevant del format original, convertint-la a l'esquema unificat del sistema. Aquest procés inclou la normalització de formats de data, estandarditzación de noms de camps, conversió de valors categòrics i assignació de nivells de severitat coherents.

5. Anàlisi Intel·ligent:

El log normalitzat es converteix en un vector numèric de característiques representatives per a l'anàlisi d'intel·ligència artificial. Aquest vector es processa simultàniament per múltiples models analítics, un algoritme de detecció d'anomalies identifiquen desviacions del comportament normal i un altre de classificació avaluen la probabilitat d'amenaça basant-se en patrons apresos. Els resultats de tots els models s'integren mitjançant lògica de fusió per generar una puntuació de risc unificada acompanyada d'indicadors de confiança i factors de risc identificats.

6. Correlació Contextual:

El component de Correlació analitza les relacions entre l'esdeveniment actual i altres esdeveniments recents dins de finestres temporals. Aplica regles basades en patrons coneguts d'atac i models de comportament maliciós per identificar seqüències, cadenes d'esdeveniments i indicadors de compromís complexos. Avalua factors contextuals com adreces IP relacionades, usuaris implicats, actius afectats i progressió temporal per determinar si els esdeveniments formen part d'activitats coordinades o amenaces persistents.

7: Generació d'Alertes

El sistema evalua els resultats combinats de l'anàlisi d'IA i la correlació per determinar si es requereix generar una alerta. Aquesta decisió es basa en llindar de puntuació de risc, nivells de confiança, severitat dels esdeveniments i patrons detectats. Quan es decideix alertar, es genera una alerta estructurada que inclou informació descriptiva, context de l'amenaça, actius afectats, evidències recollides i recomanacions d'acció.

8: Persistència

Totes les dades generades durant el procés s'emmagatzemen de manera persistent utilitzant un sistema dual d'emmagatzematge. Els logs normalitzats, resultats d'anàlisi, informació de correlació i alertes generades es distribueixen en col·leccions especialitzades amb indexació optimitzada per tipus de consulta esperats. S'actualitzen estadístiques agregades i mètriques del sistema per facilitar reporting i anàlisi de tendències. La informació es vincula mitjançant identificadors únics permetent traçabilitat completa des de logs originals fins a alertes finals.

9: Consultes

Un cop processades i emmagatzemades, totes les dades estan disponibles immediatament a través de consulta l'API Server. Els usuaris poden accedir a logs, alertes, estadístiques i resultats d'anàlisi mitjançant peticions REST. El sistema proporciona respostes optimitzades gràcies a la indexació de base de dades i permet consultes complexes per facilitar investigacions, anàlisi forense i reporting operacional.

4.2 Components

4.2.1 Generació de Logs

Descripció funcional: Aquest component fonamental actua com la font primària de les dades de seguretat processades pel sistema SIEM. Engloba una diversitat de sistemes, aplicacions, dispositius de xarxa i serveis desplegats en un entorn operatiu real o simulat, cadascun dels quals té la capacitat de generar registres d'esdeveniments. Aquests logs són la matèria primera essencial per a qualsevol anàlisi de seguretat i per a la detecció proactiva d'amenaces i anomalies. Conseqüentment, la qualitat, el nivell de detall i l'abast dels logs produïts per aquestes fonts tenen un impacte directe i significatiu en l'eficàcia i la precisió del sistema SIEM per identificar activitats malicioses o comportaments fora del normal. Actualment, el sistema por analitzar dades provinents de Firewall, IDS/IPS i Servidor de base de dades.

Tecnologies utilitzades: Aquest component representa els sistemes externs al SIEM que produeixen les dades de seguretat. Per tant, les tecnologies utilitzades per a aquesta funció específica no formen part de sistema implementat. Però, podem dir que principalment utilitza els tipus de logs provinent de les tecnologies PFsense, Suricata i MariaDB per obtenir aquestes dades.

Problemes trobats:

1. Problema: La construcció d'un entorn empresarial que repliqui la complexitat d'un escenari real mitjançant eines de virtualització amb GNS3 va resultar ser un procés llarg i propens a errors de configuració. Això va endarrerir molt les fases inicials del projecte dedicades a l'obtenció de dades de prova fiables.

> Solució: La solució proporcionada, va ser la creació del logs reals manualment per poder ingerir les dades al sistema.

4.2.2 Collect data (Client)

Descripció Funcional: Aquest component actua com un client que recull fitxers de log emmagatzemats localment i els envia al SIEM centralitzat a través del component API. La seva funció principal és llegir grans volums de dades de log de diverses fonts simulades, organitzar-les de manera eficient i enviar-les de forma fiable a l'endpoint d'ingesta del SIEM.

Llegeix fitxer a fitxer els logs d'un directori especificat, identificant el tipus de font per gestionar grans quantitats de dades de manera eficient i no sobrecarregar l'API. A mes, agrupa les línies de logs en lots en format JSON i s'envien a l'API del SIEM mitjançant peticions HTTP POST. També inclou funcionalitats per reiniciar les dades del SIEM abans d'enviar nous logs i per obtenir i mostrar un resum dels resultats de l'anàlisi un cop finalitzat l'enviament.

Tecnologies Utilitzades:

Requests: Llibreria HTTP de Python per a la comunicació amb l'API del SIEM.

> **Funcionament**: S'utilitza per construir i enviar peticions POST amb els lots de logs en format JSON per a que el sistema rebi els logs eficientment. També s'utilitza per les peticions GET a endpoints d'estat o anàlisi.

Justificació: És molt fàcil d'usar, robusta i gestiona automàticament molts detalls complexos per la gestió de connexions.

Concurrent.futures: Una llibreria de Python per executar crides asíncronament mitjançants fils.

Funcionament: S'utilitza per enviar múltiples lots de logs a l'API simultàniament, fins a un nombre màxim configurat. Això permet que mentre un fil espera la resposta de l'API per un lot, altres fils poden estar enviant altres lots, millorant el rendiment total de l'enviament.

Justificació: Permet implementar concurrència de manera relativament senzilla per accelerar el procés d'enviament sense necessitat d'una programació asíncrona més complexa (amb `asyncio`), que podria ser un excés per a aquesta tasca concreta.

Problemes d'Implementació i Solucions:

1. Problema: Processar i enviar milers o milions de línies de log d'un cop pot ser ineficient i causar problemes de memòria o timeouts.

Solució: S'implementa l'enviament en lots on els logs es llegeixen i s'agrupen en llistes més petites abans de ser enviats. La mida del lot s'intenta optimitzar dinàmicament segons el volum total de logs. A més, s'envien simultàniament per optimitzar el temps d'enviament i execució al sistema.

2. Probelma: peticions HTTP poden fallar degut a problemes de xarxa, timeouts o errors temporals a l'API del SIEM.

Solució: S'implementa una lògica de reintents automàtics si una petició de lot falla, el script intenta enviar-la de nou un nombre predefinit de vegades.

3. Problema: Els logs poden estar en format text simple, JSON, o tenir problemes de codificació de caràcters.

Solució: Es detecta el tipus de fitxer basant-se en la seva extensió per carregar-lo i processar-lo adequadament.

4.2.3 API Server

Descripcio funcional: Aquest component serveix com la porta d'entrada principal al sistema SIEM. Actua com un backend que processa les peticions entrants i coordina les operacions amb el nucli del SIEM per poder dirigir el sistema. La seva funció és doble, rebre dades de diverses fonts externes i exposar endpoints perquè altres sistemes o una interfície d'usuari puguin consultar els resultats de l'anàlisi:

1. Recepció de Dades de Logs: Rep els logs generats de diverses fonts externes mitjançant peticions HTTP/HTTPS POST, suportant l'enviament de logs individuals o en lots per optimitzar l'enviament.

2. Exposició d'endpoints per a consultes: Ofereix una interfície API REST perquè altres sistemes, aplicacions o una possible interfície d'usuari puguin consultar les dades, mes enfocat a la escalabilitat del sistema y integració amb altres eines.

Endpoints principals del component:

"GET / " Verifica que el servei API està en execució.
"POST /api/logs/batch" Rep i ingereix un lot de logs de seguretat de manera eficient.

“GET /api/alerts” Consulta alertes generades amb filtres opcionals i paginació.
“GET /api/stats” Obté estadístiques agregades dels logs i alertes.
“GET /api/analysis” Consulta un resum general de lanàlisi de seguretat del sistema.

Tecnologies utilitzades:

FastAPI: És un framework web modern i d'alt rendiment per a la construcció d'APIs basades en estàndards oberts.

Funcionament: Defineix els endpoints de l'API amb rutes URL com “/api/logs”, “/api/alerts”, mitjançants peticions HTTP (GET, POST). Realitza validació automàtica de les dades entrants basant-se en tipus de dades rebudes i gestiona el maneig bàsic d'errors. Proporciona suport natiu per a un funcionament asíncron, permetent rebre diferents fonts a l’hora sense problema.

Justificació: Es va escollir principalment pel seu rendiment excepcional a comparació d’altres eines similars. A mes, el suport asíncron que permet es molt ultil per poder projectar el desenvolupament a producció.

Alternatives:

-Flask: És un framework web més lleuger i minimalista, que permet realitzar peticions web fàcilment. Aquesta solució, obté menys rendiment per a càrregues asíncrones, cosa que perjudica la seva elecció.

-Django REST Framework: Un framework més complet orientat a aplicacions web complexes i APIs REST. Ofereix moltes funcionalitats integrades pot ser més complex per a una API pura i pot no igualar el rendiment de FastAPI en càrregues asíncrones.

Uvicorn: Servidor d'aplicacions ASGI (Asynchronous Server Gateway Interface) d'alt rendiment per a Python.

Funcionament: Es l’encarregat d’executar la tecnologia de FastAPI, gestionant les connexions de xarxa entrants i dirigint les peticions a l'aplicació FastAPI. És optimitzat per a aplicacions asíncrones, permetent gestionar un gran nombre de connexions concurrents de manera eficient.

```
# Executar uvicorn amb configuració optimitzada
uvicorn.run(
    app,
    host=args.host,
    port=args.port,
    log_level="info",          # Mantenir info per veure peticions
    access_log=True,           # Mantenir logs d'accés per monitorejar
    timeout_keep_alive=120,    # Timeout per connexions persistentes
    limit_max_requests=10000,  #Límit de peticions
    backlog=2048               # Backlog per gestionar més connexions concurrents
)
```

Figura 5. Codi de configuració uvicorn

Problemes trobats:

1. Problema: Assegurar un alt rendiment en la ingesta de logs, especialment quan es reben lots de logs simultàniament des de múltiples fonts. La recepció d'un gran volum de peticions POST podria saturar el servidor.

> **Solució Aportada:** L'elecció de FastAPI i Uvicorn, amb el seu suport natiu per a programació asíncrona, permet que el servidor API gestioni moltes connexions concurrents de manera eficient sense bloquejar els processos.

2. Problema: Validar que les dades rebudes en les peticions d'ingesta de logs tinguin el format i els camps esperats.

> **Solució Aportada:** Aprofitament de la validació automàtica de dades de FastAPI basada en tipus. En definir els models de dades esperats per als cossos de les peticions, FastAPI valida automàticament les dades entrants.

3. Problema: Assegurar una integració eficient i un estat coherent entre l'API Server i el nucli del SIEM.

> **Solució Aportada:** L'API Server interactua directament amb una instància única del nucli del SIEM. Les operacions de l'API criden directament els mètodes corresponents de la instància del SIEM per dirigir els processos.

4.2.4 Normalització (Parser)

Descripció funcional: La seva funció principal és rebre els logs amb formats diferents i transformar-los en un format de dades unificat i estandarditzat del propi sistema. Aquesta transformació implica l'extracció d'informació clau de cada log i el mapeig als camps predefinits del format unificat del sistema, el qual serà el format JSON.

Dins de cada registre a analitzar, acostuma a trobar-se molts tipus de dades, en diferents formats segons l'origen, com poden ser ports, direccions IP, protocols de xarxa i molt mes. Per aquest motiu, l'estandardització de les dades és indispensable per permetre que els components posteriors puguin processar les dades independentment de la seva font d'origen.

La implementació del component de normalització segueix un patró de disseny basat en classes on existeix una classe base "BaseParser" amb funcionalitats comunes a tots els parsers, i classes filles específiques per a cada tipus de font de log (PFSenseParser, SuricataParser...). Cada classe filla implementa la lògica de parsing específica per al format de la seva font, això permet modularitat dins del sistema i facilita l'escalabilitat per afegir noves fonts de logs en el futur, ja que només cal crear una nova classe filla heretant de la base i implementant el mètode de parsing específic.

Aquest component, actua en diferents fases per poder donar solució:

1. Identificació del format de log:

   Un cop el component anterior ha enviat en fomat d'origen els registres, el primer pas del component de Normalització és identificar el seu format d'origen. Això es pot fer basant-se en metadades proporcionades per la font i mitjançant tècniques d'inspecció del contingut del log per reconèixer patrons associats als formats coneguts.

2. Selecció i aplicació del parser adequat:

Un cop identificat el format, el component de Normalització selecciona el parser específic dissenyat per a aquesta estructura particular del log. Això vol dir que, per cada origen de dades amb un format diferent com per exemple un log JSON generat per Suricata o un missatge syslog procedent de pfSense, s'utilitzarà un parser dedicat i adaptat per cadascun.

Un cop escollit el parser corresponent, aquest analitza la cadena de text del log i extreu la informació rellevant en els camps estructurats del fomat unificat.

3. Mapeig a l'Esquema Unificat:

   Després d'extreure els camps del log sense format mitjançant el parser, es realitza un mapeig d'aquests camps als camps corresponents d'un esquema de dades unificat i estandarditzat definit per al sistema SIEM. Aquest esquema comú garanteix que tots els logs, independentment del seu origen, tinguin una estructura de dades consistent i camps amb noms i tipus de dades predefinits.

4. Enriquiment de Dades:

   Abans de finalitzar la normalització, es pot afegir informació addicional al log normalitzat que no estava present en el log original però que pot ser rellevant per a l'anàlisi com un identificador únic.

Tecnologies utilitzades:

Regex ("re"): Per analitzar els formats de log entrants i extreure la informació rellevant, s'implementen funcions de parsing a mida utilitzant manipulació de strings i expressions regulars.

Funcionament: Les expressions regulars s'apliquen sobre logs de text pla per identificar i extreure patrons específics com adreces IP, ports, timestamps i missatges d'error. Cada parser utilitza conjunts de regex especialitzats per al seu tipus de log específic, permetent l'extracció precisa de camps rellevants fins i tot de formats no estructurats.

Justificació: Les expressions regulars són una eina potent i estandarditzada per a l'extracció de dades de text, oferint flexibilitat per gestionar variacions de format i robustesa davant canvis menors en l'estructura dels logs.

Llibreria JSON: S'utilitza per a la lectura i interpretació de logs que ja venen en format JSON. Permet accedir directament als camps estructurats del log d'origen i extreure fàcilment les dades importants.

Funcionament: Per a logs ja estructurats en format JSON, la llibreria permet parsejar directament l'estructura de dades i accedir als camps específics mitjançant claus definides.

Justificació: Permet processar eficientment els logs que ja tenen una estructura definida en JSON sense necessitat de parsing basat en text, reduint la complexitat i augmentant la fiabilitat de l'extracció de dades.

Problemes trobats:

1. Problema: La gran diversitat de formats de logs existents i la necessitat de crear i mantenir parsers específics per a cadascun.

Solució Aportada: Adopció d'una arquitectura de parsing modular i extensible basada en classes. La classe base proporciona una estructura comuna i classes filles per a cada font aïlla la lògica de parsing específica.

2. Problema: Anar més enllà de la simple extracció sintàctica de camps i extreure el significat d'un esdeveniment.

Solució Aportada: Incorporació de lògica addicional dins de les classes de parser, per interpretar el contingut textual dels logs i mapejar-lo correctament a camps estandarditzats com “event_type” o “event_description” amb major precisió semàntica.

3. Problema: Els logs poden estar corruptes o tenir un format inesperat cosa que pot fer fallar el parser, pero es important guardar i mantenir aquest tipues d’errors degut a possible esdeveniments maliciosos ideats amb aquest objectius.

Solució Aportada: En cas d'error durant el parsing detallat, s'assegura que almenys es preservi el log original, s'assignin camps bàsics i es generi un esdeveniment de tipus “parsing_error” amb una descripció de l'error. Això evita la pèrdua total de la informació del log original i permet la seva posterior anàlisi manual si cal.

4.2.5 Anàlisi AI

Descripció funcional:

El component d'Anàlisi d'Intel·ligència Artificial constitueix el nucli analític avançat del sistema SIEM, encarregat de la detecció proactiva d'amenaces de seguretat mitjançant l'aplicació de tècniques d'aprenentatge automàtic. Aquest component processa els logs normalitzats procedents de diverses fonts per identificar patrons d'activitat anòmala o maliciosa que podrien evadir els mecanismes de detecció basats en regles tradicionals. El seu objectiu és proporcionar una avaluació de risc intel·ligent per a cada esdeveniment, classificar el tipus d'amenaça.

El sistema utilitza una estratègia híbrida, combinant diferents tipus d'algoritmes per augmentar la precisió i la capacitat de detecció:

1. **Algoritmes de Detecció d'Anomalies (No Supervisats):** Aquests algoritme es centra en identificar esdeveniments que s'aparten significativament del comportament normal dins del sistema. Llavors si detecta events no habituals augmenta el risc de ser un atac, on una puntuació alta indica que el log és molt diferent del comportament normal observat pel model, suggerint que podria ser sospitós.
2. **Algoritmes de Classificació (Supervisats):** Per un altre costat, a diferència del model d'anomalies, el algoritme de classificació analitza els conjunts de dades que ja han estat etiquetats, i aprenen els patrons associats a cada valor per poder classificar nous logs.

   La seva funció és determinar, basant-se en els patrons apresos, si el log encaixa amb algun tipus d'amenaça coneguda o si és un esdeveniment normal.

Puntuació de Risc: El sistema SIEM combina els resultats d'aquests diferents tipus d'algoritmes per obtenir una visió més completa i fiable:

- **Generació de Puntuacions Múltiples:** Cada model retorna el seu propi resultat, una puntuació d'anomalia del model no supervisat i una o més prediccions de classe amb les seves probabilitats associades del model supervisat.
- **Puntuació Final:** Totes les puntuacions i classificacions individuals apliquen una lògica addicional per ponderar o combinar aquests resultats. On mitjançant la combinació de els resultats s’obté un numero del 0 al 1 indicant la probabilitat del resultat.
- **Classificació Final:** A partir de la combinació de resultats, el sistema assigna una classificació final a l'esdeveniment i una puntuació de risc global. Aquesta puntuació és

un valor numèric que representa la probabilitat d'amenaça detectada en aquest log particular.

Tecnologies utilitzades:

Scikit-learn (Machine Learning): Aquesta tecnologia constitueix el nucli dels algoritmes d'aprenentatge automàtic clàssic dins del component d'anàlisi d'intel·ligència artificial. Aquesta biblioteca proporciona una implementació robusta i optimitzada de l'algoritme Isolation Forest, que és l'encarregat de la detecció d'anomalies no supervisada.

Funcionament:

La seva funció dins del sistema comença quan un log normalitzat arriba al component AI i es converteix en un vector numèric de característiques mitjançant l'extracció de paràmetres rellevants com adreces IP, ports, protocols, marques temporals i altres indicadors.

Aquest vector, que conté aproximadament 20 característiques numèriques, necessita ser preprocessat abans de ser analitzat. Mitjançant la seva classe StandardScaler, s'encarrega de normalitzar i estandarditzar les característiques a un format numèric per poder tractar aquestes dades, on transforma cada característica per obtenir una mitjana zero i una variància unitària.

Un cop les dades estan adequadament preprocessades, el vector normalitzat es passa a l'objecte IsolationForest de scikit-learn, on construeix una col·lecció d'arbres de decisió aleatòria que intenten aïllar cada punt de les dades obtingudes a partir dels logs.

El funcionament intern de l'algoritme Isolation Forest es basa en el concepte on les anomalies són valors poc freqüents dins d'un conjunt de dades i per tant són més fàcils d'aïllar que els valors normals. Cada arbre construït selecciona aleatòriament una característica i un valor de tall, dividint l'espai en regions cada vegada més petites. El nombre de divisions necessàries per aïllar un punt és una mesura de com d'estrany és. Això vol dir que si un log es pot aïllar amb molt poques divisions, és probable que sigui una anomalia, en canvi si requereix moltes divisions, el comportament és similar al de la majoria i es considera normal.

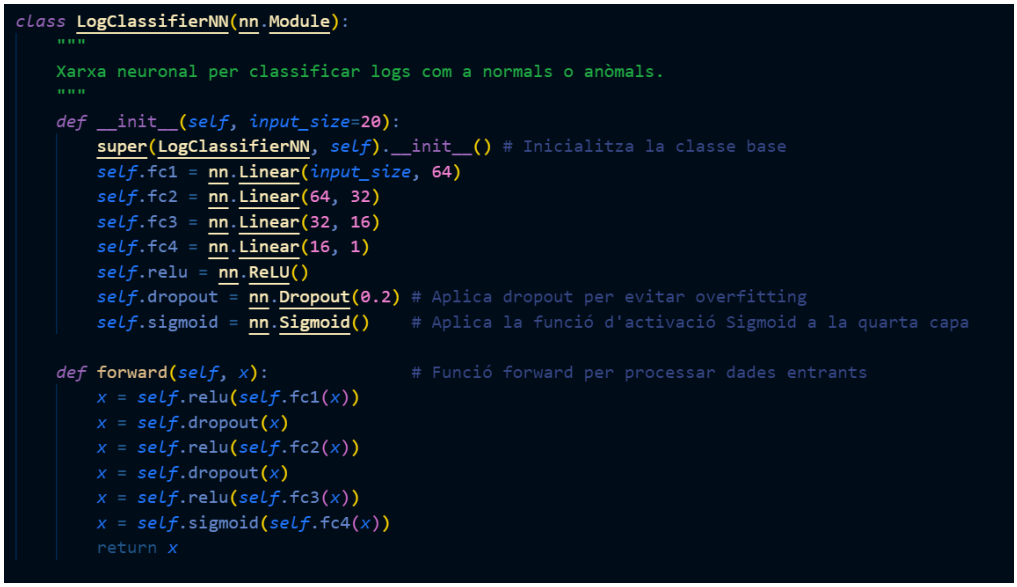
Justificació: Isolation Forest és un algoritme molt eficient i escalable per a la detecció d'anomalies en grans conjunts de dades d'alta dimensionalitat. És ràpid tant en l'entrenament com en la predicció sobre nous logs.

Alternatives:

-Autoencoders: Xarxes neuronals que detecten anomalies si no poden reconstruir bé les dades. No escollida per la seva complexitat i temps d'entrenament molt més llarg, menys pràctic per a l'anàlisi en temps real.

PyTorch (Deep Learning):

PyTorch, és un framework, que proporciona la infraestructura necessària per al desenvolupament i execució de models de xarxes neuronals dins del component d'intel·ligència artificial del sistema. Aquesta eina permet implementar solucions d'aprenentatge profund per a detectar amenaces. El model utilitzat, s'anomena LogClassifierNN, és una xarxa neuronal de tipus feedforward completament connectada, composta per quatre capes lineals amb 64, 32, 16 i 1 neurones respectivament.



```python
class LogClassifierNN(nn.Module):
    """
    Xarxa neuronal per classificar logs com a normals o anòmals.
    """
    def __init__(self, input_size=20):
        super(LogClassifierNN, self).__init__() # Inicialitza la classe base
        self.fc1 = nn.Linear(input_size, 64)
        self.fc2 = nn.Linear(64, 32)
        self.fc3 = nn.Linear(32, 16)
        self.fc4 = nn.Linear(16, 1)
        self.relu = nn.ReLU()
        self.dropout = nn.Dropout(0.2) # Aplica dropout per evitar overfitting
        self.sigmoid = nn.Sigmoid()    # Aplica la funció d'activació Sigmoid a la quarta capa

    def forward(self, x):              # Funció forward per processar dades entrants
        x = self.relu(self.fc1(x))
        x = self.dropout(x)
        x = self.relu(self.fc2(x))
        x = self.dropout(x)
        x = self.relu(self.fc3(x))
        x = self.sigmoid(self.fc4(x))
        return x
```

Figura 6. Codi del funcionament de LogClassifierNN

Funcionament: Durant la fase de predicció, cada nou log que arriba al sistema segueix un procés seqüencial específic, on inicialment el log és transforma en un vector numèric de 20 característiques per assegurar que totes les característiques puguin analitzar-se correctament. Quan aquest vector de 20 posicions, s'introdueix a la primera capa de 64 neurones, cada neurona calcula una combinació ponderada de les característiques d'entrada. Amb el producte de cada característica multiplicat per un pes específic, après durant l'entrenament, obtenim el resultat individual de cada neurona. A continuació, es sumen tots aquests productes resultants obtenint un valor total de ponderació a les 64 neurones.

Els resultats passen immediatament per una funció d'activació ReLU (Rectified Linear Unit), que és una operació matemàtica que manté els valors positius tal com són i converteix tots els valors negatius a zero (ReLU(x) = max(0, x)). La importància d'aquesta funció és que introdueix no-linealitat a la xarxa neuronal. Sense ReLU, la xarxa només podria aprendre relacions lineals simples entre les característiques com per exemple si el port és alt, llavors te més risc , cosa que és molt limitada per a la detecció de seguretat. En canvi, la xarxa pot aprendre patrons complexos i condicionals múltiples, identificant mes d'un valor a l'hora.

Per un altre costat, dins d'aquest sistema intel·ligent, podem tindre un error important, el overfitting. Aquest problema consisteix en que la xarxa neuronal aprèn massa específicament les dades d'entrenament, memoritzant els exemples particulars en lloc d'aprendre patrons generals aplicables a nous logs. Això fa que el model funcioni molt bé amb les dades amb què s'ha entrenat, però tingui un rendiment pobre quan ha d'analitzar logs nous i reals que no ha vist abans.

Per poder minimitzar aquest error, durant l'entrenament, també s'aplica una capa de dropout amb probabilitat del 20% que desactiva aleatòriament certes neurones. Aquesta funcionalitat s'implementa mitjançant la classe nn.Dropout(0.2) de PyTorch, que s'aplica després de les funcions d'activació self.relu() en les capes ocultes durant la fase d'entrenament. La funció self.dropout(x) selecciona aleatòriament el 20% de les neurones de la capa i les posa a zero, mentre que durant la inferència (predicció), el mètode model.eval() desactiva automàticament el dropout. Aquesta tècnica força la xarxa a no dependre massa de neurones específiques, obligant-la a aprendre múltiples camins diferents per arribar a la mateixa conclusió.

Els valors processats es transfereixen a la segona capa linear de 32 neurones, on cada neurona calcula una nova combinació ponderada dels 64 valors de la capa anterior, començant a combinar i abstraure les característiques detectades per formar representacions més complexes

dels patrons de seguretat. Aquest procés es repeteix amb tots els grups de neurones amb l'aplicació de ReLU i dropout fins a arribar a la neurona 1.

Finalment, aquests valors passa per una funció d'activació Sigmoid que transforma qualsevol entrada numèrica en un rang comprès entre 0 i 1, on la sortida interpreta la probabilitat que el log sigui una activitat maliciosa sent lo mes proper a 1 el mes probable.

Justificació: Les xarxes neuronals ofereixen una capacitat superior per identificar patrons complexos i no lineals, essencials per a la detecció d'amenaces avançades en logs de seguretat. PyTorch proporciona l'entorn ideal per desenvolupar models adaptats a aquest propòsit, amb eficiència, flexibilitat i compatibilitat amb sistemes de processament accelerat, permetent la detecció d'atacs sofisticats.

Problemes trobats:

1. Problema: La conversió eficient de dades de log semi-estructurades o textuals a un format numèric d'entrada adequat per a models intel·ligents, preservant la informació rellevant per a la seguretat.

> **Solució Aportada**: Creació d'un vector de 20 característiques numèriques basades en atributs normalitzats i l'ús de tècniques com la codificació numèrica i per fer les dades processables als algoritmes.

2. Problema: La necessitat de detectar tant anomalies desconegudes com patrons d'atac coneguts sense dependre exclusivament de signatures que ràpidament queden obsoletes.

> **Solució Aportada:** Isolation Forest proporciona la capacitat de detectar anomalies sense signatures identificant qualsevol desviació del comportament normal, mentre que la Xarxa Neuronal s'entrena per reconèixer patrons complexos d'atacs coneguts i simulats, proporcionant una detecció més robusta contra atacs freqüents.

3. Problema: La gestió de l'aprenentatge continu i la persistència de models en un entorn de producció per millorar la precisió al llarg del temps sense requerir intervenció manual constant.

> **Solució Aportada:** Implementació d'un mecanisme de buffer per acumular dades de logs processades amb retroalimentació (si disponible) i un sistema de reentrenament periòdic o basat en un llindar de noves dades. La utilització de joblib i les funcions de persistència de PyTorch permet guardar els models entrenats a disc, assegurant que l'aprenentatge es manté entre execucions del sistema.

4. Problema: La integració dels resultats de models estadístics/probabilístics diversos en una única mètrica de risc coherent i comprensible.

> **Solució Aportada:** Desenvolupament d'una lògica de càlcul de puntuació composta que combina els resultats d'Isolation Forest i la Xarxa Neuronal amb pesos configurables. Aquesta puntuació unificada simplifica la interpretació per als components de correlació i generació d'alertes, permetent una priorització clara basada en el risc.

4.2.6 Correlació

Descripció funcional: El component de Correlació és una capa d'anàlisi avançada dins del sistema SIEM que té com a finalitat principal identificar i agrupar esdeveniments de seguretat que no són detectables mitjançant l'anàlisi individual de cada log o la detecció d'anomalies aïllades. Aquest component, analitza les relacions temporals, contextuals i d'atributs comuns entre múltiples esdeveniments per construir una correlació d'esdeveniments. La correlació permet als analistes de seguretat comprendre la narrativa completa d'un atac, identificar les seves diferents fases i prioritzar la resposta en funció de la gravetat dels esdeveniments correlacionats.

El component implementa principalment un enfocament basat en algoritmes de regles i patrons predefinits, codificats directament en Python. No utilitza algoritmes d'aprenentatge automàtic per a la correlació, sinó que es aprofita els resultats generats per l'AI.

Les funcions i lògiques clau implementades inclouen:

Regles de Concordança Basades en Atributs: Funcions que examinen un flux de logs normalitzats buscant esdeveniments que comparteixin atributs comuns com:

- Adreces IP origen i destí: Per detectar activitat sospitosa de la mateixa font o cap al mateix objectiu.
- Noms d'usuari: Per identificar activitats de compte inusuals o compromeses.
- Ports i Protocols: Per vincular esdeveniments de xarxa.
- Detecció de Seqüències i Comportaments Anòmals: S'implementen patrons que defineixen una sèrie d'esdeveniments que han de succeir en un ordre o combinació específica per activar una correlació. Per exemple:

    MultipleFailedLogins_followed_by_Success: Busca repetits intents de login fallits des de la mateixa IP o usuari, seguits ràpidament per un login exitós des de la mateixa entitat.

    PortScan_then_ExploitAttempt: Identifica un esdeveniment de reconeixement seguit d'un intent d'explotació o accés inusual.

- Agregació: Funcions que agrupen esdeveniments similars dins d'una finestra de temps per crear una única alerta, evitant alertes repetitives o redundants.

Tecnologies utilitzades:

Libreries "defaultdict" i "counter": Es fan servir per a l'agrupació eficient d'esdeveniments i l'agregació d'informació. Defaultdict és ideal per agrupar esdeveniments per un atribut clau (com l'adreça IP d'origen) de manera dinàmica, mentre que counter facilita el recompte d'ocurrències de tipus d'esdeveniments o altres atributs dins d'un grup correlacionat.

> **Justificació**: Són natives de Python i ofereixen un bon equilibri entre rendiment i facilitat d'ús per a la manipulació d'esdeveniments.

Algoritmes basats en regles i heurístiques de detecció de patrons: La lògica central del component de correlació es basa en algorismes programats per identificar combinacions i seqüències d'esdeveniments específiques que corresponen a escenaris d'atac coneguts. Aquests algorismes implementen models com el Cyber Kill Chain (netskope, n.d) per analitzar si

una col·lecció d'esdeveniments d'una mateixa entitat mostra evidència de progressió a través de les diferents fases d'un atac.

> **Justificació**: Tot i l'ús de ML en el component AI, la correlació de múltiples esdeveniments sovint requereix l'aplicació de coneixement expert i models de seguretat predefinits. Una implementació basada en regles flexibles i heurístiques permet codificar aquest coneixement de domini.

Problemes trobats:

1. **Problema:** El gran volum de logs i alertes individuals de baixa prioritat dificultava la identificació eficient dels esdeveniments realment relacionats que formen part d'un atac.

   > Solució: Implementació d'una estratègia d'agrupació inicial eficient basada en atributs d'alta cardinalitat i significat en seguretat (principalment l'adreça IP d'origen o destí, i potencialment usuaris o noms d'actius) i l'aplicació de finestres temporals definides. Només els esdeveniments dins d'un mateix grup i període de temps són analitzats per a una possible correlació, reduint dràsticament l'espai de cerca.

2. **Problema:** Modelar la complexitat i la variabilitat dels atacs cibernètics moderns, que sovint no segueixen una seqüència lineal predictible d'esdeveniments.

   > Solució: En lloc de basar la correlació en seqüències estrictes d'esdeveniments en un ordre cronològic fix, la lògica de detecció de patrons es va centrar en la presència de conjunts d'indicadors que representen les diferents fases del Cyber Kill Chain.

4.2.7 Component: Emmagatzematge

Descripció Funcional: Aquest component s'encarrega d'emmagatzemar de manera persistent els logs normalitzats, els resultats de l'anàlisi i les alertes generades. Un emmagatzematge eficient és crucial per a permetre cerques d'incidents i possible anàlisis forenses. El component gestiona la connexió amb la base de dades, optimitza les operacions d'escriptura i lectura, i assegura la integritat i la disponibilitat de les dades a llarg termini.

Les seves funcions principals inclouen: l'emmagatzematge de logs en el format unificat establert pel component de normalització, la persistència dels resultats d'anàlisi d'IA (puntuacions d'anomalia, classificacions), l'emmagatzematge de troballs de correlació i alertes d'alt nivell, la gestió d'índexs per optimitzar consultes, i la implementació de polítiques de retenció de dades. També proporciona interfícies eficients per a consultes complexes que permeten als components d'anàlisi i correlació accedir ràpidament a dades històriques.

Tecnologies Utilitzades:

MongoDB: Base de dades NoSQL orientada a documents utilitzada com a motor d'emmagatzematge principal.

> **Funcionament:** MongoDB actua com a base de dades central del SIEM, aquest emmagatzema els logs en el seu format JSON definit al sistema. Les col·leccions s'organitzen de manera organitzada, amb una col·lecció principal per a logs, altra per a resultats d'anàlisi, i una més per a correlacions.

Justificació: MongoDB és ideal per a logs perquè suporta esquemes flexibles, ofereix un rendiment excel·lent per a operacions d'inserció massiva, i proporciona capacitats de consulta potents per a anàlisis complexes. La seva capacitat per indexar camps específics permet cerques ràpides per IP, usuari, temps o qualsevol altre atribut rellevant.

Alternatives:

- Elasticsearch: Optimitzat específicament per a cerques i anàlisis de logs, però amb major complexitat de configuració i recursos. Aquesta tecnologia també implementa mongoDB dins dels seus requeriments, però esta mes enfocada a producció d'un entorn mes complex.

Memory Storage: Sistema d'emmagatzematge alternatiu implementat com a solució quan MongoDB no està disponible.

Funcionament: Emmagatzemar temporalment logs, alertes i estadístiques directament a la memòria RAM. Implementa les mateixes interfícies que MongoDB per mantenir compatibilitat amb la resta del sistema, permetent operacions de cerca, filtrat i agregació sobre les dades emmagatzemades en memòria.

Justificació: Proporciona robustesa operacional assegurant que el sistema pugui funcionar en entorns de desenvolupament, proves o situacions d'emergència sense dependències externes. És especialment útil per a demos, testing automatitzat.

Problemes trobats:

1. Problema: Assegurar que les cerques i consultes sobre les dades emmagatzemades siguin ràpides i eficients.

- Solució: Creació d'índexs optimitzats per als patrons de consulta més comuns, ús del framework d'agregació de MongoDB per a consultes complexes, i implementació de tècniques de cache per a consultes repetitives.

2. Problema: Assegurar que les dades es mantinguin íntegres malgrat les fallades del sistema o operacions concurrents.

- Solució: Ús de transaccions de MongoDB quan és necessari, implementació de lògica de retry per a operacions fallides.També s'implementen validacions periòdiques per detectar possibles corrupcions de dades.

3. Problema: Definir un esquema de base de dades que sigui flexible i adequat per a les dades normalitzades i les relacions entre ells.

- Solució: Adopció d'un esquema de registre base que és compatible amb diferents tipus de logs mentre manté camps estandarditzats per a consultes eficients.

4.3 Desenvolupament i execució

4.3.1 Entorn de desenvolupament

Llenguatges de programació:

Python 3.8.10 o superior: utilitzat principalment per desenvolupar el sistema central SIEM, incloent l'API de recepció, normalització, anàlisi d'IA i correlació. S'ha escollit degut a les eines que ofereix i a la simplicitat del llenguatge.

Llibreries (requirements.txt):

pymongo>=3.12.0: Controlador oficial de MongoDB per a Python, utilitzat per a operacions síncrones amb la base de dades.
scikit-learn>=1.0.0: Llibreria fonamental per a l'algoritme Isolation Forest i el preprocessament de dades.
torch>=1.10.0: Framework de Deep Learning per implementar la xarxa neuronal.
numpy>=1.21.0: Biblioteca per a càlcul numèric i operacions amb arrays multidimensionals.
pandas>=1.3.0: Manipulació i anàlisi de dades estructurades dels logs.
scipy>=1.7.0: Funcions científiques i estadístiques avançades per a l'anàlisi.
regex>=2022.1.18: Processament avançat de patrons per a la normalització de logs.
python-dateutil>=2.8.2: Parser intel·ligent de dates i timestamps en diferents formats.
requests >=2.26.0: Client HTTP per a comunicacions amb l'API del SIEM. Utilitzat a collect_data.py.
colorama>=0.4.4: Millora de la visualització en terminal amb colors per al script de recollida.
pytest>=6.2.5: Per a proves unitàries i d'integració dels components.
black>=21.12b0: Eina de formatat automàtic de codi.
flake8>=4.0.1: Linter per verificar bones pràctiques de codi Python.
mypy>=0.910: Verificació de tipus estàtics.
joblib>=1.1.0: Serialització i paral·lelització per als models de Machine Learning.
motor>=3.1.2: Client MongoDB asíncron, útil si s'utilitza FastAPI amb operacions async.
fastapi>=0.68.0: Framework lleuger per crear APIs REST modernes.
uvicorn>=0.15.0: Servidor ASGI per executar aplicacions FastAPI.
pytest-cov>=2.12.1: Complement de pytest per mesurar la cobertura de codi durant les proves.

IDE utilitzats

Visual Studio Code / Cursor: Entorns principals utilitzats per a desenvolupar tot el projecte SIEM.

MongoDB Compass: Eina visual per a dissenyar, administrar i consultar la base de dades

Serveis i base de dades:

MongoDB 6.0 o superior: base de dades NoSQL principal per emmagatzemar logs processats i resultats d'anàlisi.

Servidor de desenvolupament: les instàncies de l'API FastAPI, i el nucli SIEM s'executen actualment en localhost per testejar el projecte.

4.3.2 Entorn d'execució

Requisits per posa en marxa el sistema:

1. Descarregar el projecte.
2. Instal·lar Python.
3. Instal·lació de llibreries dependent.
4. Instal·lació de la base de dades MongoDB
5. Configuracio de les variables d'entorn dins de " /.env "

Pas a pas per executar el sistema SIEM-AI:

1. Iniciar el servei API:

    Run: **" python api_server.py "**

    Paràmetres principals:

        --host: Adreça IP del servidor (per defecte: 127.0.0.1)

        --port: Port del servidor (per defecte: 8000)

        --reload: Recarrega automàtica de les dades del SIEM

        --config: Arxiu de configuració personalitzat

2. Executar el col·lector de dades

    Run: **" python collect_data.py --directory name_test_folder --reset "**

    Paràmetres principals:

        --directory: Important especificar la carpeta dins de test_logs/ la qual conte els test de logs a processar.

        --batch-size: Mida dels lots d'enviament (per defecte 100)

        --api-url: URL de l'API (per defecte http://127.0.0.1)

        --reset: Reinicia les dades abans d'enviar nous logs per no identificar anàlisis antics.

        --concurrent: Nombre de fils concurrents (per defecte 3)

5 Proves

5.1 Definició

Objectiu del Pla de Proves: El pla de proves té com a objectiu verificar la correcta ingesta i processament de logs de diferents fonts i escenaris d'atac per part del sistema SIEM, així com la seva capacitat per detectar i correlacionar amenaces mitjançant intel·ligència artificial. Les proves es basaran en l'ús del script collect_data.py, que permet enviar els fitxers de log presents a la carpeta test_logs a l'API del SIEM.

Metodologia de Proves: La metodologia emprada per a la realització del pla de proves es basa en l'execució d'un casos de prova dissenyats per validar els requisits tècnics del sistema. Aquest cas de prova consisteix en un atac orquestrat a una empresa real, on es realitzarà un conjunt de atacs complexos que el sistema ha de poder tractar correctament.

Condicions Inicials per a Cada Prova:

1. **Reset del sistema:** Abans de cada prova, s'executarà --reset per garantir un estat net
2. **Base de dades:** MongoDB ha d'estar operativa i accessible
3. **API activa:** El servidor API del SIEM ha d'estar executant-se.
4. **Models d'IA:** Els models Isolation Forest i Neural Network han d'estar carregats

5.1.1 Proves funcionals

Prova 1: Recopilació de Registres en Temps Real

Objectiu: Verificar que el sistema és capaç de recopilar registres de seguretat en temps real provinents de diferents fonts.

Requisits avaluats: RF1, RF3

Procediment de prova:

1. Verificar la connectivitat del sistema amb diferents fonts de logs.
2. Iniciar la ingesta de logs en temps real.
3. Monitoritzar la recepció correcta de registres de múltiples fonts.

Resultats esperats: El sistema ha de recopilar correctament logs de diferents fonts sense pèrdua de dades, mantenint la sincronització temporal i processant-los en temps real.

Prova 2: Detecció d'Incidents Automatitzada

Objectiu: Verificar que el sistema detecta i alerta sobre incidents de seguretat de forma automatitzada.

Requisits avaluats: RF2, RF4

Procediment de prova:

1. Processar logs que continguin indicadors d'incidents de seguretat.
2. Verificar la detecció automàtica d'amenaces.
3. Comprovar la generació automàtica d'alertes.

Resultats esperats: El sistema ha de detectar automàticament incidents de seguretat i generar alertes corresponents sense intervenció manual.

Prova 3: Anàlisi Automatitzat amb Intel·ligència Artificial

Objectiu: Verificar que el sistema analitza els registres mitjançant tècniques d'IA per identificar comportaments anòmals.

Requisits avaluats: RF3

Procediment de prova:

1. Verificar l'estat dels models d'intel·ligència artificial.
2. Processar logs amb patrons normals i anòmals.
3. Analitzar les puntuacions d'anomalia generades pels models.
4. Verificar la identificació de comportaments anòmals.

Resultats esperats: Els models d'IA han d'identificar correctament comportaments anòmals amb puntuacions de confiança adequades i classificar correctament amenaces potencials.

Prova 4: Sistema d'Alertes en Temps Real

Objectiu: Verificar que el sistema genera alertes en temps real al responsable de seguretat.

Requisits avaluats: RF4, RF2

Procediment de prova:

1. Configurar els paràmetres del sistema d'alertes.
2. Simular incidents que requereixin notificació immediata.
3. Mesurar el temps de generació d'alertes.
4. Verificar la recepció correcta de notificacions.

Resultats esperats: El sistema ha de generar alertes en temps real amb temps de resposta inferiors a 30 segons i entregar les notificacions correctament

Prova 5: Correlació d'Esdeveniments Multi-Font

Objectiu: Verificar que el sistema correlaciona diversos esdeveniments de fonts diferents per identificar atacs complexos.

Requisits avaluats: RF5, RF3

Procediment de prova:

1. Processar logs simultanis de diferents fonts.
2. Analitzar la identificació de relacions entre esdeveniments.
3. Verificar la construcció de cadenes d'atac.
4. Comprovar la generació de correlacions temporals.

Resultats esperats: El sistema ha de correlacionar correctament esdeveniments de diferents fonts, identificant patrons d'atac complexos i relacions temporals entre incidents.

5.1.2 Proves no funcionals

Prova 6: Rendiment de Processament

Objectiu: Verificar que el sisema pot processar i analitzar una gran quantitat de logs eficientment.

Requisits avaluats: RNF1

Procediment de prova:

1. Executar proves de càrrega amb volums alts de logs.
2. Mesurar la velocitat de processament.
3. Monitoritzar l'ús de recursos del sistema.
4. Verificar el manteniment del rendiment sota càrrega.

Resultats esperats: El sistema ha de processar mes de 2 logs per segon mantenint l'eficiència i sense degradació del rendiment.

Prova 7: Disponibilitat i Persistència de Dades

Objectiu: Verificar que el sistema assegura la persistència de dades i alertes obtingudes.

Requisits avaluats: RNF2

Procediment de prova:

1. Processar logs i generar alertes.
2. Verificar l'emmagatzematge persistent a la base de dades.
3. Comprovar la integritat de les dades emmagatzemades.
4. Verificar la recuperació de dades històriques.

Resultats esperats: El sistema ha de emmagatzemar el 100% de logs i alertes, mantenint disponibilitat superior al 99% i integritat completa de dades.

Prova 8: Eficiència en l'Ús de Recursos

Objectiu: Verificar que els processos optimitzen l'ús dels recursos per garantir un funcionament fluid.

Requisits avaluats: ROP1, ROP2

Procediment de prova:

1. Executar operacions estàndard del sistema.
2. Monitoritzar l'ús de CPU, memòria i disc.
3. Analitzar l'eficiència dels processos.
4. Verificar l'optimització de recursos.

Resultats esperats: El sistema ha de mantenir un ús de recursos raonable per no saturar els recursos i assegurar un correcte funcionament.

5.1.3 Resultats esperats de l'atac proposat

-Informe d'Atac Empresarial:

Atacant Extern:

IP: 203.0.113.45
Descripció: Actor maliciós extern que inicia el vector d'atac.

Servidor Maliciós:

IP: 198.51.100.88
Descripció: Servidor extern controlat pels atacants on envien dades exfiltrades.

-Fases de l'Atac:

1. **Reconeixement Extern**
   - **IP:** 203.0.113.45
   - **Acció:** Escaneig de ports i serveis de la xarxa externa.
   - **Objectiu:** Identificar vulnerabilitats per accedir a la xarxa interna.
2. **Compromís del Pacient Zero**
   - **IP:** 192.168.10.15
   - **Acció:** Infecció inicial mitjançant phishing o exploit remot.
   - **Resultat:** Entrada a la xarxa; inici del control des del servidor maliciós.
3. **Comunicació amb Servidor Maliciós**
   - **IP:** 198.51.100.88
   - **Acció:** El pacient zero estableix connexió amb el servidor d'atacants.
   - **Funció:** Rebre comandes i enviar dades robades.
4. **Moviment Lateral**
   - **IPs:** 192.168.10.29, 192.168.10.48, 192.168.10.93
   - **Acció:** Propagació del malware a altres equips d'usuari.
   - **Objectiu:** Ampliar el control dins la xarxa.
5. **Escaneig Intern i Reconneixement**
   - **IPs Escanejades:** 192.168.10.100 – 119
   - **Acció:** Escaneig de ports i serveis interns múltiples.
   - **Objectiu:** Identificar sistemes crítics i vulnerables.
6. **Compromís de Sistemes Crítics**
   - **IPs Compromeses:** 192.168.10.101, .110 - .118
   - **Acció:** Accés i control dels sistemes clau.
   - **Resultat:** Accés a dades sensibles i estructures internes.
7. **Exfiltració de Dades**
   - **IPs Origen:** 192.168.10.103 - .119 (Servidors de dades)
   - **IP Destinació:** 198.51.100.88
   - **Acció:** Transferència massiva de documents i fitxers confidencials.
8. **Fallida en Compromís d'Alguns Hosts**
   - **IPs** Resistents: 192.168.10.100, 192.168.10.102
   - **Resultat:** Sistemes escanejats però no compromesos.

Aquest atac ha estat dissenyat per tindre complexitat i poder identificar molts tipus d'esdeveniments diferents. El total de dades recopilades serà un total de 2400 logs on provindran de un Firewall i un IDS.

5.2 Avaluació de requisits

Prova 1: Recopilació de Registres en Temps Real

En aquest escenari específic, el sistema processa dos arxius crítics (pfsense_enterprise_traffic.log i suricata_enterprise_detection.json) amb 16 lots cadascun, demostrant la capacitat de gestionar 2.400 logs totals de manera eficient i controlada.

Podem observar com el sistema implementa una estratègia d'optimització mitjançant el processament per lots (batch), on els logs s'agrupen en blocs de 75 registres per millorar l'eficiència del sistema. Cada log es troba en un format original, on s'ha d'extreure la informació important (verificar **Annex 1** per veure alguns logs que el sistema ingereix).

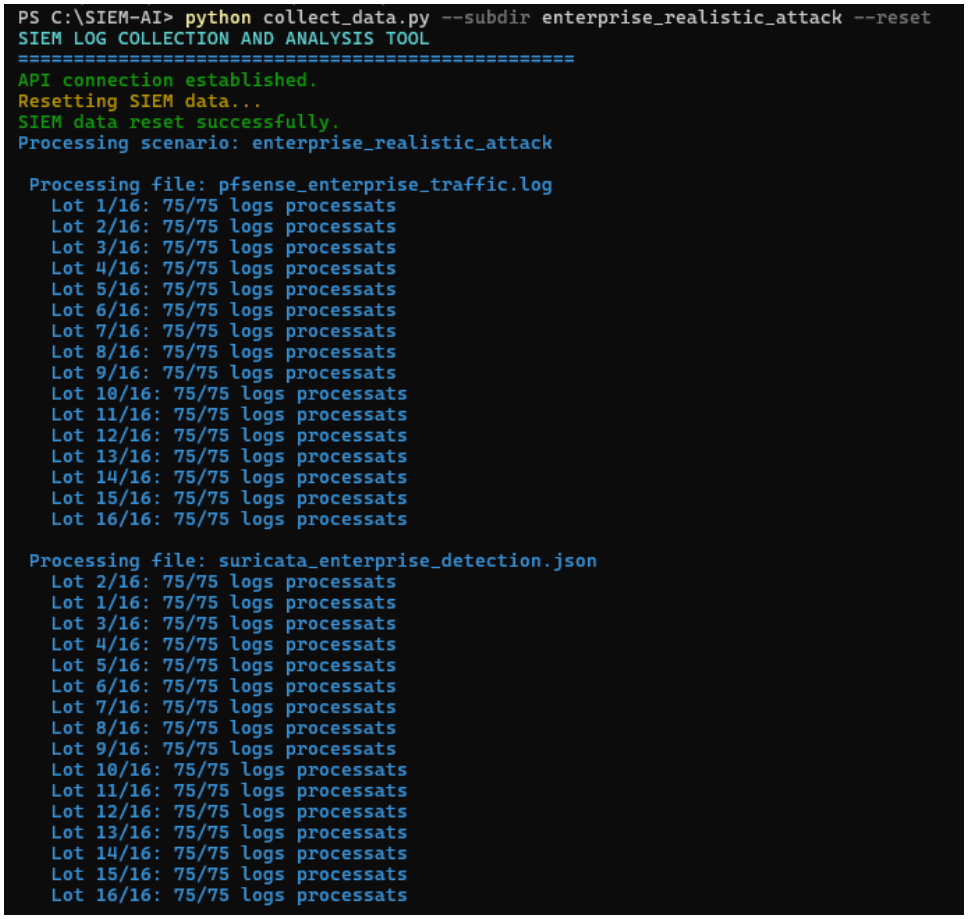


Figura 7. Output de l'enviamet dels logs al sistema

En cas d'error d'enviament, el sistema contempla reintents per enviar un altre cop els logs que no s'han proccesat:

```
Processing scenario: enterprise_realistic_attack

 Processing file: pfsense_enterprise_traffic.log
   Lot 1/16: 75/75 logs processats
   Lot 2/16: 75/75 logs processats
   Lot 3/16: 75/75 logs processats
   Lot 4/16: 75/75 logs processats
   Lot 5/16: 75/75 logs processats
   Lot 6/16: 75/75 logs processats
   Lot 7/16: 75/75 logs processats
   Lot 8/16: 75/75 logs processats
   Lot 9/16: 75/75 logs processats
   Lot 10/16: 75/75 logs processats
   Lot 11/16: 75/75 logs processats
   Lot 12/16: 75/75 logs processats
   Lot 13/16: 75/75 logs processats
   Lot 14/16: Error HTTPConnectionPool(host='localhost', port=8000): Read timed out. (read timeout=67.5)
   Reintentant lot 14 (intent 2/2)...
   Lot 15/16: Error HTTPConnectionPool(host='localhost', port=8000): Read timed out. (read timeout=67.5)
   Reintentant lot 15 (intent 2/2)...
   Lot 14/16: 75/75 logs processats
   Lot 15/16: 75/75 logs processats
   Lot 16/16: 75/75 logs processats
```

Figura 8. Output d'error al enviament dels logs

Resultats: El sistema ha processat correctament 2.400 logs provinents de diferents fonts amb diferents tipus de dades. Els resultats de la prova, demostren que els logs de diferents fonts, son recopilats i enviats al sistema SIEM correctament. Les dades son normalitzades en el format unuificat del sistema (veure **Annex 2** per visualitzar els formats unificats)**.**

Prova 2: Detecció d'Incidents Automatitzat

Un cop les dades s'han rebut correctament al sistema SIEM, el sistema analitza les dades rebudes i genera un resum amb la informació que ha obtingut. El sistema ha processat de manera autònoma els 2.400 logs de l'escenari empresarial realista, on moltes de les dades son tràfic normal.

```
==================================================
THREAT DETECTION SUMMARY
==================================================

APT CAMPAIGN:
  Severity: MEDIUM
  Detections: 30
  Source IPs: 192.168.10.15, 192.168.10.21, 192.168.10.29, 192.168.10.48, 192.168.10.93
  Target IPs: 192.168.10.101, 192.168.10.103, 192.168.10.105, 192.168.10.108, 192.168.10.110, 192.168.10.112, 192.168.10.113, 192.168.10.114, 192.168.10.118, 198.51.100.88
  Time range: 08:57:33 26/05/2025
              to 08:59:57 26/05/2025
  AI Confidence: 95.0%
  Anomaly Score: 0.980

COMMAND CONTROL COMMUNICATION:
  Severity: MEDIUM
  Detections: 24
  Source IPs: 192.168.10.103, 192.168.10.104, 192.168.10.105, 192.168.10.106, 192.168.10.108, 192.168.10.109, 192.168.10.119, 192.168.10.15, 192.168.10.21, 192.168.10.29, 192.168.10.48, 19
2.168.10.93
  Target IPs: 198.51.100.88
  Time range: 08:57:33 26/05/2025
              to 08:59:54 26/05/2025
  AI Confidence: 91.2%
  Anomaly Score: 0.950

DATA EXFILTRATION:
  Severity: MEDIUM
  Detections: 9
  Source IPs: 192.168.10.103, 192.168.10.104, 192.168.10.105, 192.168.10.106, 192.168.10.108, 192.168.10.109, 192.168.10.119
  Target IPs: 198.51.100.88
  Time range: 08:57:54 26/05/2025
              to 08:59:36 26/05/2025
  AI Confidence: 85.0%
  Anomaly Score: 0.900

DATA_EXFILTRATION:
  Severity: LOW
  Detections: 4
  Source IPs: 192.168.10.106, 192.168.10.108, 192.168.10.109, 192.168.10.119
  Target IPs: 198.51.100.88
  Time range: 08:59:03 26/05/2025
              to 08:59:36 26/05/2025
```

Figura 9. Output dels resultats d'alertes 1/3

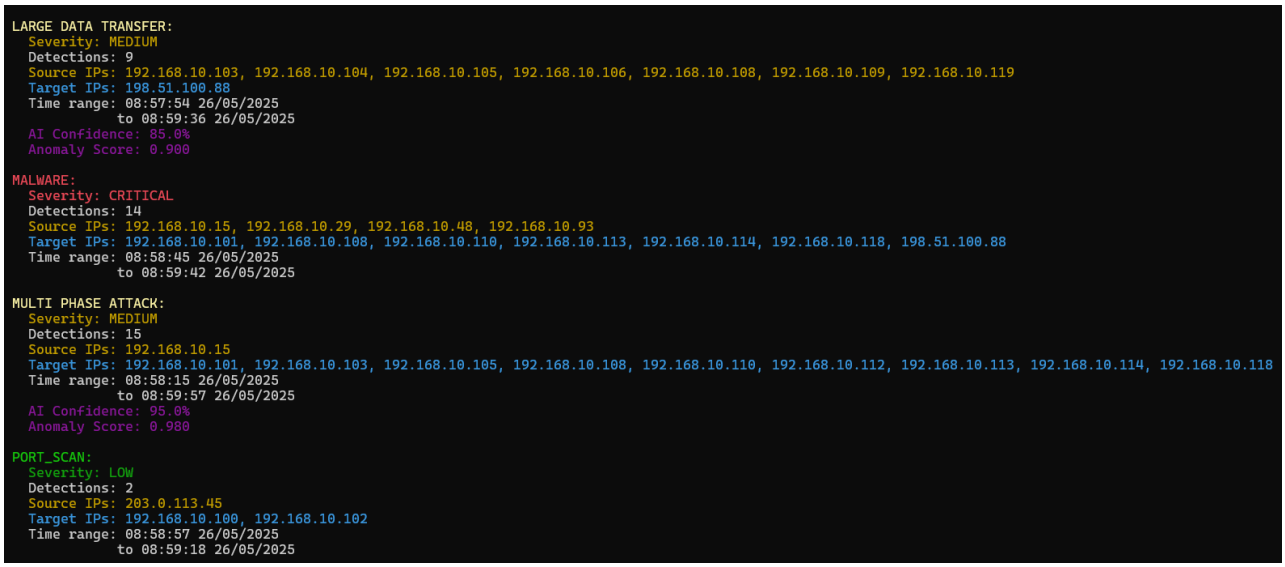


Figura 10. Output dels resultats d'alertes 2/3

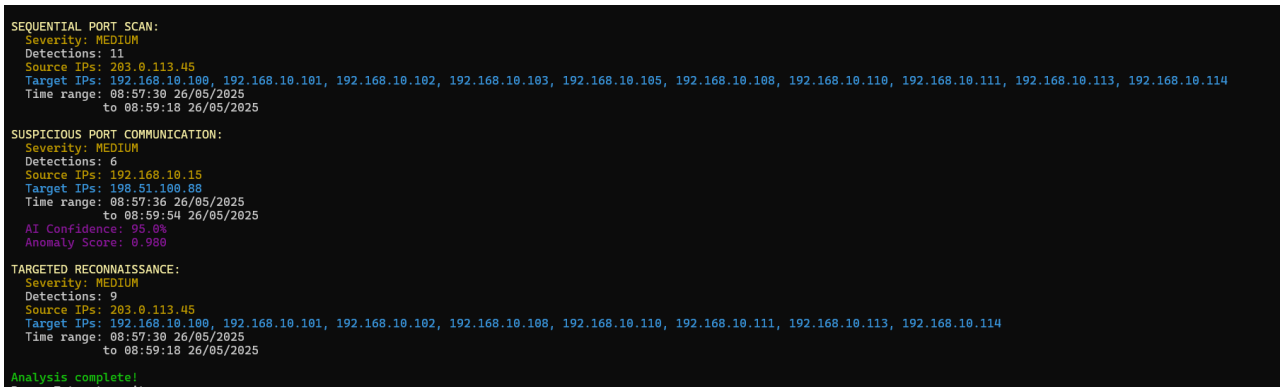


Figura 11. Output dels resultats d'alertes 3/3

Resultats: El sistema ha identificat amb precisió els diferents tipus d'atacs complexos, demostrant una capacitat excepcional per diferenciar tràfic normal i tràfic maliciós dins de l'entorn empresarial analitzat:

S'ha detectat les IPs compromeses internes (192.168.10.x) i els servidors de comandament i control externs (198.51.100.88, 203.0.113.45), establint mapes de relacions entre atacants i objectius. Per cada detecció, el sistema proporciona context específic incloent les IPs involucrades, el tipus d'atac, el nivell de severitat, i la finestra temporal de l'activitat, permetent als analistes comprendre completament l'abast i impacte de l'incident.

Prova 3: Anàlisi Automatitzat amb Intel·ligència Artificial

Cada resultat detectat per els algoritmes d'Intel·ligència Artificial, proporciona les dades obtingudes per poder revisar quin grau de confiança tenen els valors obtinguts.

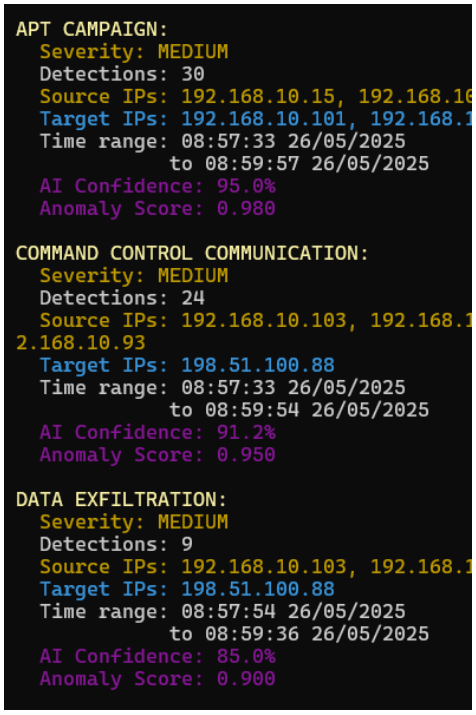


Figura 12. Output dels valors obtinguts per la AI

Resultats: Els algoritmes d'intel·ligència artificial han demostrat una capacitat excepcional per identificar amenaces sofisticades que romanen invisibles als sistemes de detecció tradicionals. Els models d'aprenentatge automàtic implementats han evidenciat una robustesa notable en la identificació d'atacs complexos, superant significativament les limitacions dels enfocaments basats únicament en regles pre-definides.

Aquesta demostració valida el potencial transformador de la intel·ligència artificial en ciberseguretat, confirmant que l'automatització intel·ligent pot elevar significativament les capacitats defensives organitzacionals més enllà dels límits dels sistemes tradicionals.

Prova 4: Sistema d'Alertes en Temps Real

Un cop el sistema ha detectat algun tipus d'alerta, els resultats son enviades a la base de dades MongoDB on es recopilen per la seva persistència i anàlisis a futur.

alerts_summary

	count Int32	date String	severity String	source_type String	status String
1	44	"2025-05-26"	"medium"	"suricata"	"new"
2	14	"2025-05-26"	"medium"	"pfsense"	"new"
3	12	"2025-05-26"	"low"	"pfsense"	"new"
4	76	"2025-05-26"	"critical"	"suricata"	"new"

Figura 13. Resum de les alertes detectades a la Base de Dades

Figura 14. Exemple de alerta dins de la Base de Dades

Resultats: Podem veure que s'han detectat diferents tipus d'alertes segons la seva severitat , on les alertes son classificades amb tota la informació que el sistema genera de manera detallada (veure en **Annex 3** tota la informació que recopila cada alerta).

Prova 5: Correlació d'Esdeveniments Multi-Font

El sistema analitza la capacitat de correlacionar esdeveniments aparentment independents per identificar atacs coordinats i campanyes sofisticades que es desenvolupen a través de múltiples fases temporals i sistemes diferents.

```
CORRELATION ANALYSIS:
Found 3 correlations between events

  Correlation 1:
  Alert: Escenario de malware detectado (SCN-malware)
  Source IP: 192.168.10.29, 192.168.10.93, 192.168.10.15, 192.168.10.48
  Target IP: 192.168.10.101, 192.168.10.113, 192.168.10.118, 192.168.10.108, 192.168.10.110, 198.51.100.88, 192.168.10.114
  Risk score: 0.70
  Related events: 14
    Pattern 1: malware
      Description: Escenario de malware detectado
      Confidence: 0.70
      Severity: critical

  Correlation 2:
  Alert: Escenario de data_exfiltration detectado (SCN-data_exfiltration)
  Source IP: 192.168.10.108, 192.168.10.119, 192.168.10.109, 192.168.10.106
  Target IP: 198.51.100.88
  Risk score: 0.70
  Related events: 4
    Pattern 1: data_exfiltration
      Description: Escenario de data_exfiltration detectado
      Confidence: 0.70
      Severity: medium

  Correlation 3:
  Alert: Escenario de port_scan detectado (SCN-port_scan)
  Source IP: 203.0.113.45
  Target IP: 192.168.10.100, 192.168.10.102
  Risk score: 0.70
  Related events: 2
    Pattern 1: port_scan
      Description: Escenario de port_scan detectado
      Confidence: 0.70
      Severity: medium
```

Figura 15. Output de les correlacions obtingudes

Resultats: El motor de correlació ha identificat amb èxit tres escenaris d'atac interconnectats, demostrant la capacitat del sistema per reconstruir narratives complexes d'amenaces:

- Correlació d'infecció per malware: El sistema ha correlacionat múltiples esdeveniments de malware distribuït, identificant un patró d'infecció coordinada que afecta diversos hosts de manera simultània, indicant una campanya de propagació automatitzada.
- Correlació d'exfiltració de dades: Ha detectat i correlacionat activitats de transferència de dades sospitoses, reconstruint un patró d'exfiltració sistemàtica dirigida cap a infraestructura externa controlada pels atacants.
- Correlació de reconeixement de xarxa: El motor ha identificat activitats de reconeixement i escaneig sistemàtic, correlacionant esdeveniments que revelen una fase preparatòria d'atac dirigit contra objectius específics.

Prova 6: Rendiment de Processament

Per poder verificar el rendiment del sistema, s'ha calculat el tems de enviament i processament de cada log.

```
PROCESSING SUMMARY
==================================================
  Files processed: 2/2
  Total logs sent: 2400
  Total time: 361.5 seconds
  Average speed: 6.6 logs/second
  Success rate: 100.0%
```

Figura 16. Output del rendiment obtingut

Resultat: El temps de processament ha sigut de 361.5 segons, es a dir 6.6 logs per segon, per a 2.400 logs reflecteix l'eficiència de l'arquitectura implementada, incloent l'optimització del processament per lots, la comunicació HTTP eficient entre components i la capacitat de l'API Server per gestionar múltiples peticions de manera asíncrona.

Prova 7: Disponibilitat i Persistència de Dades

Aquesta prova verifica la capacitat del sistema per emmagatzemar, mantenir i recuperar tota la informació crítica processada, garantint la disponibilitat constant de dades tant per a consultes en temps real com per a anàlisi forense posterior.

logs

	timestamp Date	source_type String	raw_log String	event_category String
26	2025-05-26T08:00:50.000+0…	"pfsense"	"May 26 08:00:50 pfsense …	"network"
27	2025-05-26T08:00:52.000+0…	"pfsense"	"May 26 08:00:52 pfsense …	"network"
28	2025-05-26T08:00:54.000+0…	"pfsense"	"May 26 08:00:54 pfsense …	"network"
29	2025-05-26T08:00:56.000+0…	"pfsense"	"May 26 08:00:56 pfsense …	"network"
30	2025-05-26T08:00:58.000+0…	"pfsense"	"May 26 08:00:58 pfsense …	"network"
31	2025-05-26T08:01:00.000+0…	"pfsense"	"May 26 08:01:00 pfsense …	"network"
32	2025-05-26T08:01:02.000+0…	"pfsense"	"May 26 08:01:02 pfsense …	"network"
33	2025-05-26T08:01:04.000+0…	"pfsense"	"May 26 08:01:04 pfsense …	"network"
34	2025-05-26T08:01:06.000+0…	"pfsense"	"May 26 08:01:06 pfsense …	"network"
35	2025-05-26T08:01:08.000+0…	"pfsense"	"May 26 08:01:08 pfsense …	"network"
36	2025-05-26T08:01:10.000+0…	"pfsense"	"May 26 08:01:10 pfsense …	"network"
37	2025-05-26T08:01:12.000+0…	"pfsense"	"May 26 08:01:12 pfsense …	"network"
38	2025-05-26T08:01:14.000+0…	"pfsense"	"May 26 08:01:14 pfsense …	"network"
39	2025-05-26T08:01:16.000+0…	"pfsense"	"May 26 08:01:16 pfsense …	"network"
40	2025-05-26T08:01:18.000+0…	"pfsense"	"May 26 08:01:18 pfsense …	"network"
41	2025-05-26T08:01:20.000+0…	"pfsense"	"May 26 08:01:20 pfsense …	"network"

Figura 17. Taula resum dels logs normalitzats a la Base de Dades


```
timestamp : 2025-05-26T08:02:32.000+00:00
source_type : "pfsense"
raw_log : "May 26 08:02:32 pfsense filterlog: 5,,,1000000176,igb0,match,pass,out,…"
event_category : "network"
dns_answer : null
event_description : "PFSense pass tcp connection from 192.168.10.14:59046 to 1.1.1.1:80"
firewall_action : "pass"
process_id : null
event_type : "connection_allowed"
dst_ip : "1.1.1.1"
▸ additional_fields : Object
src_port : "59046"
dst_port : "80"
_id: ObjectId('684607b3c341678e303950b7')
event_severity : "info"
user_id : null
ids_signature : null
event_id : "85e7450a-8ee6-4c6c-996b-b697c454d0ee"
http_status : null
protocol : "tcp"
source_name : "pfsense"
src_ip : "192.168.10.14"
src_mac : null
process_name : null
dns_query : null
▾ Show 5 more fields
```


Figura 18. Exemple de Log Normalitzat

alerts

	_id ObjectId	id String	timestamp Date	source_type String	event_type String
101	ObjectId('68460c1f947138…	"ALT-1749421086-6945"	2025-05-26T08:57:42.000+…	"suricata"	"alert"
102	ObjectId('68460c1f947138…	"ALT-1749421086-8832"	2025-05-26T08:57:45.000+…	"suricata"	"alert"
103	ObjectId('68460c1f947138…	"ALT-1749421086-1911"	2025-05-26T08:57:48.000+…	"suricata"	"alert"
104	ObjectId('68460c1f947138…	"ALT-1749421086-8574"	2025-05-26T08:57:51.000+…	"suricata"	"alert"
105	ObjectId('68460c1f947138…	"ALT-1749421086-5733"	2025-05-26T08:57:54.000+…	"suricata"	"alert"
106	ObjectId('68460c1f947138…	"ALT-1749421087-9987"	2025-05-26T08:57:57.000+…	"suricata"	"alert"
107	ObjectId('68460c1f947138…	"ALT-1749421087-4725"	2025-05-26T08:58:00.000+…	"suricata"	"alert"
108	ObjectId('68460c1f947138…	"ALT-1749421087-8478"	2025-05-26T08:58:03.000+…	"suricata"	"alert"
109	ObjectId('68460c1f947138…	"ALT-1749421087-7602"	2025-05-26T08:58:06.000+…	"suricata"	"alert"
110	ObjectId('68460c24947138…	"ALT-1749421087-2530"	2025-05-26T08:58:09.000+…	"suricata"	"alert"
111	ObjectId('68460c24947138…	"ALT-1749421087-1532"	2025-05-26T08:58:12.000+…	"suricata"	"alert"
112	ObjectId('68460c24947138…	"ALT-1749421088-8756"	2025-05-26T08:58:15.000+…	"suricata"	"alert"
113	ObjectId('68460c24947138…	"ALT-1749421088-4079"	2025-05-26T08:58:18.000+…	"suricata"	"alert"
114	ObjectId('68460c24947138…	"ALT-1749421088-5130"	2025-05-26T08:58:21.000+…	"suricata"	"alert"

Figura 19. Taula resum del les alertes generades a la Base de Dades

Resultats: El sistema ha demostrat una capacitat integral d'emmagatzematge i recuperació de dades crítiques:

- **Persistència completa de logs originals:** El sistema manté un registre detallat de tots els logs d'entrada assegurant la traçabilitat completa de la informació original per a auditories i anàlisi forense.
- **Emmagatzematge d'alertes:** Les deteccions generades per la intel·ligència artificial s'emmagatzemen amb metadades extenses incloent totes les dades que el sistema ha identificat.

Prova 8: Eficiència en l'Ús de Recursos

Els resultats són analitzats mitjançant una eina especialitzada de monitorització que supervisa els recursos de la base de dades en temps real durant l'execució del sistema SIEM.

Figura 20. Monitorització dels recursos durant l'execució del atac

Resultats: L'anàlisi de rendiment demostra una eficiència excepcional en l'ús de recursos del sistema durant el processament dels 2.400 logs de l'escenari empresarial. En cap moment es sobrecarregen els recursos i les connexions, on l'ús de memòria resident de 270MB representa un consum moderat i controlat.

5.3 Avaluació d'objectius

Validació dels Objectius Tècnics Mínims:

-Objectiu 1: Recopilació i tractament correcte de dades externes amb classificació adequada

Els resultats de la Prova 1 demostren que el sistema ha processat amb èxit 2.400 logs provinents de múltiples fonts (PFsense i Suricata) mitjançant una estratègia optimitzada de processament per lots. El sistema ha implementat parsers especialitzats que normalitzen efectivament diferents formats de dades (logs de firewall i deteccions d'IDS) a un esquema unificat, mantenint la integritat de la informació original mentre facilita el processament posterior. La classificació automàtica per tipus de font i la validació de format confirmen que el sistema tracta correctament la diversitat de dades externes.

Objectiu 2: Detecció de tipus comuns d'atacs en entorns reals

La Prova 2 evidencia que el sistema ha identificat amb precisió múltiples vectors d'atac dins de l'escenari empresarial realista, incloent reconeixement extern, compromís de sistemes, moviment lateral, escaneig intern i exfiltració de dades. El sistema ha detectat automàticament IPs compromeses internes i servidors de comandament extern, demostrant capacitat per identificar tant atacs simples com campanyes sofisticades. La diversitat d'amenaces detectades (malware, exfiltració, reconeixement) confirma la robustesa del sistema davant tipus comuns d'atacs cibernètics.

Objectiu 3: Millora de deteccions mitjançant algoritmes intel·ligents

Els resultats de la Prova 3 confirmen que els algoritmes d'intel·ligència artificial (Isolation Forest i xarxes neuronals) han superat significativament les limitacions dels sistemes basats únicament en regles predefinides. Els models han demostrat capacitat per identificar amenaces sofisticades que romanen invisibles als enfocaments tradicionals, proporcionant puntuacions de confiança quantificables i identificant patrons complexos d'atac. L'anàlisi automatitzada ha generat alertes amb alt grau de precisió, validant l'efectivitat dels algoritmes intel·ligents implementats.

Objectiu 4: Revisió de dades i resultats al llarg del temps

La Prova 7 demostra que el sistema garanteix la persistència completa de logs originals, resultats d'anàlisi i alertes generades mitjançant emmagatzematge estructurat a MongoDB. Totes les dades processades romanen disponibles per a consultes històriques, anàlisi forense i auditories, amb metadades extenses que faciliten la traçabilitat temporal. La capacitat de recuperació de dades històriques i la integritat mantinguda al llarg del temps confirmen que els analistes poden revisar evolucions d'amenaces i tendències de seguretat.

Objectiu 5: Correlació d'esdeveniments per identificar atacs complexos

Els resultats de la Prova 5 evidencien que el motor de correlació ha reconstruït amb èxit narratives complexes d'amenaces, identificant tres escenaris d'atac interconnectats: infecció per malware coordinada, exfiltració sistemàtica de dades i reconeixement de xarxa preparatori. El sistema ha demostrat capacitat per correlacionar esdeveniments aparentment independents de múltiples fonts temporals, generant una comprensió holística de campanyes sofisticades que es desenvolupen a través de diferents fases i sistemes.

6 Distribució, manteniment i evolució

6.1 Distribució

La solució es distribueix com una aplicació Python, amb els seus components principals i dependències gestionades a través d'un fitxer requirements.txt. Es distribueix com a codi font que s'executa directament sobre un intèrpret de Python.

La distribució del sistema SIEM-AI s'ha dissenyat seguint un enfocament modular i escalable que permet adaptabilitat a diferents entorns empresarials, des de petites organitzacions fins a infraestructures corporatives complexes.

Components del sistema:

-Core SIEM:
API Server
Motors d'anàlisi IA
Sistema de correlació
Base de dades MongoDB

-Agents de recollida:
Col·lectors de logs
Parsers específics

-Configuració:
Fitxers de configuració
Scripts d'inicialització

Procés d'instal·lació i execució:

Prerequisits:

- Python 3.8+
- MongoDB instal·lat i actiu
- Descarregar dependències

Instal·lació:

- Clonar el repositori o descarregar l'arxiu zip
- Instal·lar dependències
- Configurar el sistema segons l'entorn

Actualització:

Les actualitzacions en un futur seran gestionades mitjançant un sistema de control de versions. Quan les actualitzacions impliquin canvis en l'estructura de la base de dades o els models d'intel·ligència artificial, s'inclouran scripts de migració automatitzats que assegurin la compatibilitat amb les dades existents. A més, es publicarà una guia d'actualització amb cada versió per facilitar el procés a administradors i usuaris.

6.2 Manteniment

La solució requereix un manteniment actiu per garantir el seu funcionament òptim i la seva eficàcia en la detecció d'amenaces.

Perfils Requerits:

- **Administrador de Sistemes:** Aquest perfil seria responsable del desplegament, el monitoratge de la infraestructura (servidor, base de dades), la gestió de còpies de seguretat i l'aplicació d'actualitzacions del programari.
- **Analista de Seguretat:** L'analista de seguretat és l'usuari principal del sistema. S'encarregaria d'analitzar les alertes generades, investigar incidents i ajustar o crear noves regles de correlació per millorar la detecció.

Tasques de Manteniment:

Monitoratge:

- Vigilar els logs de l'aplicació per a detectar errors.
- Monitorar el rendiment del servidor i sistemes.
- Utilitzar l'endpoint /health de l'API per a verificacions d'estat automatitzades.

Gestió de Dades:

- Si s'utilitza una base de dades persistent, cal realitzar còpies de seguretat periòdiques.
- Gestionar el creixement de l'emmagatzematge de logs i alertes.

Gestió de Models d'IA:

- Els models emmagatzemats al directori saved_models/ han de ser avaluats i, si és necessari, re-entrenats periòdicament amb dades recents per evitar la degradació del seu rendiment.
- Actualització de Regles i Lògica de Detecció: Donat que el panorama d'amenaces canvia constantment, la lògica de correlació i les regles de detecció han de ser revisades i actualitzades regularment per reflectir noves tàctiques i tècniques d'atac.

6.3 Evolució

Versió 1.1 (Curt Termini - pròxims 3 mesos)

Millora de Parsers de Logs: Afegir suport natiu per a més fonts de dades rellevants per poder tindre major flexibilitat amb altres sistemes.

Refinament de Regles de Correlació: Ampliar el motor de correlació per a detectar escenaris d'atac més complexos i de múltiples etapes.

UI Web Bàsica: Desenvolupar un quadre de comandament senzill per a visualitzar estadístiques i realitzar les tasques principals, facilitant la feina a l'analista.

Telegram: Interconnectar alertes critiques a un compte de telegram per rebre alertes instantàniament sense entrar al la UI del sistema.

Certificats: Afegir seguretat a les comunicacions del sistema (HTTPS/SSL).

Versió 1.2 (Mig Termini - pròxims 6-9 mesos)

Nous algoritmes AI: Implementar algoritmes mes complexos per millorar la detecció d'atacs.

Tipus d'atacs: Afegir i millorar els tipus d'atacs que el sistema identifica i alerta.

Interfície d'Usuari Avançada: Expandir la UI web per a convertir-la en una eina completa d'investigació, amb cerques complexes de logs, visualitzacions de línies de temps d'atacs i generació d'informes.

Inici de sessió: Implementar un inici de sessió per a que nomes els analistes puguin

Versió 2.0 (Llarg Termini - 12+ mesos)

Arquitectura Cloud-Native: Refactoritzar la solució per a un desplegament en contenidors Docker i orquestració amb Kubernetes, evolucionant cap a una arquitectura de microserveis.

Entorn real: El sistema ha d'acabar de preparar-se per un entorn de producció real per poder automatitzar-lo correctament.

7 Conclusions

Throughout the development of the project, significant progress has been made both technically and personally. From the outset, the growing need for proactive threat detection systems in cybersecurity was identified systems capable of adapting to real-world contexts and responding effectively to complex threats. The system had to be able to collect, process, and analyze large volumes of logs using artificial intelligence. Overall, this objective has been successfully achieved, even surpassing some of the initial expectations.

On one hand, the structured planning based on the waterfall methodology made it possible to complete each task. However, I would like to highlight that many phases were delayed due to errors, technical issues, and, in some cases, a lack of initial knowledge in the field. These challenges, however, became opportunities to deepen my understanding and make well-informed decisions.

On the other hand, the modular architecture based on solid design patterns proved to be effective, facilitating the scalability and maintainability of the system. Furthermore, the combination of intelligent algorithms, both supervised and unsupervised, enabled a high level of accuracy in threat detection, meeting both the functional and non-functional requirements defined at the start.

The implementation posed key challenges, such as the difficulty of generating a realistic testing environment and the need to interpret and transform logs from various sources and formats. Faced with the impossibility of fully replicating a real business environment, manually generating representative logs was chosen, allowing the system to be validated in a controlled manner. Another important challenge was tuning and training the AI models, especially to reduce false positives; this was addressed by fine-tuning the algorithm parameters and iteratively testing with different datasets.

The results of the testing phase demonstrated that both the technical requirements and objectives were met, validating the correct operation of the system under various attack scenarios. Ultimately, this project has not only been a demanding technical challenge but also a personal growth experience that reinforces my commitment to continuous learning in the cybersecurity field.

On a personal level, the project has provided deep learning about machine learning techniques applied to cybersecurity, data management, and the design of distributed systems. The experience has also strengthened key skills such as problem-solving, organization, and decision-making.

In conclusion, the development of this project has not only met the defined technical goals but has also exceeded personal growth expectations in the cybersecurity domain, demonstrating the viability of integrating artificial intelligence to improve resilience and response against cyber threats in real environments.

8 Glossari

1. **SIEM:** Sistema de gestió d'informació i esdeveniments de seguretat.
2. **Logs:** Registres d'activitat generats per sistemes, aplicacions o dispositius.
3. **Firewall:** Sistema de seguretat que controla el trànsit de xarxa entrant i sortint.
4. **IDS/IPS:** Sistemes de detecció i prevenció d'intrusions.
5. **PFsense:** Solució de firewall basada en FreeBSD.
6. **Suricata:** Eina d'anàlisi de trànsit i detecció d'intrusions.
7. **Parser**: Component que transforma logs en format estructurat.
8. **Normalització**: Conversió de formats diversos a un format unificat.
9. **Regex**: Expressions regulars per analitzar i extreure informació de text.
10. **BaseParser**: Classe base per crear parsers específics.
11. **JSON**: Format de dades llegible estructurat en clau-valor.
12. **Endpoint**: Punt final d'una API per interactuar amb dades.
13. **AI Analysis:** Anàlisi basada en tècniques d'intel·ligència artificial.
14. **Machine Learning**: Aprenentatge automàtic a partir de dades.
15. **Deep Learning:** Aprenentatge amb xarxes neuronals profundes.
16. **Vector de característiques:** Representació numèrica d'un log.
17. **Ports**: Identificadors de serveis en una xarxa.
18. **IP**: Identificador únic d'un dispositiu en una xarxa.
19. **Protocol**: Conjunt de normes per a la comunicació entre dispositius.
20. **Timestamp**: Marca temporal d'un esdeveniment.
21. **Anomalia**: Esdeveniment que es desvia del comportament esperat.
22. **Correlació d'esdeveniments:** Relació entre diferents esdeveniments per detectar atacs.
23. **Cache**: Emmagatzematge temporal per accelerar consultes.
24. **Emmagatzematge**: Component que conserva dades processades.
25. **Chain of Responsibility:** Patró que passa una petició per una cadena.

26. **Cyber Kill Chain:** Model de fases d'un atac cibernètic.

27. **Heurística**: Mètode per identificar patrons.

28. **Model predictiu:** Algoritme que anticipa comportaments futurs.

29. **Dataset:** Conjunt de dades utilitzat per entrenar o avaluar models.

30. **Validació de dades:** Comprovació que la informació és coherent i completa.

31. **Finestra temporal:** Interval en què s'analitzen esdeveniments relacionats.

32. **Interfície d'usuari (UI):** Part visual amb què interactua l'usuari.

33. **Dashboard:** Panell gràfic que mostra dades o alertes.

34. **Alertes:** Notificacions automàtiques per esdeveniments crítics.

35. **Entorn de producció:** Sistema operatiu real en què es desplega una aplicació.

36. **Docker:** Plataforma de contenidors per empaquetar aplicacions.

37. **Kubernetes:** Sistema per orquestrar contenidors.

38. **Microserveis:** Arquitectura basada en serveis independents.

39. **Logs estructurats:** Registres amb format predefinit i fàcilment analitzable.

40. **Logs no estructurats:** Registres desordenats, difícils de processar.

41. **Format syslog:** Estàndard per a logs de sistemes Unix-like.

42. **Metadades:** Informació addicional sobre un fitxer o esdeveniment.

43. **Recollida de dades:** Captura d'informació des de diverses fonts.

44. **Sistema modular**: Format per components independents però interconnectats.

45. **Escalabilitat**: Capacitat d'un sistema de créixer o adaptar-se.

46. **Persistència**: Emmagatzematge permanent de dades.

9 Bibliografia

- Revista Byte TI. (2024). Ciberseguridad 2025: claves y tendencias. https://revistabyte.es/actualidad-it/ciberseguridad-2025-3/
- Isnum. (n.d). Malas prácticas de ciberseguridad y riesgos en el trabajo. https://isnum.com/malas-practicas-de-ciberseguridad-y-riesgos-en-el-trabajo/
- Akamai. (2023). Ransomware on the Move: Evolving Exploitation Techniques. https://www.akamai.com/blog/security-research/ransomware-on-the-move-evolving-exploitation-techniques
- Agència Nacional de Ciberseguretat d'Andorra. (2025). Prediccions i tendències clau en ciberseguretat per al 2025. https://www.anc.ad/prediccions-i-tendencies-clau-ciberseguretat-per-al-2025-2/
- Ara Andorra. (2024). Miler d'atacs hacker a Andorra el 2024. https://www.ara.ad/societat/miler-d-atacs-hacker-andorra-2024_1_5286826.html
- Digital Andorra. (2024). Andorra va patir més de mil ciberatacs el 2024. https://digitalandorra.com/societat/andorra-va-patir-mes-de-mil-ciberatacs-el-2024/
- PymeLegal. (n.d). Normativa andorrana de protección de datos (RGPD). https://www.pymelegal.es/noticias/rgpd/normativa-andorrana-de-proteccion-de-datos
- Andorra Difusió. (2024). Més de mil ciberatacs aquest any. https://www.andorradifusio.ad/noticies/mes-mil-ciberatacs-any
- IONOS. (n.d). Seguridad de endpoint: qué es y cómo funciona. https://www.ionos.es/digitalguide/servidores/seguridad/endpoint-security/
- IONOS. (n.d). ¿Qué es SIEM y para qué sirve?. https://www.ionos.es/digitalguide/servidores/seguridad/que-es-siem/
- IBM. (n.d). SIEM (Security Information and Event Management). https://www.ibm.com/mx-es/topics/siem
- eSecurityPlanet. (2024). Top SIEM Tools & Software 2024. https://www.esecurityplanet.com/products/siem-tools/
- Splashtop. (2024). What is SIEM and How Does it Work?. https://www.splashtop.com/es/blog/what-is-siem-security-information-and-event-managementheading-6
- Exabeam. (2024). What Is Log Analysis? Process, Techniques, and Best Practices. https://www.exabeam.com/explainers/log-management/what-is-log-analysis-process-techniques-and-best-practices/
- ManageEngine. (2024). What is Log Correlation?. https://www.manageengine.com/products/eventlog/logging-guide/what-is-log-correlation.html
- Panther. (2024). Zero False Positives from Your SIEM. https://panther.com/blog/zero-false-positives-from-your-siem
- IgniteTechnology. (2024). The Evolution of Cyber Threats: Staying Ahead in 2024. https://ignitetechnology.com/the-evolution-of-cyber-threats-staying-ahead-in-2024/

- Cato Networks. (n.d). Network Security Architecture. https://www.catonetworks.com/network-security/network-security-architecture/
- Creately. (n.d). Network Security Diagram Example. https://creately.com/diagram/example/fbdvo93cFek/network-security-diagram-example
- Wikipedia. (2023). Requisito (sistemas). https://es.wikipedia.org/wiki/Requisito_(sistemas)
- RedesZone. (2024). Configurar IDS e IPS en pfSense, firewall profesional. https://www.redeszone.net/tutoriales/seguridad/pfsense-firewall-profesional-configuracion/388954-ids-ips
- Netskope.(n.d). Cadenade eliminación de seguridad cibernètica. https://www.netskope.com/es/security-defined/cyber-security-kill-chain
- ChatGPT. (2025). Millora del informe https://chatgpt.com/
- Claude. (2025). Millora i ajuda d'implementació. https://claude.ai/

10 Annexos

Annex 1: Exemples de logs sense normalitzar

PFsense:

```
May 26 08:37:08 pfsense filterlog:
5,,,1000001214,igb0,match,pass,out,4,0x0,,64,0,0,DF,6,tcp,52,192.168.10.1
8,google.com,64370,993,0,S,1234569004,,1460,,mss;nop;nop;sackOK;nop;wscal
e

May 26 08:37:10 pfsense filterlog:
5,,,1000001215,igb0,match,pass,out,4,0x0,,64,0,0,DF,6,tcp,52,192.168.10.3
8,github.com,58796,80,0,S,1234569005,,1460,,mss;nop;nop;sackOK;nop;wscale

May 26 08:37:12 pfsense filterlog:
5,,,1000001216,igb0,match,pass,out,4,0x0,,64,0,0,DF,6,tcp,52,192.168.10.1
3,stackoverflow.com,65347,25,0,S,1234569006,,1460,,mss;nop;nop;sackOK;nop
;wscale

May 26 08:37:14 pfsense filterlog:
5,,,1000001217,igb0,match,pass,out,4,0x0,,64,0,0,DF,6,tcp,52,192.168.10.9
7,1.1.1.1,52417,443,0,S,1234569007,,1460,,mss;nop;nop;sackOK;nop;wscale

May 26 08:37:16 pfsense filterlog:
5,,,1000001218,igb0,match,pass,out,4,0x0,,64,0,0,DF,6,tcp,52,192.168.10.2
4,google.com,61124,80,0,S,1234569008,,1460,,mss;nop;nop;sackOK;nop;wscale

May 26 08:37:18 pfsense filterlog:
5,,,1000001219,igb0,match,pass,out,4,0x0,,64,0,0,DF,6,tcp,52,192.168.10.9
2,8.8.8.8,58896,995,0,S,1234569009,,1460,,mss;nop;nop;sackOK;nop;wscale

May 26 08:37:20 pfsense filterlog:
5,,,1000001220,igb0,match,pass,out,4,0x0,,64,0,0,DF,6,tcp,52,192.168.10.4
2,outlook.office365.com,59876,995,0,S,1234569010,,1460,,mss;nop;nop;sackO
K;nop;wscale

May 26 08:37:22 pfsense filterlog:
5,,,1000001221,igb0,match,pass,out,4,0x0,,64,0,0,DF,6,tcp,52,192.168.10.2
6,salesforce.com,55523,993,0,S,1234569011,,1460,,mss;nop;nop;sackOK;nop;w
scale

May 26 08:37:24 pfsense filterlog:
5,,,1000001222,igb0,match,pass,out,4,0x0,,64,0,0,DF,6,tcp,52,192.168.10.7
6,teams.microsoft.com,53225,995,0,S,1234569012,,1460,,mss;nop;nop;sackOK;
nop;wscale

May 26 08:37:26 pfsense filterlog:
5,,,1000001223,igb0,match,pass,out,4,0x0,,64,0,0,DF,6,tcp,52,192.168.10.1
4,8.8.8.8,51092,443,0,S,1234569013,,1460,,mss;nop;nop;sackOK;nop;wscale

May 26 08:37:28 pfsense filterlog:
5,,,1000001224,igb0,match,pass,out,4,0x0,,64,0,0,DF,6,tcp,52,192.168.10.4
1,adobe.com,57039,993,0,S,1234569014,,1460,,mss;nop;nop;sackOK;nop;wscale
```

Suricata:


```
{"timestamp":"2025-05-
26T08:58:42Z","flow_id":1621479841881174,"in_iface":"eth0","event_type":"
alert","src_ip":"192.168.10.104","src_port":51700,"dest_ip":"198.51.100.8
8","dest_port":443,"proto":"TCP","alert":{"action":"allowed","gid":1,"sig
nature_id":2025003,"rev":1,"signature":"ET POLICY Large Data Transfer -
Potential Data Exfiltration","category":"Potentially Bad
Traffic","severity":2},"http":{"hostname":"exfil-
server.com","url":"/upload/database_dump.zip","http_user_agent":"curl/7.6
8.0","http_method":"POST","status":200,"length":52428800}}

{"timestamp":"2025-05-
26T08:58:45Z","flow_id":1621479841881175,"in_iface":"eth0","event_type":"
alert","src_ip":"192.168.10.15","src_port":56669,"dest_ip":"198.51.100.88
","dest_port":443,"proto":"TCP","alert":{"action":"allowed","gid":1,"sign
ature_id":2025005,"rev":1,"signature":"ET TROJAN Command and Control
Communication - Encrypted Channel","category":"A Network Trojan was
detected","severity":1},"tls":{"subject":"CN=legitimate-looking-
domain.com","issuerdn":"CN=FakeCA","fingerprint":"aa:bb:cc:dd:ee:ff:00:11
:22:33:44:55:66:77:88:99:aa:bb:cc:dd"}}

{"timestamp":"2025-05-
26T08:58:48Z","flow_id":1621479841881176,"in_iface":"eth0","event_type":"
alert","src_ip":"192.168.10.93","src_port":56785,"dest_ip":"198.51.100.88
","dest_port":443,"proto":"TCP","alert":{"action":"allowed","gid":1,"sign
ature_id":2025001,"rev":1,"signature":"ET MALWARE Advanced Persistent
Threat - Initial Payload Download","category":"A Network Trojan was
detected","severity":1},"http":{"hostname":"malicious-c2-
server.com","url":"/download/stage1.exe","http_user_agent":"Mozilla/5.0
(Windows NT 10.0; Win64;
x64)","http_method":"GET","status":200,"length":2048576}}

{"timestamp":"2025-05-
26T08:58:51Z","flow_id":1621479841881177,"in_iface":"eth0","event_type":"
alert","src_ip":"192.168.10.15","src_port":61984,"dest_ip":"192.168.10.11
4","dest_port":445,"proto":"TCP","alert":{"action":"allowed","gid":1,"sig
nature_id":2025002,"rev":1,"signature":"ET TROJAN SMB Lateral Movement -
Credential Harvesting","category":"A Network Trojan was
detected","severity":1}}

{"timestamp":"2025-05-
26T08:58:54Z","flow_id":1621479841881178,"in_iface":"eth0","event_type":"
alert","src_ip":"192.168.10.15","src_port":49887,"dest_ip":"192.168.10.11
8","dest_port":445,"proto":"TCP","alert":{"action":"allowed","gid":1,"sig
nature_id":2025002,"rev":1,"signature":"ET TROJAN SMB Lateral Movement -
Credential Harvesting","category":"A Network Trojan was
detected","severity":1}}

{"timestamp":"2025-05-
26T08:58:57Z","flow_id":1621479841881179,"in_iface":"eth0","event_type":"
alert","src_ip":"203.0.113.45","src_port":12345,"dest_ip":"192.168.10.100
","dest_port":135,"proto":"TCP","alert":{"action":"allowed","gid":1,"sign
ature_id":2001219,"rev":20,"signature":"ET SCAN Advanced Port Scan -
Multiple Targets","category":"Attempted Information Leak","severity":2}}
```

Annex 2: Exemple de log normalitzat amb totes les dades


```
{
  "timestamp": {
    "$date": "2025-05-26T08:02:32.000Z"
  },
  "source_type": "pfsense",
  "raw_log":      "May      26      08:02:32      pfsense      filterlog:
5,,,1000000176,igb0,match,pass,out,4,0x0,,64,0,0,DF,6,tcp,52,192.168.10.1
4,1.1.1.1,59046,80,0,S,1234567966,,1460,,mss;nop;nop;sackOK;nop;wscale",
  "event_category": "network",
  "dns_answer": null,
  "event_description":    "PFSense    pass    tcp    connection    from
192.168.10.14:59046 to 1.1.1.1:80",
  "firewall_action": "pass",
  "process_id": null,
  "event_type": "connection_allowed",
  "dst_ip": "1.1.1.1",
  "additional_fields": {
    "rule_number": "5",
    "interface": "igb0",
    "direction": "out",
    "protocol_id": "6",
    "packet_size": "52",
    "tcp_data": "0",
    "tcp_flags": "S",
    "port_scan_likelihood": 0
  },
  "src_port": "59046",
  "dst_port": "80",
  "_id": {
    "$oid": "684607b3c341678e303950b7"
  },
  "event_severity": "info",
  "user_id": null,
  "ids_signature": null,
  "event_id": "85e7450a-8ee6-4c6c-996b-b697c454d0ee",
  "http_status": null,
  "protocol": "tcp",
  "source_name": "pfsense",
  "src_ip": "192.168.10.14",
  "src_mac": null,
  "process_name": null,
  "dns_query": null,
  "source_ip": null,
  "dst_mac": null,
  "http_url": null,
  "http_method": null,
  "ai_analysis": {
    "analysis_timestamp": "2025-06-08T23:59:11.780487",
    "anomaly_score": 0,
    "detection_methods": [],
    "confidence": 0,
    "classification": "normal",
```

```
    "detected_threats": [],
    "ml_analysis": {
      "isolation_forest": {
        "anomaly_score": 0.6152813815380412,
        "is_anomaly": false,
        "method": "isolation_forest"
      },
      "neural_network": {
        "anomaly_score": 0,
        "is_anomaly": false,
        "method": "neural_network"
      }
    }
  }
}
```

Annex 3: Exemple complet d'una alerta generada amb tots els logs involucrats


```
{
  "_id": {
    "$oid": "68460b6f94713874ace7873d"
  },
  "id": "ALT-1749420906-4857",
  "timestamp": {
    "$date": "2025-05-26T08:39:12.000Z"
  },
  "source_type": "pfsense",
  "event_type": "connection_blocked",
  "severity": "medium",
  "title": "Escaneo de puertos detectado desde 91.234.99.12",
  "description": "PFSense block tcp connection from 91.234.99.12:12345 to
192.168.10.116:55793",
  "attack_type": "port_scan",
  "attack_details": {
    "source_ip": "91.234.99.12",
    "target_ip": "192.168.10.116",
    "target_port": "55793",
    "protocol": "tcp",
    "severity": 0.5,
    "confidence": 0.7
  },
  "details": {
    "log": {
      "timestamp": "2025-05-26T08:39:12Z",
      "source_type": "pfsense",
      "source_name": "pfsense",
      "source_ip": null,
      "event_id": "44463353-ad61-4502-9aef-869cc4155fcf",
      "event_type": "connection_blocked",
      "event_severity": "medium",
      "event_category": "network",
      "event_description":    "PFSense    block    tcp    connection    from
91.234.99.12:12345 to 192.168.10.116:55793",
      "src_ip": "91.234.99.12",
      "dst_ip": "192.168.10.116",
      "src_port": "12345",
      "dst_port": "55793",
      "protocol": "tcp",
      "src_mac": null,
      "dst_mac": null,
      "firewall_action": "block",
      "ids_signature": null,
      "http_method": null,
      "http_url": null,
      "http_status": null,
      "dns_query": null,
      "dns_answer": null,
      "user_id": null,
```

```
      "process_id": null,
      "process_name": null,
      "raw_log":     "May     26     08:39:12     pfsense     filterlog:
5,,,1000001276,igb0,match,block,in,4,0x0,,64,0,0,DF,6,tcp,40,91.234.99.12
,192.168.10.116,12345,55793,0,S,1234569066,,1460,,",
      "additional_fields": {
        "rule_number": "5",
        "interface": "igb0",
        "direction": "in",
        "protocol_id": "6",
        "packet_size": "40",
        "tcp_data": "0",
        "tcp_flags": "S",
        "port_scan_likelihood": 0
      },
      "ai_analysis": {
        "analysis_timestamp": "2025-06-09T00:15:06.546316",
        "anomaly_score": 1,
        "detection_methods": [
          "neural_network"
        ],
        "confidence": 0.75,
        "classification": "suspicious",
        "detected_threats": [
          "ml_anomaly"
        ],
        "ml_analysis": {
          "isolation_forest": {
            "anomaly_score": 0.5482844520736629,
            "is_anomaly": false,
            "method": "isolation_forest"
          },
          "neural_network": {
            "anomaly_score": 1,
            "is_anomaly": true,
            "method": "neural_network"
          }
        }
      }
    },
    "analysis": {
      "basic_analysis": {
        "severity": 0.5,
        "confidence": 0.7,
        "indicators": [
          "port_scan"
        ],
        "threats": [
          "reconnaissance"
        ],
        "port_scan_likelihood": 0.8,
        "attack_details": {
          "type": "port_scan",
          "source_ip": "91.234.99.12",
```

```
          "target_ip": "192.168.10.116",
          "target_port": "55793"
        }
      },
      "ai_analysis": {
        "analysis_timestamp": "2025-06-09T00:15:06.546316",
        "anomaly_score": 1,
        "detection_methods": [
          "neural_network"
        ],
        "confidence": 0.75,
        "classification": "suspicious",
        "detected_threats": [
          "ml_anomaly"
        ],
        "ml_analysis": {
          "isolation_forest": {
            "anomaly_score": 0.5482844520736629,
            "is_anomaly": false,
            "method": "isolation_forest"
          },
          "neural_network": {
            "anomaly_score": 1,
            "is_anomaly": true,
            "method": "neural_network"
          }
        }
      },
      "correlation": {
        "related_events": 2,
        "risk_score": 0.94875,
        "patterns": [
          {
            "name": "ai_enhanced_brute_force",
            "description": "Ataque de fuerza bruta detectado con IA
(métodos: ai_anomaly)",
            "confidence": 0.3,
            "severity": "medium",
            "attack_type": "brute_force",
            "detection_methods": [
              "ai_anomaly"
            ],
            "auth_events": 0,
            "anomaly_events": 2,
            "ai_metrics": {
              "avg_anomaly_score": 0.6666666666666666,
              "suspicious_ratio": 0.6666666666666666,
              "total_ai_detections": 3
            }
          },
          {
            "name": "ai_enhanced_malware",
            "description": "Malware detectado con IA (métodos:
ai_detection, ai_threat_classification)",
```

```
            "confidence": 0.6000000000000001,
            "severity": "high",
            "attack_type": "malware",
            "detection_methods": [
              "ai_detection",
              "ai_threat_classification"
            ],
            "traditional_indicators": 0,
            "ai_detections": 2,
            "threat_types": [
              "ml_anomaly"
            ],
            "ai_metrics": {
              "avg_anomaly_score": 0.6666666666666666,
              "threat_diversity": 1
            }
          },
          {
            "name": "ai_enhanced_reconnaissance",
            "description": "Reconnaissance detectado con IA (métodos:
ai_network_analysis)",
            "confidence": 0.4,
            "severity": "medium",
            "attack_type": "reconnaissance",
            "detection_methods": [
              "ai_network_analysis"
            ],
            "scan_indicators": 0,
            "network_anomalies": 2,
            "ai_metrics": {
              "avg_anomaly_score": 0.6666666666666666
            }
          }
        ],
        "campaign": null,
        "campaign_confidence": 0,
        "ai_correlation": {
          "avg_anomaly_score": 0.6666666666666666,
          "max_anomaly_score": 1,
          "avg_confidence": 0.5,
          "suspicious_ratio": 0.6666666666666666,
          "anomaly_ratio": 0,
          "threat_diversity": 1,
          "total_ai_detections": 3,
          "threat_types": [
            "ml_anomaly"
          ]
        },
        "attack_sequence": [
          {
            "phase": "persistence",
            "timestamp": "2025-06-09T00:15:06.389042",
            "attack_type": "malware",
            "anomaly_score": 1,
```

```
            "confidence": 0.75,
            "event_id": "d3385bb9-8998-4587-abc6-777b46bab257"
          },
          {
            "phase": "persistence",
            "timestamp": "2025-06-09T00:15:06.691566",
            "attack_type": "malware",
            "anomaly_score": 1,
            "confidence": 0.75,
            "event_id": "44463353-ad61-4502-9aef-869cc4155fcf"
          }
        ],
        "threat_level": "critical",
        "correlation_confidence": 0.8633333333333333
      }
    },
    "risk_score": 0.94875,
    "anomaly_score": 1,
    "confidence": 0.75
  },
  "correlation": {
    "patterns": [
      {
        "name": "ai_enhanced_brute_force",
        "description": "Ataque de fuerza bruta detectado con IA (métodos:
ai_anomaly)",
        "confidence": 0.3,
        "severity": "medium",
        "attack_type": "brute_force",
        "detection_methods": [
          "ai_anomaly"
        ],
        "auth_events": 0,
        "anomaly_events": 2,
        "ai_metrics": {
          "avg_anomaly_score": 0.6666666666666666,
          "suspicious_ratio": 0.6666666666666666,
          "total_ai_detections": 3
        }
      },
      {
        "name": "ai_enhanced_malware",
        "description": "Malware detectado con IA (métodos: ai_detection,
ai_threat_classification)",
        "confidence": 0.6000000000000001,
        "severity": "high",
        "attack_type": "malware",
        "detection_methods": [
          "ai_detection",
          "ai_threat_classification"
        ],
        "traditional_indicators": 0,
        "ai_detections": 2,
        "threat_types": [
```

```
          "ml_anomaly"
        ],
        "ai_metrics": {
          "avg_anomaly_score": 0.6666666666666666,
          "threat_diversity": 1
        }
      },
      {
        "name": "ai_enhanced_reconnaissance",
        "description": "Reconnaissance detectado con IA (métodos:
ai_network_analysis)",
        "confidence": 0.4,
        "severity": "medium",
        "attack_type": "reconnaissance",
        "detection_methods": [
          "ai_network_analysis"
        ],
        "scan_indicators": 0,
        "network_anomalies": 2,
        "ai_metrics": {
          "avg_anomaly_score": 0.6666666666666666
        }
      }
    ],
    "campaigns": [],
    "mitre_techniques": [],
    "risk_score": 0.94875
  },
  "status": "new",
  "priority": 1,
  "tags": [
    "severity:medium",
    "source:pfsense",
    "event:connection_blocked",
    "scenario:{'type': 'port_scan', 'source_ip': '91.234.99.12',
'target_ip': '192.168.10.116', 'target_port': '55793', 'severity': 0.5,
'confidence': 0.7}",
    "src_ip:91.234.99.12"
  ],
  "raw_log": "May 26 08:39:12 pfsense filterlog:
5,,,1000001276,igb0,match,block,in,4,0x0,,64,0,0,DF,6,tcp,40,91.234.99.12
,192.168.10.116,12345,55793,0,S,1234569066,,1460,,"
}
```