ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

Trabajo de Integración Curricular previo a la obtención del título de Ingeniero en Tecnologías de la Información

Tema: Diseño e Implementación de una plataforma XSOAR en Docker: Automatización de detección y respuesta a incidentes en el laboratorio de Ciberseguridad.

Autor: Cuadrado Avilés, Santiago Alejandro

Tutor: Ing. Bustos Sánchez, Cristian Santiago, PhD.

Sangolquí, 6 de Agosto de 2026



Tabla de Contenido

1. Introducción y fundamentos

2. Metodología y Arquitectura

3. Implementación e integraciones

4. Evaluación y Resultados

5. Conclusiones y Trabajos Futuros

Introducción

Punto de partida

Laboratorio en etapa inicial

Infraestructura base

Sin Automatización

No existía aplicaciones de orquestación automática

Sin Monitoreo

Sin herramientas de supervisión

Oportunidad de Construcción

Entorno desde cero

Necesidad Identificada

Necesidad de un entorno Integrado
Supervisar eventos
Centralizar alertas
Analizar indicadores
Coordinar acciones de respuesta
Generar evidencias

Planteamiento del Problema y Objetivos

Planteamiento del Problema

- Ausencia de monitoreo centralizado
- Procesos manuales ante incidentes
- Falta de análisis integrado de indicadores
- Sin métricas para evaluar la respuesta

Objetivo General

Diseñar e implementar un entorno SOAR para automatizar la detección, análisis y respuesta ante incidentes en el laboratorio.

Objetivos Específicos

1. Analizar requerimientos
2. Diseñar la arquitectura
3. Implementar el entorno
4. Evaluar mediante escenarios y métricas

Estado del Arte y Marco Teórico

Estado del Arte

Tendencia hacia plataformas SOAR y XSOAR para integrar detección, análisis y respuesta

Wazuh, TheHive, Cortex y MISP son soluciones de código abierto ampliamente adoptadas.

Docker permite entornos aislados, reproducibles y fáciles de desplegar

Estudios demuestran que la automatización reduce tiempos de respuesta y mejora la gestión de incidentes.

Marco Teórico

Integración y automatización de procesos de seguridad para mejorar la detección, análisis y respuesta a incidentes.

Recolección y correlación de eventos para detectar amenazas y generar alertas en tiempo real.

Uso de indicadores y contexto de amenazas para enriquecer el análisis y priorizar incidentes.

Ejecución automática de acciones para contener, investigar y responder incidentes.

Empaquetado y orquestación de servicios para garantizar portabilidad, aislamiento y escalabilidad

Estos conceptos sustentan el diseño e implementación de la plataforma SOAR en el laboratorio, permitiendo una gestión de incidentes más eficiente y automatizada.

Metodología

Enfoque
Tecnológico aplicado

Alcance
Exploratorio y descriptivo

Método
DSRM

Fases de Design Science Research Methodology

Problema → Objetivos → Diseño → Demostración → Evaluación → Comunicación

Diseño de evaluación

Evaluación cuasiexperimental

Carga de archivos: 5 → 10 → 20

Compara MTTD, MTTR, clasificación, automatización y recursos.

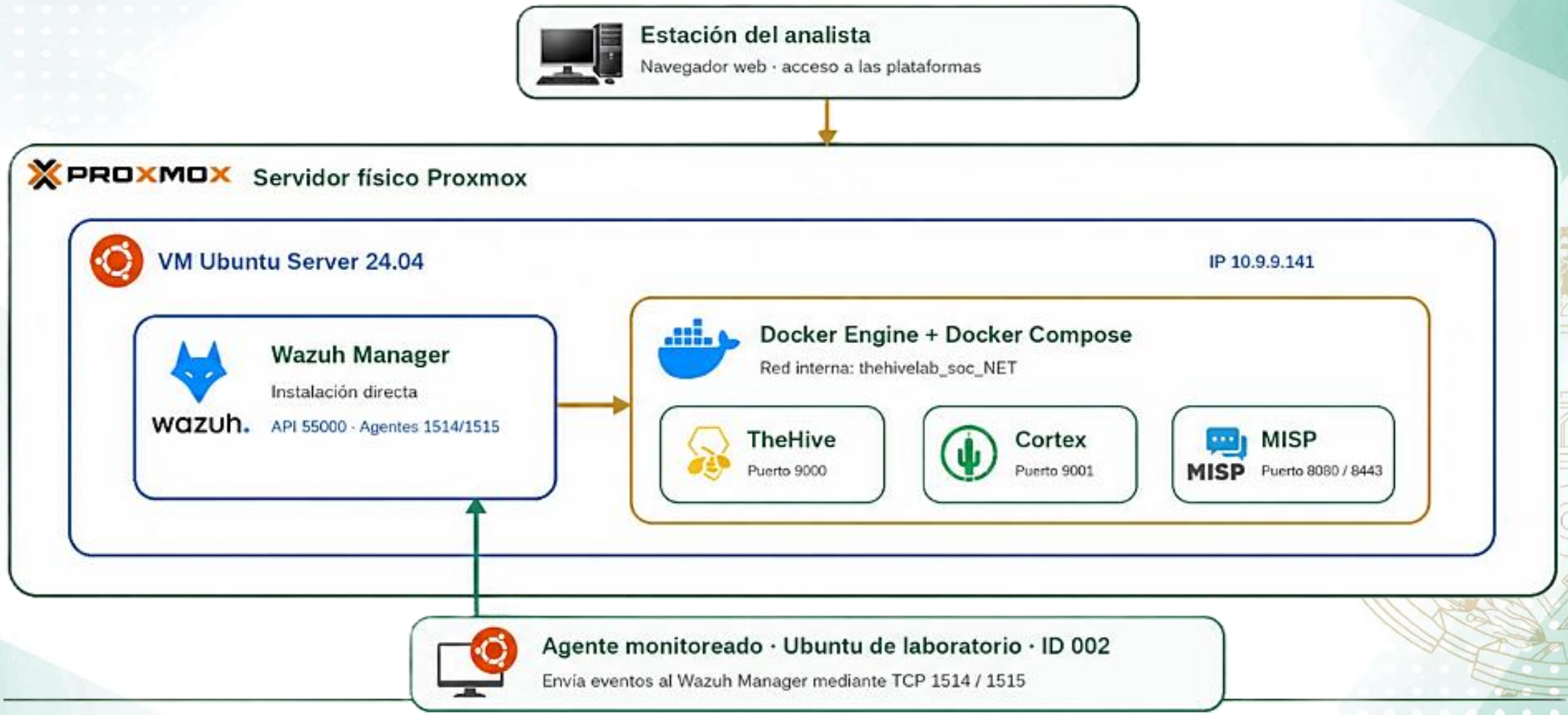
Validación funcional

Fuerza bruta SSH

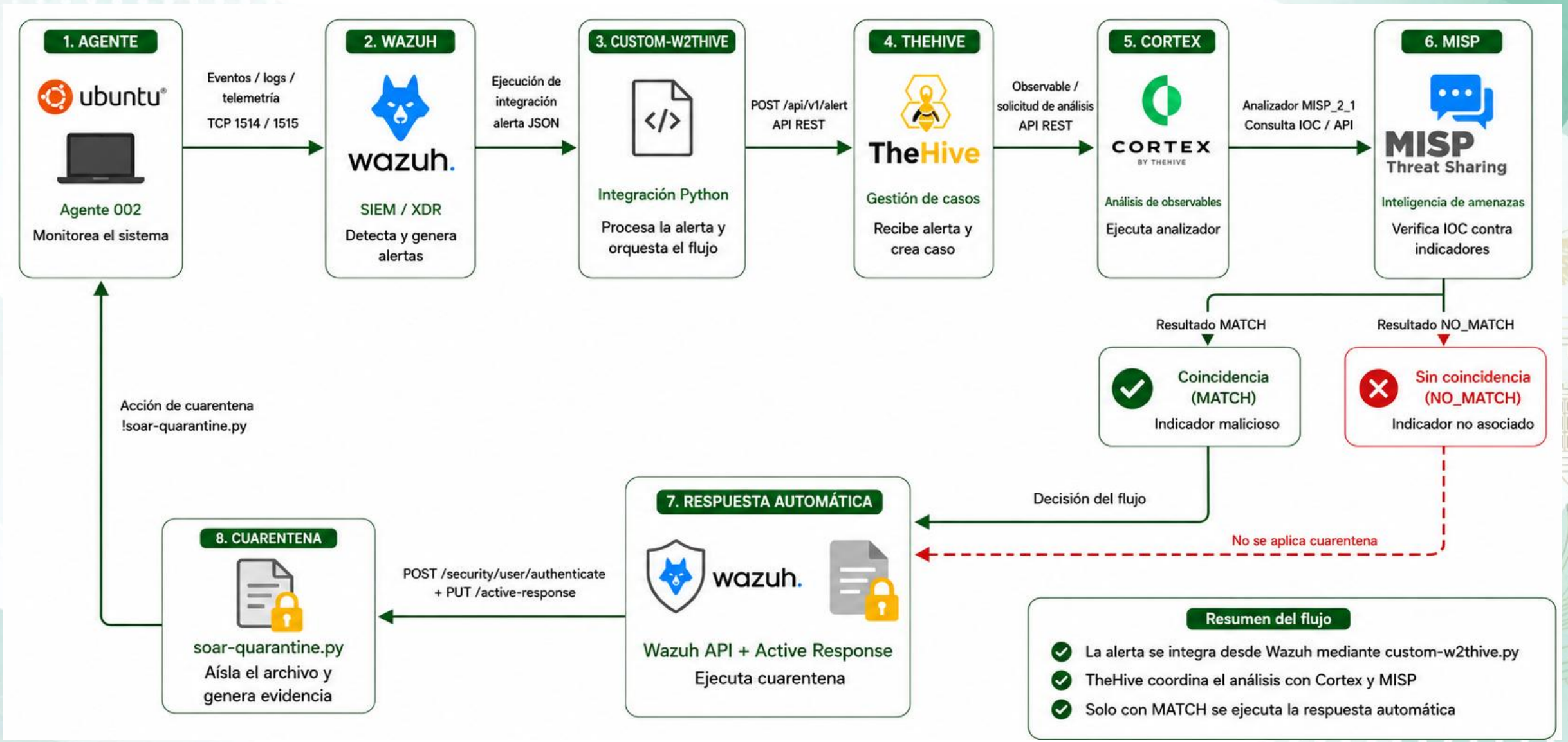
Movimiento no autorizado

Comprueba el flujo ante eventos de distinta naturaleza.

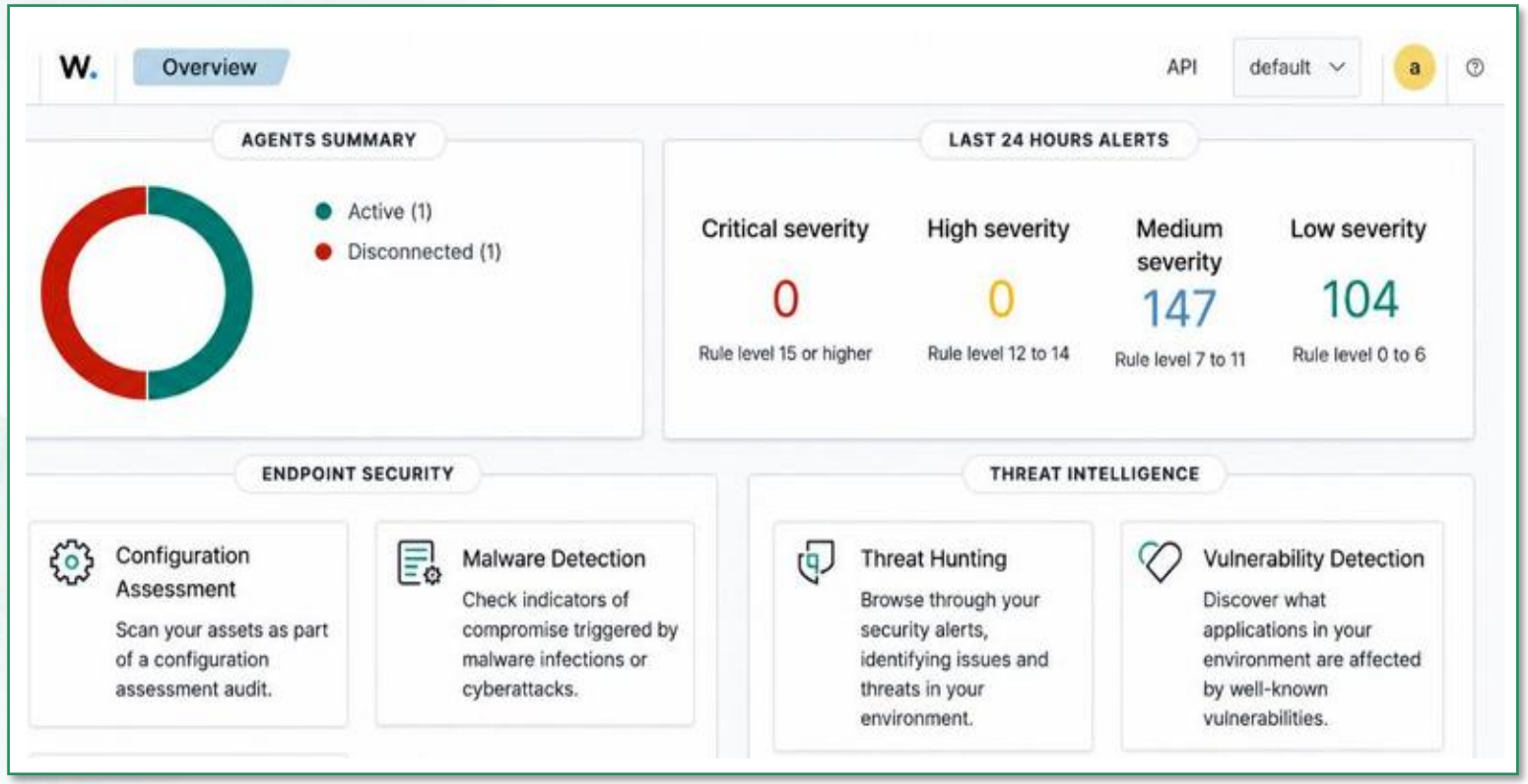
Arquitectura Física



Arquitectura Lógica y comunicaciones



Implementación del Entorno



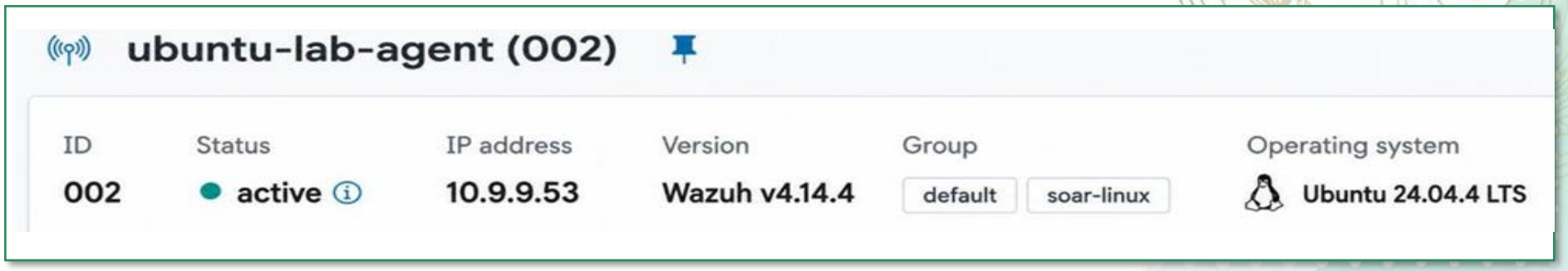
Dashboard Wazuh

Imágenes Docker

```
CONTAINER ID   IMAGE
               PORTS
a610846908fb   strangebee/thehive:5.2
               0.0.0.0:9000->9000/tcp
24392dcf3ec7   coolacid/misp-docker:core-latest
               0.0.0.0:8080->80/tcp, 0.0.0.0:8443->443/tcp
f1519fda4d07   coolacid/misp-docker:modules-latest

431d93109671   thehiveproject/cortex:latest
               0.0.0.0:9001->9001/tcp
```

Agente activo



Integraciones Principales

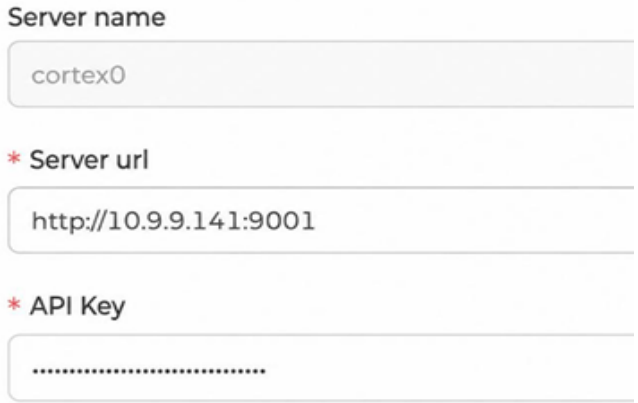
Wazuh - TheHive

custom-w2thhive configurado dentro de ossec.conf

```
82-<integration>
83:     <name>custom-w2thive</name>
84-     <level>5</level>
85-     <hook_url>http://10.9.9.141:9000</hook_url>
86-     <api_key>w4IUKLvtS0cnSSDf4p6fUD0IKUwrKn4L</api_key>
87-     <alert_format>json</alert_format>
88-   </integration>
```

TheHive - Cortex

Servidor Cortex conectado por URL y API Key



Cortex - MISP

Analizador habilitado para consultar observables



Escenarios de Evaluación

Escenario Principal

Archivos Sospechosos



Carga de archivos controlados para medir tiempos, clasificación y recursos.

Archivos	35
Cargas Evaluadas	5 - 10 - 20
Coincidencias MISP	14 / 21

Escenarios Secundarios

Fuerza Bruta SSH



Intentos fallidos de autenticación para validar alerta, análisis y caso.

Intentos fallidos	8
Tiempo observado	120 s
Detección	Exitosa

Movimiento no Autorizado



Archivo protegido trasladado fuera de la ubicación permitida

Tipo de Archivo	PDF simulado
Criterio	Política
Respuesta	Cuarentena

Evaluación cuasiexperimental

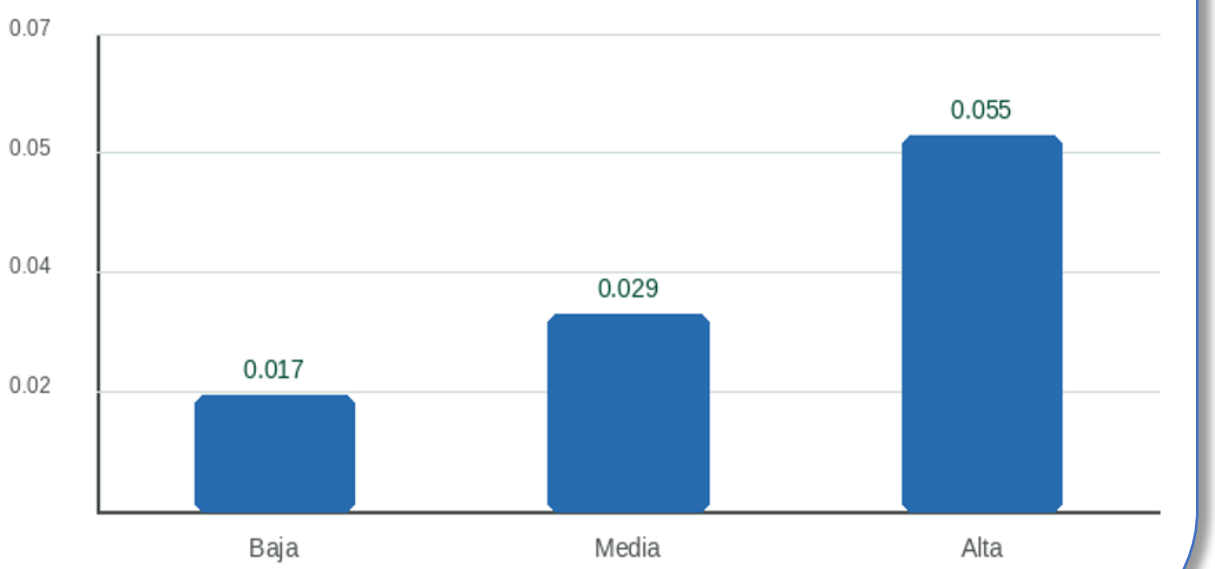
Validación funcional

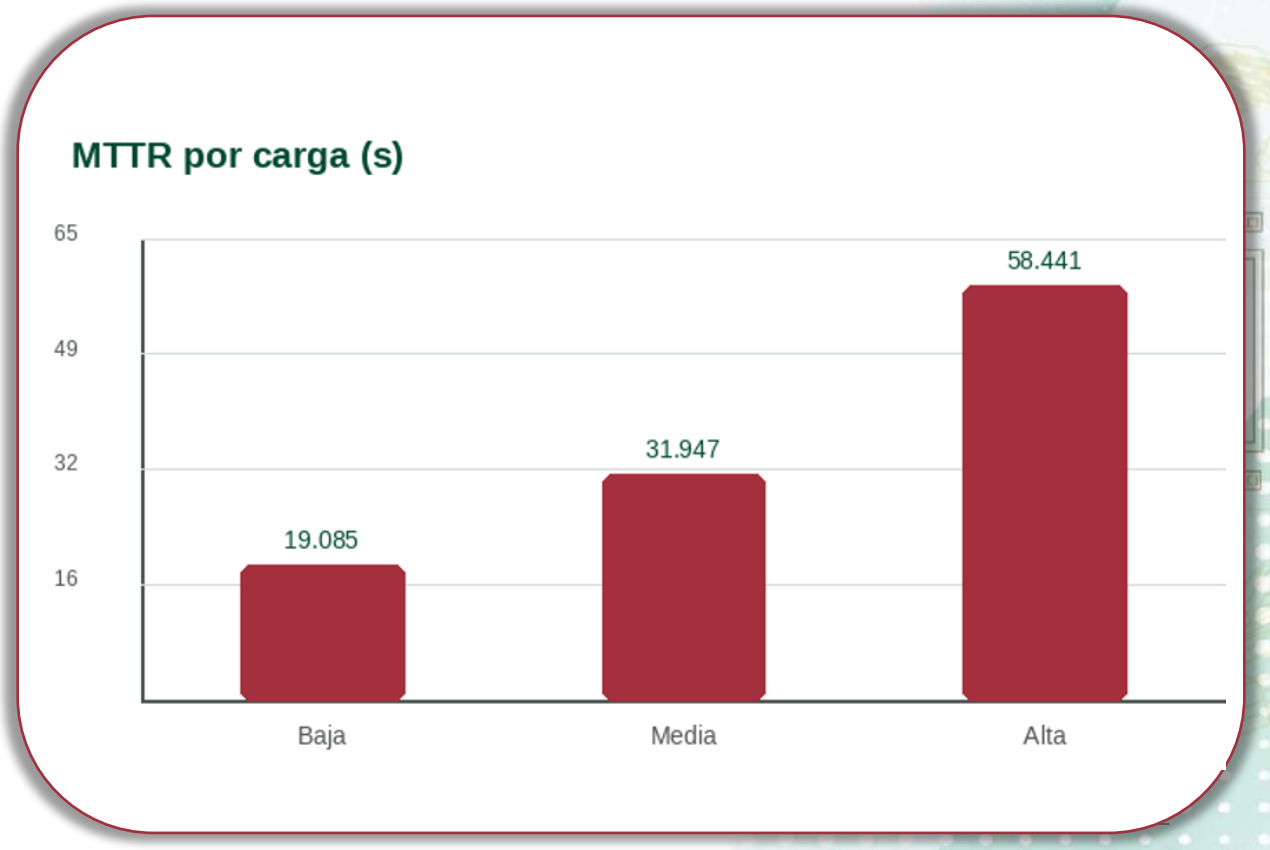
Resultados Obtenidos

Tiempos de detección y Respuesta del Escenario Principal

0,007 S
MTTD promedio.

15,734 S
MTTR promedio.





Resultados Obtenidos

Clasificación y Automatización del flujo

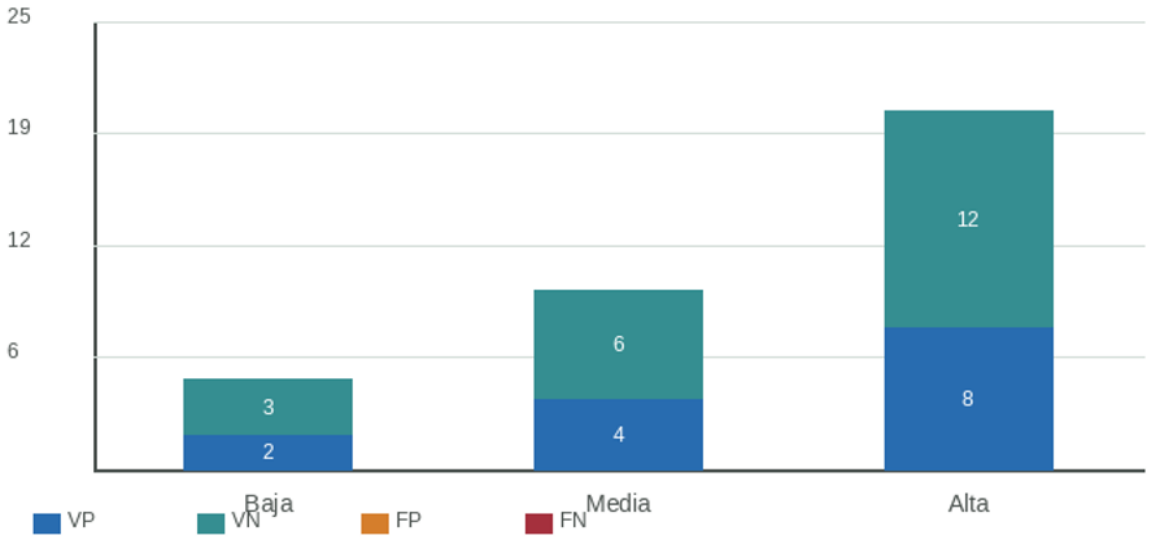
14 VP
Verdaderos positivos

21 VN
Verdaderos negativos

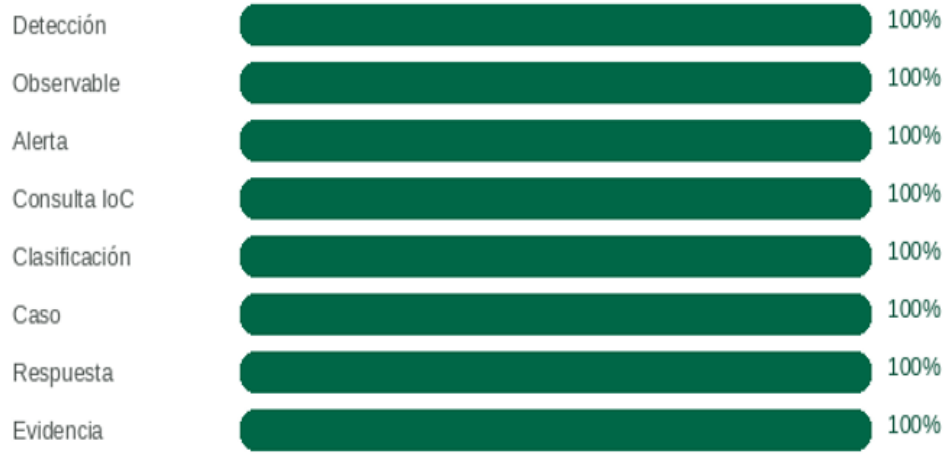
0 FP/FN
Sin falsos resultados

100%
Automatización.

Clasificación por nivel de carga



Automatización del flujo

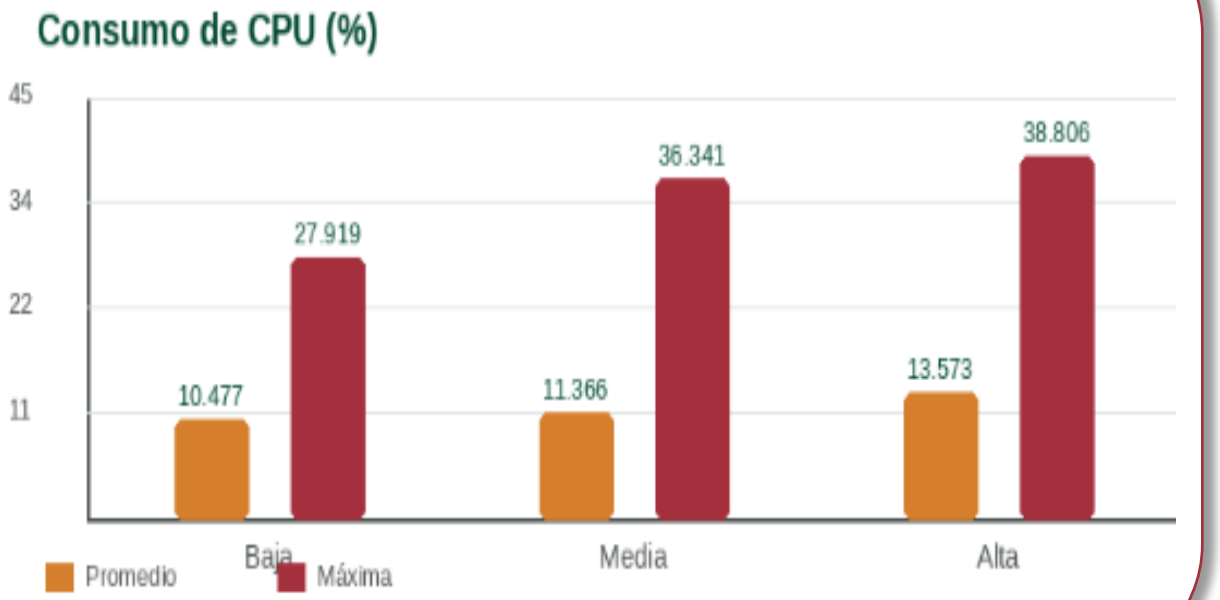


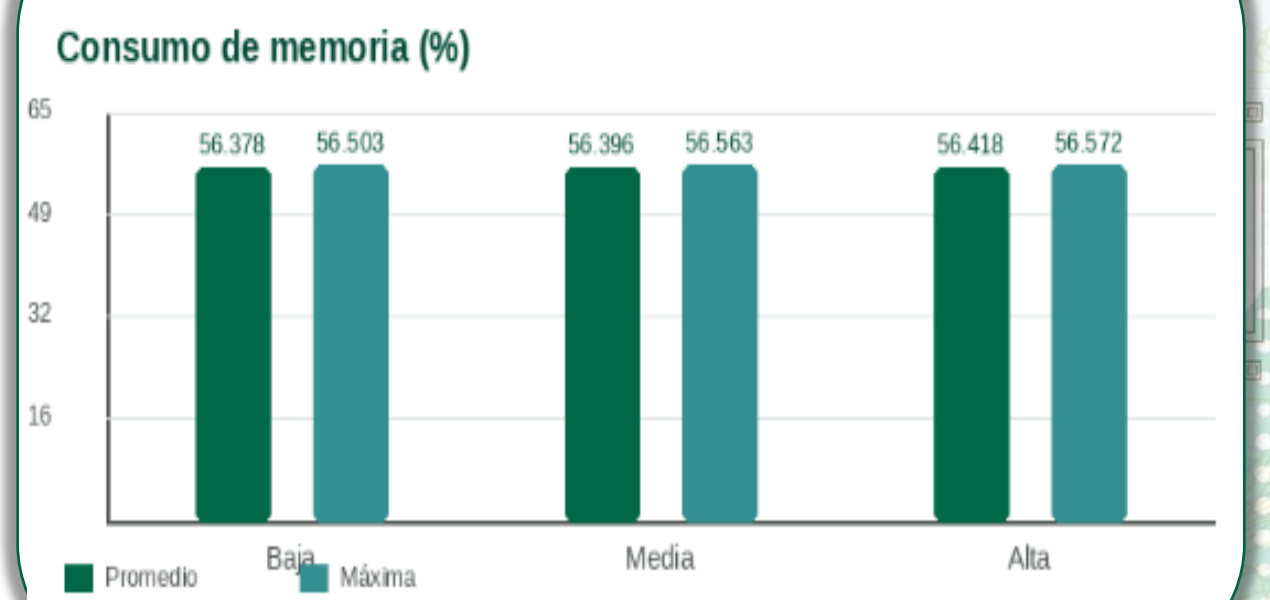
Resultados Obtenidos

Consumo de recursos durante la carga de archivos.

38,806 %
CPU Máxima

56,572 %
Memoria Máxima





Resultados Principales

Escenarios complementarios y recuperación.

Fuerza bruta SSH

8 intentos fallidos en 120 segundos. Coincidencia en MISP, alerta generada, caso registrado en TheHive y tiempo de respuesta de 15,844segundos.

Movimiento no autorizado

PDF simulado movido fuera de la ubicación permitida. Hashes SHA-256, SHA-1 y MD5analizados. Acción aplicada: cuarentena.

Recuperación

Servicios contenerizados recuperados: 3 de 3. Tiempo de recuperación: 56 segundos. Nueva ejecución: 21,82 segundos.

1 Wazuh activo → 2 Agente reconectado → 3 Docker recuperado → 4 Flujo ejecutado

Conclusiones

Síntesis de los hallazgos del proyecto

Entorno funcional

El proyecto logró el diseño e implementación de un entorno SOAR funcional en el laboratorio de ciberseguridad.

Integración efectiva

Wazuh, TheHive, Cortex y MISP trabajaron en un flujo integrado para la detección, análisis y respuesta.

Automatización comprobada

El workflow ejecutó de forma automática la detección, alertamiento, creación casos, análisis de observables y respuesta.

Resultados favorables

La evaluación obtuvo clasificación correcta de los 35 archivos, tiempos de detección bajos y uso estable de recursos.

Recomendaciones y Trabajos Futuros

Recomendaciones

- Procesamiento asíncrono / colas de eventos
- Más ejecuciones y muestras
- Más escenarios y analizadores
- Notificaciones automáticas

Trabajos Futuros

- Arquitectura más distribuida
- Monitoreo ampliado al Departamento de Ciencias de la Computación
- Más herramientas y fuentes: routers, switches, phishing y malware

ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

Gracias por su atención

Preguntas