



TRABAJO DE FIN DE GRADO

Grado en Ingeniería en Tecnologías de

Telecomunicación

ESCUELA POLITÉCNICA SUPERIOR

Desarrollo e Implementación de un Centro de Operaciones de Seguridad (SOC) para la Protección Cibernética en PYMEs y SOHOs

Celia Gómez San Martín

Tutor: Antonio García Herraiz

2025

UNIVERSIDAD DE ALCALÁ

ESCUELA POLITÉCNICA SUPERIOR

Grado en Ingeniería en Tecnologías de Telecomunicación

Trabajo de Fin de Grado

Autor: Celia Gómez San Martín

Tutor: Antonio García Herraiz

Tribunal:

Presidente: Enrique de la Hoz de la Hoz

Vocal 1º: Luis de la Cruz Piris

Fecha de depósito: 15/09/2025

“La mujer ocupará en el mundo científico el puesto que le corresponda de acuerdo con su capacidad, y no necesitaremos cuotas ni nada de eso.”

Margarita Salas.

Agradecimientos

A mi tutor Antonio García Herraiz, por confiar y apoyar este proyecto, sin cuestionarlo desde el principio.

A mis padres, en especial a mi madre, por ser apoyo incondicional, por acompañarme en todo el camino y sufrirlo conmigo.

Mamá, gracias por ser refugio en los momentos más difíciles, por tu paciencia y por recordarme siempre que si podía.

A mis compañeros de vida, mis amigos, Laura, Rebeca, Adrián, Juancar, Sergio y Rubén sin los que este recorrido no habría tenido sentido.

A mi abuela Amparo, la mujer más inteligente. Pionera en las mujeres científicas de su generación, que siempre me inspira en todo lo que hago en la vida.

A mi familia, pero especialmente a mis tíos Rodolfo, Enrique y Amparo, quienes estarían muy orgullosos de verme siendo ingeniera.

A la Universidad de Alcalá y el claustro de profesores de la Escuela Politécnica por brindarme el espacio académico y personal en el que pude crecer, aprender y formarme, no solo como profesional, sino también como persona.

Y por último a mí misma, por mi perseverancia, por mi esfuerzo y valentía para no rendirme incluso cuando me dijeron que no podía.

Resumen

Este trabajo desarrolla un Centro de Operaciones de Seguridad (SOC) adaptado a PYMEs, proponiendo un laboratorio práctico y replicable con software libre. El objetivo es demostrar cómo implementar un SOC mediante tecnologías accesibles y escalables, fomentando un enfoque proactivo en ciberdefensa. Se busca mitigar riesgos y ofrecer un marco que fortalezca la resiliencia empresarial frente a incidentes.

La metodología incluye un estudio inicial de ciberseguridad, el diseño del laboratorio en AWS con herramientas open-source y la validación mediante ataques simulados.

Este enfoque responde a la falta de soluciones asequibles, ya que muchas PYMEs carecen de capacidades de monitorización.

Abstract

This coursework develops a Security Operations Center (SOC) adapted to SMEs, proposing a practical and replicable laboratory with open-source software. The objective is to demonstrate how to implement a SOC using accessible and scalable technologies, fostering a proactive approach to cyber defense. It seeks to mitigate risks and provide a framework that strengthens business resilience against incidents.

The methodology includes an initial cybersecurity study, the design of the laboratory in AWS with open-source tools and validation through simulated attacks.

This approach responds to the lack of affordable solutions, as many SMEs lack monitoring capabilities.

Resumen extendido

El trabajo desarrollado se centra en el diseño e implementación de un Centro de Operaciones de Seguridad adaptado a las necesidades de las pequeñas y medianas empresas y de los entornos SOHO. Estas organizaciones, a pesar de su relevancia económica y social, suelen carecer de los recursos técnicos y financieros necesarios para adoptar soluciones avanzadas de ciberseguridad como las que despliegan las grandes corporaciones. En consecuencia, la brecha de seguridad que se genera las convierte en objetivos atractivos para los ciberdelincuentes, que aprovechan sus limitaciones en monitorización, respuesta e incluso concienciación del personal. El estudio parte de este diagnóstico para proponer un modelo práctico, replicable y económico de SOC, basado en herramientas de software libre y en una infraestructura cloud escalable que permite acercar la seguridad avanzada a entornos con recursos limitados.

La investigación comienza con un repaso de los fundamentos de la ciberseguridad, sustentados en los principios de confidencialidad, integridad y disponibilidad, y con una descripción detallada de las amenazas que más afectan a las pequeñas empresas. Se presentan riesgos recurrentes que pueden comprometer la continuidad de un negocio como el phishing, los ransomware, los ataques de denegación de servicio, la explotación de vulnerabilidades en la cadena de suministro o los ataques de fuerza bruta. Estos incidentes no solo implican pérdidas económicas inmediatas, sino también consecuencias reputacionales y legales, dado que muchos implican la filtración de datos sensibles. Informes recientes señalan que hasta el 60% de las PYMEs que sufren un ciberataque grave cierran en un plazo inferior a seis meses, lo que evidencia que la ciberseguridad no puede considerarse opcional sino un elemento esencial para la supervivencia.

Frente a un panorama en el que las soluciones tradicionales basadas en antivirus o firewalls ya no resultan suficientes, el SOC es en cambio, una alternativa capaz de ofrecer un enfoque proactivo. Mientras las herramientas clásicas responden una vez que el ataque ya se ha materializado, el SOC permite la monitorización constante, la correlación de eventos y la orquestación de respuestas automatizadas que reducen de manera drástica los tiempos de detección y reacción.

En las grandes organizaciones, los SOC se implementan con tecnologías de pago, equipos de analistas especializados y presupuestos elevados, algo inalcanzable para una pequeña empresa. Por este motivo, la propuesta del

Este proyecto se centra en aprovechar soluciones open-source como Wazuh, TheHive, Cortex y MISP, desplegadas en la nube de AWS bajo un modelo de pago por uso, reduciendo costes y manteniendo al mismo tiempo las capacidades esenciales de monitorización, detección y respuesta.

La fase de investigación preliminar permitió identificar las tecnologías más adecuadas y definir los escenarios de ataque que se reproducirían en el laboratorio. Se valoraron alternativas de virtualización local, pero se optó por AWS para desplegar el SOC debido a sus ventajas en escalabilidad, seguridad y disponibilidad. El laboratorio se completó con máquinas virtuales locales en VMware que representaban al atacante y a la víctima, generando así un entorno híbrido capaz de reproducir situaciones realistas de ciberataques.

En el diseño del laboratorio, el SIEM se desplegó en una instancia EC2 t3.medium con Ubuntu, garantizando recursos suficientes para procesar y visualizar logs en tiempo real en Wazuh. Paralelamente, en otra instancia EC2 se configuró el ecosistema SOAR compuesto por TheHive, MISP y Cortex, encargados de la gestión de incidentes, del enriquecimiento de información mediante inteligencia de amenazas y la respuesta a las alertas. El atacante se implementó en Kali Linux, con acceso a herramientas como Hydra y Zphisher, mientras que la víctima se representó mediante una máquina Windows 10 y un servidor Windows Server 2019 configurado como controlador de dominio, simulando un entorno corporativo realista. La conectividad entre los distintos componentes se resolvió mediante integraciones específicas, de manera que las alertas de Wazuh se transformaban automáticamente en alertas en TheHive, con posibilidad de respuesta a través de Cortex y enriquecimiento con los indicadores de compromiso almacenados en MISP.

Los escenarios experimentales confirmaron la viabilidad del SOC diseñado. En el caso del ataque de fuerza bruta contra el puerto SSH, el atacante empleó Hydra para lanzar múltiples intentos de autenticación sobre la máquina Windows de la víctima. Los registros de los intentos fallidos fueron recogidos por el agente de Wazuh, procesados y correlacionados hasta generar una alerta que fue enviada a TheHive. Allí, la alerta se creó de forma automática y permitió ejecutar un responder desde Cortex para crear inmediatamente un ticket en Jira, demostrando así la capacidad de detección temprana del sistema.

En el escenario de phishing con robo de credenciales, el atacante utilizó Zphisher para clonar la página de inicio de sesión de Microsoft y alojarla en la red local, distribuyendo el enlace a través de un correo diseñado con

apariencia legítima. Cuando la víctima accedió e introdujo sus credenciales, la actividad fue registrada y generó una alerta crítica en Wazuh. La alerta llegó a TheHive, que abrió un caso y permitió lanzar un responder configurado en Cortex para notificar por correo electrónico al equipo de seguridad.

Los resultados experimentales de ambos escenarios validan que la arquitectura propuesta es capaz de detectar ataques en fases tempranas, clasificarlos bajo un marco de referencia reconocido y activar mecanismos de respuesta, a pesar de que en la versión *Community* la automatización de respuesta es limitada y la ejecución de *responders* tiene que ser de forma manual. La integración entre SIEM y SOAR se consolidó como el núcleo de la defensa, garantizando tanto la detección como la trazabilidad de la respuesta.

Las conclusiones del proyecto apuntan a que es posible ofrecer a las PYMEs un marco de referencia de bajo coste y alta efectividad para elevar su nivel de ciberseguridad. El laboratorio diseñado demuestra que mediante herramientas libres y una infraestructura cloud escalable, las empresas de menor tamaño pueden acceder a capacidades similares a las de un SOC corporativo sin necesidad de grandes inversiones. Asimismo, se resalta que la mayor parte de la inversión no radica en los recursos cloud, cuyo coste fue mínimo, sino en el tiempo y dedicación necesarios para el montaje, configuración y validación del entorno.

Finalmente, el trabajo plantea futuras líneas de evolución, como la automatización avanzada de respuestas mediante *webhooks*, la inclusión de escenarios más complejos como ransomware o ataques de denegación de servicio, y la aplicación de algoritmos de *machine learning* para mejorar la detección de anomalías y amenazas desconocidas. Estas líneas permitirían dotar al SOC de mayores capacidades y consolidar su utilidad como modelo replicable en la realidad empresarial.

En definitiva, el estudio no solo demuestra la viabilidad técnica del SOC en entornos reducidos, sino que también contribuye a cerrar la brecha de ciberseguridad que afecta a un sector especialmente vulnerable.

Índice general

1. INTRODUCCIÓN	12
1.1. PRESENTACIÓN	12
1.2. LA BRECHA DE CIBERSEGURIDAD EN LAS PYMEs Y SOHOs	13
1.3. AMENAZAS E IMPACTO	14
1.4. LA NECESIDAD DE UN ENFOQUE PROACTIVO	19
1.5. COMPOSICIÓN Y ARQUITECTURA TECNOLÓGICA DE UN SOC	19
1.6. COMPARATIVA SOC EMPRESARIAL VS UN SOC PARA PYMEs	21
2. INVESTIGACIÓN PRELIMINAR	23
2.1. TECNOLOGÍAS DE VIRTUALIZACIÓN Y CLOUD	23
2.2. IDENTIFICACIÓN DE VULNERABILIDADES Y ESCENARIO	23
3. DISEÑO DEL ENTORNO DE LABORATORIO	25
3.1. INTRODUCCIÓN	25
3.2. DESARROLLO DEL ENTORNO DE EXPERIMENTACIÓN	26
3.2.1. CREACIÓN Y CONFIGURACIÓN DE LA INFRAESTRUCTURA DE LA RED	26
3.2.2. CREACIÓN Y CONFIGURACIÓN DEL SIEM	29
3.2.3. CREACIÓN Y CONFIGURACIÓN DEL SOAR	33
3.2.4. CREACIÓN Y CONFIGURACIÓN DE LA VÍCTIMA	42
3.2.5. CREACIÓN Y CONFIGURACIÓN DEL ATACANTE	52
3.3. CONECTIVIDAD E INTERRELACIÓN ENTRE COMPONENTES	53
3.3.1. CONEXIÓN DE THEHIVE CON MISP	53
3.3.2. CONEXIÓN DE THEHIVE CON CORTEX	56
3.3.3. CONEXIÓN DE MISP CON CORTEX	59
3.3.4. CONEXIÓN DE THEHIVE CON WAZUH	60
3.3.5. CONEXIÓN DE WAZUH CON LA ESTACIÓN DE TRABAJO	62
4. RESULTADOS	63
4.1. ESCENARIO 1. ATAQUE POR FUERZA BRUTA AL PUERTO SSH	63
4.1.1. ROLES Y PARTICIPANTES:	63
4.1.2. OBJETIVOS Y ALCANCE	63
4.1.3. METODOLOGÍA DE ATAQUE (MITRE ATT&CK)	64
4.1.4. PROCEDIMIENTO PASO A PASO	65
4.1.5. RESULTADOS EXPERIMENTALES	71
4.2. ESCENARIO 2. PHISHING CON ROBO DE CREDENCIALES	72
4.2.1. ROLES Y PARTICIPANTES:	72
4.2.2. OBJETIVOS Y ALCANCE	73
4.2.3. METODOLOGÍA DE ATAQUE (MITRE ATT&CK)	74
4.2.4. PROCEDIMIENTO PASO A PASO	74
4.2.5. RESULTADOS EXPERIMENTALES	82
5. CONCLUSIONES Y LÍNEAS FUTURAS	84
5.1. CONCLUSIONES	84
5.2. LÍNEAS FUTURAS	85
BIBLIOGRAFÍA	86
ANEXOS	88
PRESUPUESTO	88
CÓDIGO FUENTE	91
RESPONDER ATAQUE FUERZA BRUTA	91
RESPONDER PHISHING	94

Lista de acrónimos

AWS	Amazon Web Services.
COBIT	Control Objectives for Information and Related Technology.
EC2	Amazon Elastic Compute Cloud.
ESC	Escenario.
DoS	Denial-of-service.
DDoS	Distributed Denial-of-service.
GDPR	General Data Protection Regulation.
INCIBE	Instituto Nacional de Ciberseguridad.
IT	Information Technology.
NAT	Network Address Translation.
MFA	Multi-Factor Authentication.
ML	Machine Learning.
PYMEs	Pequeñas y medianas empresas.
RAM	Random-access memory.
SIEM	Security Information and Event Management.
SMEs	Small and medium enterprises.
SOAR	Security Orchestration, Automation and Response.
SOHOs	Small Office/Home Office.
SOC	Security Operations Center.
VM	Virtual Machine.
VPC	Virtual Private Cloud.
VPN	Virtual Private Network.

Capítulo 1

1. Introducción

1.1. Presentación

La ciberseguridad se define como «La práctica de defender ordenadores, servidores, dispositivos móviles, sistemas electrónicos, redes y datos de ataques malintencionados.» [1].

En definitiva, es la protección de los activos de la información. El enfoque de este tipo de seguridad se sustenta en tres principios fundamentales conocidos como el triángulo CIA: confidencialidad, integridad y disponibilidad.

El primero de ellos, confidencialidad, se enfoca en garantizar la privacidad de los datos permitiendo que solo usuarios autorizados accedan a los datos. Para lograr este objetivo, es imprescindible implementar mecanismos de control que prevengan accesos no autorizados y, al mismo tiempo, aseguren la integridad de los datos. En el contexto empresarial, este principio exige adoptar políticas de gestión de accesos, donde los privilegios de cada usuario estén alineados con las necesidades de su rol. Además, técnicas como el cifrado de datos y protocolos de autenticación como los MFA, refuerzan este objetivo.

Alineado con lo anterior, mientras que la confidencialidad garantiza el control de quién accede a los datos, la integridad asegura que dicha información conserve su autenticidad, precisión y confiabilidad.

Sin embargo, si la disponibilidad a estos datos no se cumple, los otros dos fundamentos carecen de valor. Este pilar exige que los sistemas, redes y aplicaciones mantengan un funcionamiento óptimo y continuo, permitiendo tanto a clientes como al personal autorizado acceder a la información oportuna sin retardos que comprometan la calidad del servicio. Este principio puede verse comprometido por causas intencionales como son los ataques de denegación de servicio (DoS), pero también por causas no maliciosas como desastres naturales que puedan comprometer la disponibilidad de la información si no hay un sistema de recuperación ante estos desastres.

1.2. La brecha de ciberseguridad en las PYMEs y SOHOs

La sociedad se encuentra en una era cada vez más digital y donde la dependencia tecnológica de las organizaciones crece día a día, y con ello los ataques cibernéticos evolucionan también de manera exponencial, por esto, la ciberseguridad ha adquirido papel fundamental.

Teniendo en cuenta este contexto social, las grandes empresas han desarrollado infraestructuras de seguridad que incluyen, entre otros, Centros de Operaciones de Seguridad para la monitorización, detección y respuesta ante incidentes en tiempo real. De esta forma, pueden enfrentarse a las amenazas cibernéticas de manera más eficiente y temprana. Estas infraestructuras representan una inversión significativa, lo que deja a las pequeñas y medianas empresas (PYMEs) y a las oficinas de pequeño tamaño (SOHOs) en desventaja, ya que a menudo carecen de los recursos financieros y del personal especializado necesario para implementar este tipo de soluciones. Muchas de estas empresas carecen incluso de personal especializado en IT, por lo que delegan la gestión de la seguridad a empleados con conocimiento básicos o a soluciones genéricas que no cubren las amenazas actuales.

Esta brecha se agrava por la falsa percepción de que las empresas pequeñas no son objetivos valiosos para los cibercriminales. Sin embargo, los datos muestran lo contrario ya que las pequeñas y medianas empresas están enfrentando un aumento alarmante en los ciberataques, según un estudio reciente [2]. El número de ataques a este tipo de empresas representa más del 40% y esto se debe precisamente a que representan un blanco atractivo para los ciberatacantes debido a sus limitaciones en ciberdefensa [3]. Además, la creciente digitalización de los procesos empresariales, desde el comercio electrónico hasta el teletrabajo, ha expandido la superficie de ataque sin que se hayan reforzado de forma paralela las medidas de protección. El resultado es un escenario donde muchas empresas operan con sistemas obsoletos, copias de seguridad insuficientes y sin protocolos de respuesta ante incidentes.

Otro factor clave es la falta de concienciación sobre los riesgos reales. Mientras que las grandes organizaciones invierten una gran parte del presupuesto anual en formación y simulacros de seguridad, en las PYMEs estos aspectos suelen descuidarse hasta que ocurre un ataque. El desconocimiento sobre prácticas básicas, las autenticaciones multifactor (MFA) o la segmentación de redes, aumenta el riesgo. Por ello, la formación del personal es imprescindible ya que, ataques como el phishing son una de las amenazas constantes en el sector.

Esta combinación de factores crea una brecha de seguridad que no solo pone en peligro a las empresas individuales, sino también a toda su cadena de valor, incluyendo a clientes y proveedores.

Cerrar esta brecha requiere de soluciones accesibles y adaptadas a la realidad de las PYMEs y SOHOs como un SOC simplificado y de software libre que les permita acceder a sistemas de seguridad robustos sin incurrir en altos costes económicos o requerir una infraestructura tecnológica sofisticada.

1.3. Amenazas e impacto

Todas las organizaciones, independientemente de su dimensión se encuentran expuestas a un amplio espectro de amenazas ciberneticas. Sin embargo, como se ha comentado anteriormente, las pequeñas empresas y SOHOs enfrentan un panorama de ciberamenazas cada vez más agresivo.

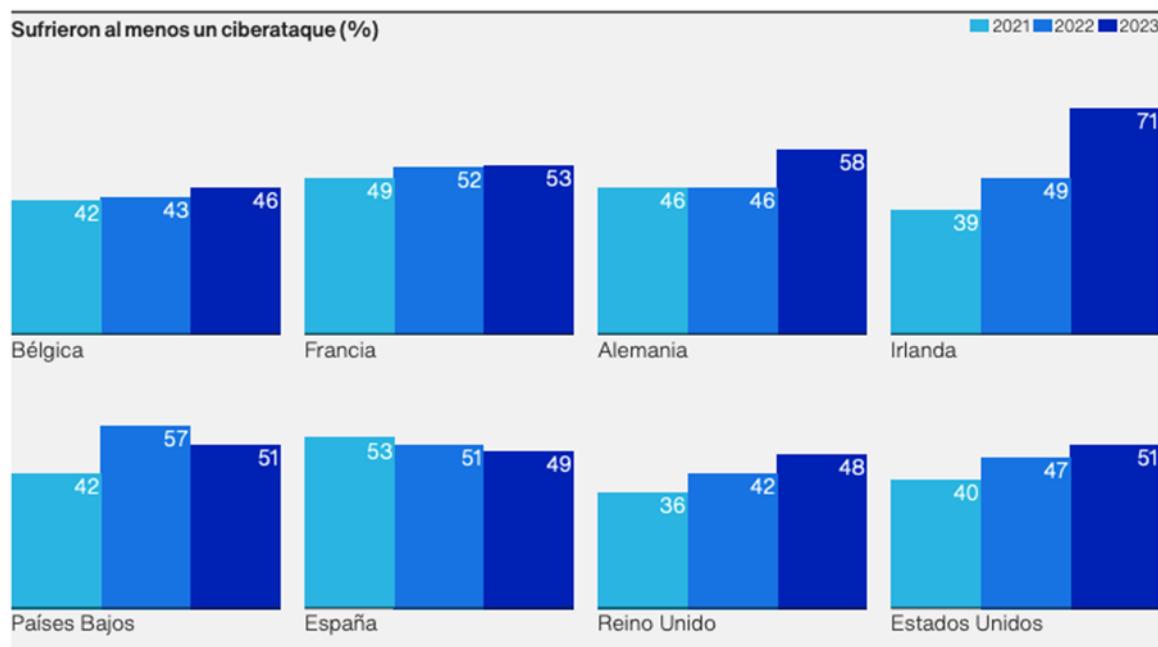


Figura 1. Porcentaje de PYMEs que sufrieron al menos un ciberataque (2021–2023).

Nota. Tomado de *Informe de Ciberpreparación 2023* (p. 18), por Hiscox, 2023.
<https://www.hiscox.es/...>

Entre las amenazas más destacadas a este tipo de empresas se encuentran [4]:

- **Phishing.** Ciberataques basados en técnicas de ingeniería social cuyo objetivo es engañar a los usuarios para obtener credenciales de acceso, datos personales o información sensible. Estas campañas han evolucionado desde los correos genéricos hasta sofisticadas

operaciones de spear-phishing que imitan comunicaciones legítimas de proveedores, clientes o entidades bancarias. Esta modalidad de fraude digital representa una de las amenazas más extendidas debido a su bajo coste y elevada efectividad. Según el último informe del INCIBE, el 78% de las PYMEs españolas ha sufrido al menos un intento de phishing en el último año [5]. La falta de concienciación del personal, unida a la ausencia de filtros avanzados de correo, multiplica el riesgo en estos entornos.

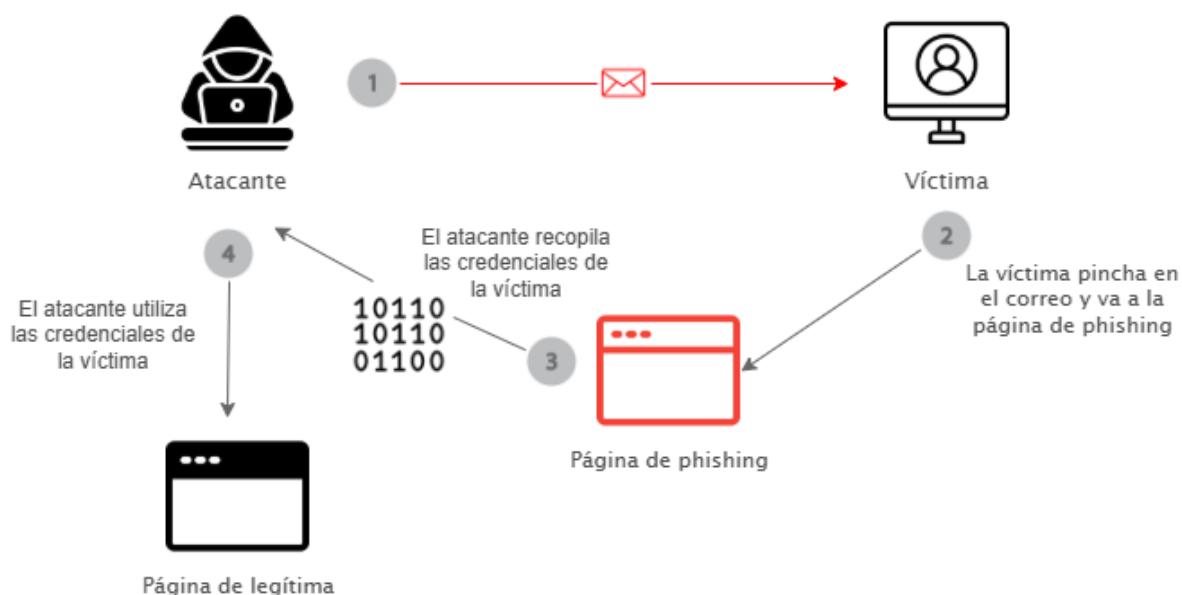


Figura 2. Flujo de ataque Phishing.

- **Malware.** Softwares diseñados con el propósito de comprometer sistemas, es decir, de dañar o alterar un sistema informático. Entre sus variantes más comunes se incluyen virus, códigos maliciosos y ransomware. Este último caracterizado por el cifrado de archivos, lo que impide el acceso a los mismos y exige el pago de un rescate a los atacantes para su recuperación. En los últimos años, ha surgido como una de las amenazas ciberneticas más relevantes y perjudiciales para las organizaciones debido a su creciente sofisticación y al impacto que genera. Según el informe State of Ransomware 2024 de Sophos, más de la mitad de las víctimas (56%) que sufrieron cifrado de datos pagaron el rescate para intentar recuperarlos [6]. Además, los tiempos de recuperación pueden extenderse durante semanas, lo que representa un evento crítico para muchas PYMEs.

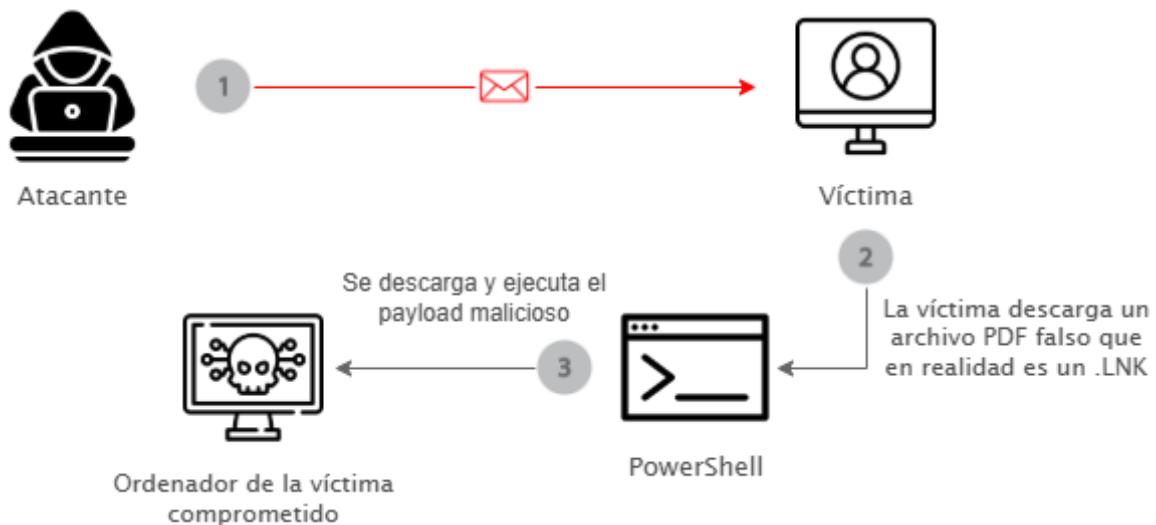


Figura 3. Flujo de ataque Malware.

- **Ataques de denegación de servicio (DoS).** Ofensiva cibernética dirigida a interrumpir la disponibilidad del sistema sobrecargándolo con un volumen masivo de solicitudes o tráfico malicioso. Estos ataques se manifiestan mediante el agotamiento de recursos como ancho de banda o capacidad de procesamiento impidiendo el acceso legítimo a los servicios afectados. Su impacto es devastador en muchas ocasiones ya que, en sectores como el comercio electrónico o el financiero, la disponibilidad continua de sus servicios digitales es esencial. Estos ataques, que pueden alquilarse por una cantidad ínfima de dinero, generan pérdidas que oscilan entre 5.000€ y 20.000€ por hora de inactividad en la empresa víctima [7].

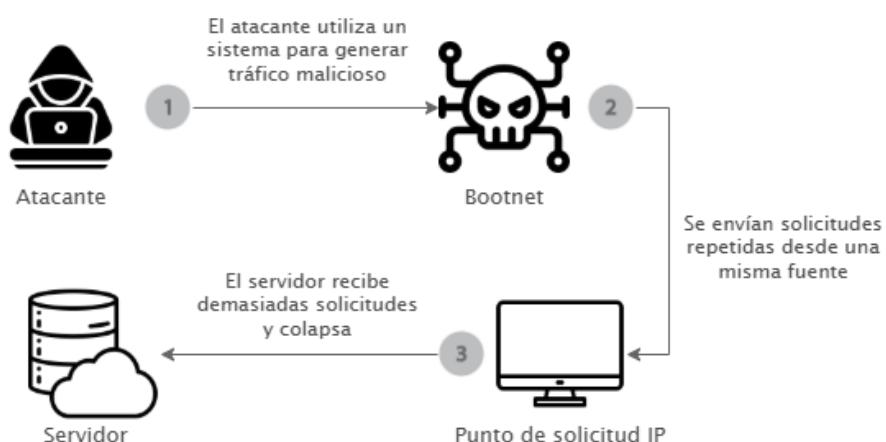


Figura 4. Flujo de ataque DoS.

- **Ataque a la cadena de suministro.** Tácticas maliciosas que comprometen la seguridad de una organización a través de

vulnerabilidades en sus proveedores o socios comerciales. Se basan en la infiltración indirecta en los sistemas objetivos a través de las relaciones de confianza establecidas entre entidades interconectadas. Las consecuencias de estos incidentes pueden materializarse, por ejemplo, en el acceso no autorizado a información sensible.



Figura 5. Flujo de ataque cadena de suministro.

- **Ataques de fuerza bruta.** Técnica de ciberseguridad que utiliza la prueba de múltiples combinaciones de contraseñas para acceder a sistemas o cuentas. La persistencia de contraseñas débiles y la falta de autenticación multifactor, convierten a muchos sistemas de PYMEs en objetivos fáciles. Un gran porcentaje de estos ataques exitosos podrían prevenirse con medidas básicas como MFA o políticas de contraseñas robustas.



Figura 6. Flujo de ataque de fuerza bruta.

El impacto de estos incidentes en las pequeñas empresas puede ser devastador, tanto desde el punto de vista económico como operacional. El 60% de empresas de este sector afectadas por un ataque cibernético se ven obligadas a cerrar en un plazo de seis meses, debido a su limitada capacidad para asumir las repercusiones financieras y técnicas derivadas del incidente [3].

La interrupción de servicios clave, por ejemplo, debido a un ransomware o

una filtración de datos, genera pérdidas económicas inmediatas. Además de los costos adicionales asociados al rescate y/o restauración de sistemas que pueden superar con creces el capital de la empresa llevándole al cierre definitivo.

Desde el punto de vista operacional, un incidente de seguridad no solo afecta a la infraestructura tecnológica sino también a la confianza de clientes y socios comerciales. Una brecha de datos y especialmente si se trata de información sensible, además de suponer consecuencias legales por el incumplimiento de normativas como el GDPR, supone también un daño reputacional difícil de reparar. A esto se le suma la falta de planes de recuperación ante desastres en la mayoría de PYMEs, lo que prolonga los tiempos de inactividad y agrava las pérdidas.

Estos factores demuestran que la ciberseguridad no es un gasto opcional, sino una necesidad crítica para la supervivencia de este tipo de empresas. Implementar medidas proactivas como un SOC adaptado a sus capacidades no solo reduciría el riesgo de ataques, sino que también mitigaría el impacto de estos en caso de que ocurrieran asegurando así la continuidad del negocio.

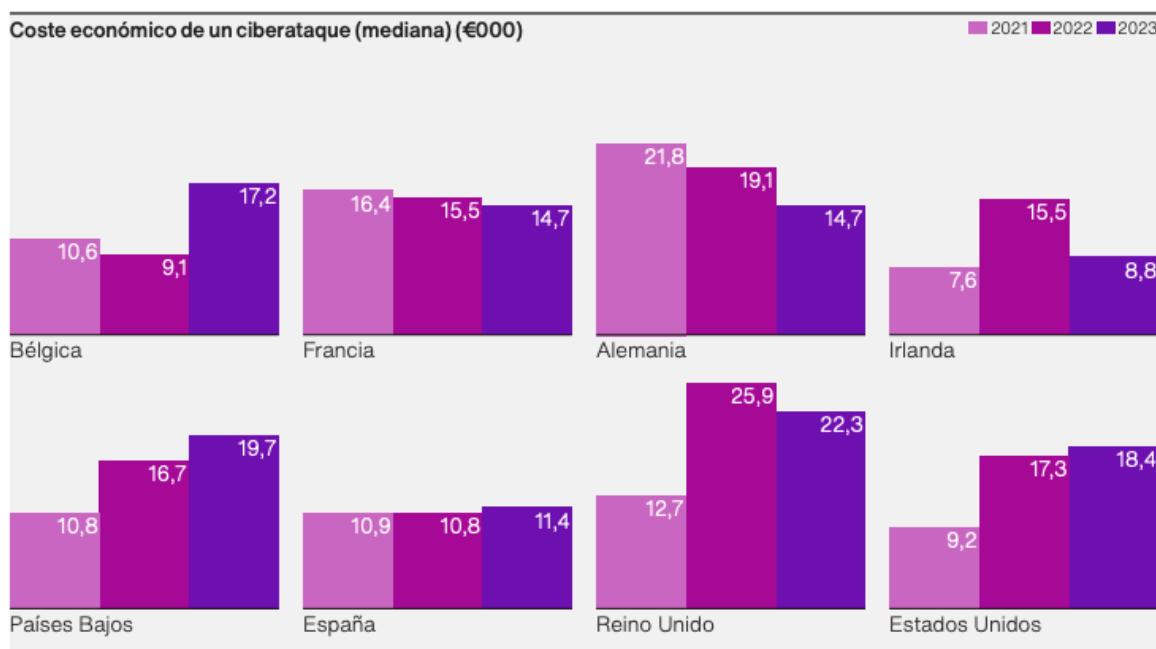


Figura 7. Coste económico de un ciberataque en medianas empresas (2021–2023 en miles de €).

Nota. Tomado de Informe de Ciberpreparación 2023 (p. 18), por Hiscox, 2023.
<https://www.hiscox.es/...>

1.4. La necesidad de un enfoque proactivo

En el contexto actual de ciberamenazas en constante evolución, las medidas de seguridad tradicionales como antivirus y firewalls son claramente insuficientes y muestran graves limitaciones. Estos sistemas tienen un enfoque reactivo, esperan a que ocurra un ataque para intentar mitigarlo, lo que, para las amenazas actuales como los ransomware, los ataques *zero day* o el phishing dirigido son insuficientes ya que, una vez estas herramientas los detectan, el daño suele estar hecho. Este modelo obsoleto deja a las pequeñas empresas con un tiempo de respuesta crítico para su supervivencia.

Frente a esta problemática, un Centro de Operaciones de Seguridad (SOC) cambia radicalmente el enfoque. Se trata de un sistema integral que combina tecnologías avanzadas, procesos estandarizados y personal especializado que opera como el núcleo centralizado de defensa de una organización.

Lo que diferencia al SOC de las herramientas tradicionales es su naturaleza preventiva y su capacidad de análisis contextual. Gracias a la monitorización constante de la infraestructura digital, es capaz de identificar patrones de comportamientos sospechosos. Además, tiene la capacidad de correlacionar eventos aparentemente inconexos, distinguir entre actividades legítimas y potencialmente dañinas, y detectar ataques en fases iniciales, cuando aún pueden ser contenidos. Este cambio implica una transición hacia un nuevo escenario centrado en la seguridad proactiva.

La implementación de un SOC trae consigo transformaciones profundas en la postura de seguridad de una PYME. Al pasar de un modelo reactivo a uno proactivo, las organizaciones ganan capacidad de anticipación. La respuesta a incidentes se acelera pudiendo, por ejemplo, aislar sistemas comprometidos en minutos. Por otro lado, cada incidente documentado por el SOC refina los mecanismos de detección mejorando la defensa.

1.5. Composición y arquitectura tecnológica de un SOC

Un SOC integra múltiples capas tecnológicas que trabajan de forma coordinada para ofrecer protección integral. No existe un estándar determinado, cada organización lo implementa según sus necesidades y recursos.

La arquitectura básica comienza con los componentes de red como routers y firewalls que actúan como primera línea de defensa. Estos dispositivos no solo gestionan el tráfico, sino que incorporan funciones avanzadas como la

inspección de paquetes (DPI) y la prevención de intrusiones (IPS). Además, se configuran para el envío de logs detallados a los sistemas centrales de análisis, permitiendo correlacionar eventos de red con otras alertas de seguridad.

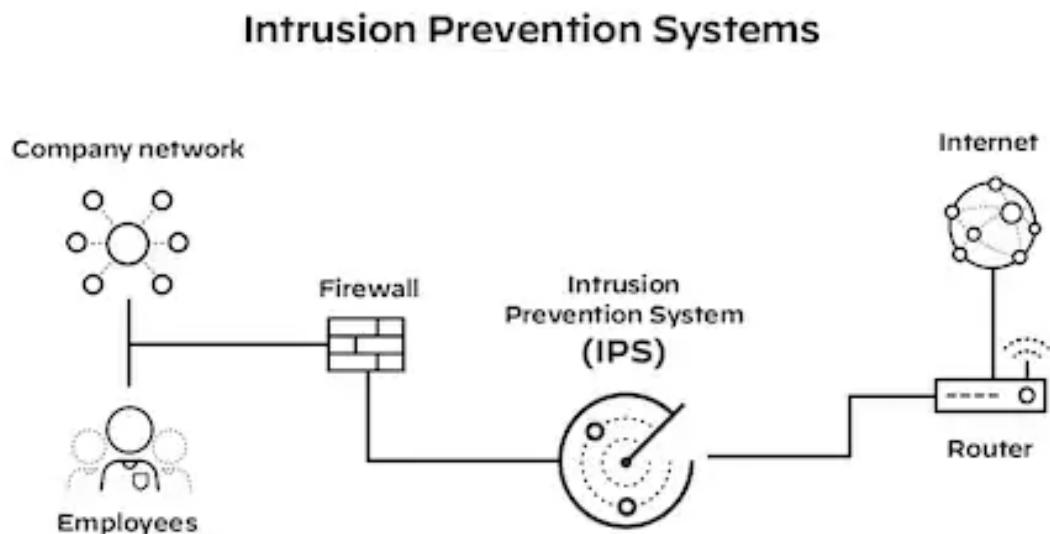


Figura 8. Esquema de red con IPS.

Nota. Tomado de What is an Intrusion Prevention System? por Paloalto Network.
<https://www.paloaltonetworks.com/...>

El núcleo analítico del SOC reside en las plataformas SIEM (Security Information and Event Management). Los SIEM se encargan de recopilar, normalizar y correlacionar eventos de seguridad de todos los dispositivos de red, servidores y aplicaciones a tiempo real. Su valor principal radica en identificar patrones sospechosos. La ventaja clave de utilizar este tipo de plataformas es la capacidad de contextualizar alertas aparentemente inconexas que podrían indicar un ataque en progreso. Sistemas como Splunk o Elastic Security utilizan técnicas de machine learning y técnicas de análisis de comportamiento para establecer líneas base de comportamiento normal y detectar anomalías. Sin embargo, existe una alternativa open-source, Wazuh, que se centra en una detección basada en reglas y análisis de logs y es una solución ideal para PYMEs y SOHOs.

La automatización de respuestas se gestiona mediante plataformas SOAR (Security Orchestration, Automation and Response). Estas herramientas permiten automatizar el flujo de trabajo de respuesta a incidentes. Cuando el SIEM detecta una amenaza, el SOAR ejecuta automáticamente protocolos de contención como aislar endpoints afectados, resetear credenciales

comprometidas o incluso bloquear direcciones IP maliciosas en el firewall con la mínima intervención humana. Este enfoque reduce los tiempos de respuesta ante ataques.

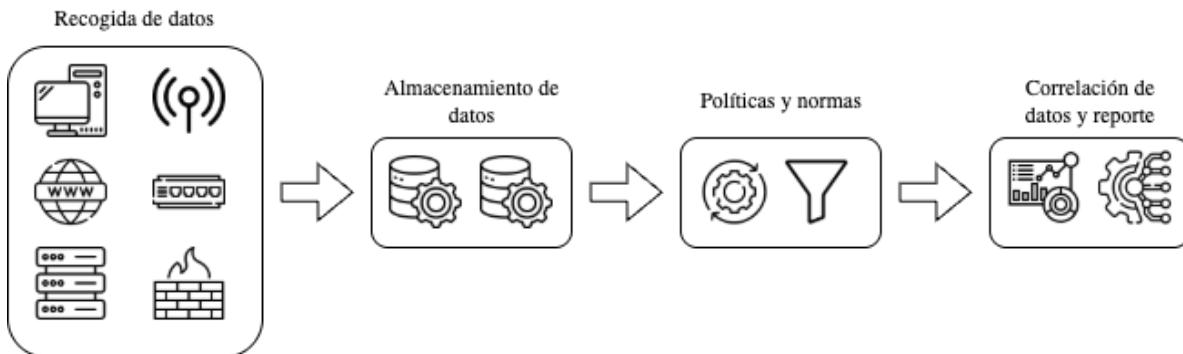


Figura 9. Flujo de trabajo del SIEM.

1.6. Comparativa SOC empresarial vs un SOC para PYMEs

En el ámbito corporativo, los componentes del SOC suelen implementarse a máxima escala con herramientas empresariales, centros físicos y equipos especializados inalcanzables para la mayoría de las pequeñas y medianas empresas.

Un SOC empresarial típico puede invertir cientos de miles de euros en licencias de software. Su arquitectura suele desplegarse en DataCenters privados, integrando herramientas comerciales para cada función específica: firewalls de última generación como Palo Alto Networks, SIEMs con capacidades avanzadas de machine learning como Splunk, y plataformas SOAR completamente personalizadas como IBM Resilient. Además, su complejidad operativa requiere múltiples niveles de analistas y procesos bien documentados según el marco COBIT.

Frente a este modelo, la propuesta que se plantea en este proyecto aprovecha las ventajas de la nube y el software open-source para ofrecer las capacidades de un SOC esenciales reduciendo costes para que pequeñas y medianas empresas puedan acceder a él. La solución se construye sobre una infraestructura cloud en AWS, eliminando la necesidad de hardware dedicado y aprovechando la escalabilidad bajo demanda. El SIEM central se puede implementar con Wazuh, procesando logs y generando alertas mediante reglas. La automatización SOAR se simplifica usando herramientas como TheHive, MISP y Cortex permitiendo contener amenazas comunes sin intervención manual y analizando las alertas. A diferencia del SOC corporativo, donde cada herramienta es un producto comercial

independiente, esta arquitectura se basa en la interoperabilidad de soluciones open-source reduciendo costes, pero requiriendo mayor esfuerzo inicial de configuración.

La diferencia fundamental radica en el equilibrio entre capacidades y recursos. Mientras un SOC empresarial busca cobertura total contra cualquier posible amenaza, la versión para PYMEs se optimiza para las vulnerabilidades más probables según su perfil de riesgo. Donde una corporación analiza el 100% de los eventos con equipos dedicados, la solución cloud prioriza alertas mediante correlación básica y la monitorización 24/7 se logra mediante reglas de notificaciones en vez de con turnos de personal.

Por tanto, al combinar plataformas open-source, infraestructura cloud pagada por uso, y automatización selectiva, las PYMEs pueden acceder a un 80% de las capacidades SOC críticas con menos del 20% de la inversión requerida por grandes empresas, demostrando así, que es posible implementar un SOC funcional con un presupuesto ajustado, sacrificando algunas capacidades avanzadas, pero cubriendo los vectores de ataque más relevantes para organizaciones de menor tamaño.

Categoría	SOC Empresarial	SOC para PYMEs
Infraestructura	Datacenters privados con redundancia total	Infraestructura cloud (AWS), sin hardware físico
Plataforma SIEM	SIEMs premium con ML (Splunk)	SIEM open-source (Wazuh)
Automatización (SOAR)	SOAR personalizado (IBM Resilient)	SOAR simplificado con TheHive y MISP
Coste aproximado	Cientos de miles de euros en licencias	Menos del 20% del coste empresarial
Personal y operaciones	Múltiples niveles de analistas; procesos COBIT	Automatización con reglas y notificaciones móviles
Escalabilidad	Alta, con recursos dedicados	Escalabilidad bajo demanda en la nube
Cobertura de amenazas	Cobertura completa de amenazas	Foco en amenazas más probables
Modelo de implementación	On-premise, altamente personalizado	Cloud, pagado por uso
Tecnología base	Software comercial propietario	Software open-source interoperable

Figura 10. Comparativa SOC empresarial vs SOC para PYMEs.

Capítulo 2

2. Investigación preliminar

2.1. Tecnologías de virtualización y Cloud

La virtualización tiene un papel clave en la implementación de SOCs ya que proporciona la flexibilidad y eficiencia para la creación de entornos aislados y seguros donde se puedan desplegar y probar herramientas de seguridad sin la necesidad de una infraestructura física. Gracias a esta tecnología, es posible simular redes completas, replicar amenazas y segmentar funciones críticas. Además, permite aislar procesos en máquinas virtuales independientes, lo que facilita la contención de amenazas y reduce el riesgo de un fallo que comprometa a todo el sistema. Por otro lado, la capacidad de simular incidentes en un entorno controlado es especialmente importante de cara a validar procedimientos de detección y respuesta en un ataque real. También, la virtualización proporciona escalabilidad y adaptabilidad, ya que los recursos pueden ajustarse según la carga de trabajo, algo esencial en un entorno de defensa dinámica donde las amenazas cambian continuamente.

Durante la fase de investigación de este proyecto se valoraron soluciones de virtualización local como VMware o VirtualBox para la implementación del SOC. Sin embargo, se optó por trabajar con una infraestructura cloud utilizando AWS ya que permite desplegar servicios de manera ágil, segura y escalable. Además, garantiza una alta disponibilidad y permite un modelo de pago por uso que resulta adecuado para proyectos con recursos limitados como los de las PYMES o este mismo. La infraestructura cloud también facilita la automatización de despliegues, la aplicación de políticas de seguridad centralizadas y el acceso remoto al entorno desde cualquier lugar lo que amplía las posibilidades de monitorización y respuesta.

En definitiva, la decisión de utilizar AWS como base del laboratorio responde tanto a criterios técnicos como económicos, y refleja a su vez una solución realista y replicable para las pequeñas organizaciones.

2.2. Identificación de vulnerabilidades y escenario

Tal como se ha expuesto en el apartado 1.3, el ecosistema de amenazas incluye una variedad de ataques recurrentes como phishing, ransomware,

denegación de servicio (DoS), ataques a la cadena de suministro y técnicas de fuerza bruta.

La ausencia de mecanismos robustos de autenticación como los multifactor (MFA), facilita accesos no autorizados a través de credenciales filtradas o robadas. A esto se suma la existencia de contraseñas débiles o reutilizadas, que pueden ser fácilmente explotadas mediante ataques de fuerza bruta automatizados. Otro punto débil recurrente es la falta de actualización de sistemas y software, lo que deja expuestas vulnerabilidades conocidas que pueden ser aprovechadas mediante exploits disponibles públicamente. Además, es frecuente la exposición innecesaria de puertos y servicios a internet, sin mecanismos de filtrado ni supervisión. También, la carencia de segmentación de red y la mala gestión de permisos internos agravan aún más la situación, ya que permiten que un atacante que compromete un único equipo pueda escalar privilegios y propagarse lateralmente.

En base a estas observaciones, se han definido un par de escenarios de prueba representativos para validar la capacidad del SOC implementado en la detección y gestión de incidentes:

ESC-1. Ataque de fuerza bruta sobre servidor SSH

Se plantea un ataque de fuerza bruta sobre un servidor SSH, simulando múltiples intentos de acceso con credenciales incorrectas, para comprobar si el sistema activa alertas por actividad anómala.

ESC-2. Simulación de phishing con credenciales comprometidas

Se emula un ataque de phishing mediante un correo, con el objetivo de capturar credenciales de acceso, para evaluar si el SOC detecta el intento de acceso posterior con credenciales robadas.

Capítulo 3

3. Diseño del entorno de laboratorio

3.1. Introducción

El entorno del laboratorio se ha estructurado en dos componentes principales. Por un lado, el SOC, cuya implementación se ha realizado en la plataforma Amazon Web Services (AWS), utilizando una cuenta del tipo Free Tier, tal como se ha mencionado en capítulos anteriores. Por otro lado, se han desplegado las máquinas correspondientes al atacante y a la víctima en un entorno de virtualización local mediante VMware.

La decisión de utilizar AWS para el despliegue del SOC responde a la necesidad de contar con una infraestructura flexible, accesible y escalable, sin incurrir en costes elevados. Esta plataforma permite implementar una arquitectura funcional que se adapta a los requerimientos del laboratorio.

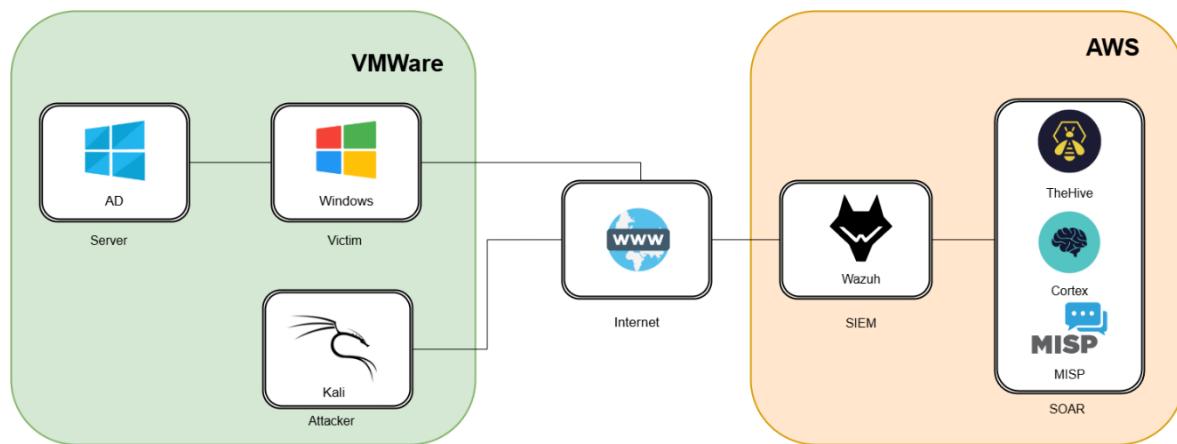


Figura 11. Estructura del laboratorio.

Como primera línea de defensa al SOC, se implementan los Security Groups y las listas de control de acceso a la red propios de AWS. Los Security Groups actúan como firewalls definiendo reglas de entrada y salida para cada instancia.

El núcleo del SOC está conformado por un SIEM, específicamente Wazuh, desplegado sobre una instancia EC2 con Ubuntu. Este componente se encarga de la recolección y análisis de logs provenientes de la víctima, permitiendo detectar comportamientos anómalos y generando alertas en tiempo real.

Para completar la capacidad de detección del SIEM, se integra un sistema SOAR compuesto por TheHive, MISP y Cortex desplegados en una misma instancia EC2. TheHive permite gestionar los incidentes detectados, documentar evidencias y coordinar acciones de respuesta. En cambio, MISP correlaciona los eventos registrados en el entorno con los indicadores de compromiso. Y Cortex se encarga de la respuesta al incidente. El SIEM y el SOAR trabajan en conjunto para ofrecer un ciclo completo de detección, análisis y respuesta ante incidentes.

El atacante se ha implementado en una máquina virtual Kali Linux, desplegada en VMWare. Esta máquina tiene como objetivo llevar a cabo ataques dirigidos contra los empleados simulados, representados por una máquina con Windows 10, también montada en VMWare. Esta última genera logs de actividad legítima y anómala, aportando contexto realista a los escenarios de simulación y análisis.

3.2. Desarrollo del entorno de experimentación

3.2.1. Creación y configuración de la infraestructura de la red

Como base de la infraestructura, se ha creado una Virtual Private Cloud (VPC), que permite el aislamiento lógico de los recursos desplegados y proporcionar un entorno seguro, controlado y adaptado a las necesidades del laboratorio. La definición de un espacio de direcciones IP propio permite organizar y segmentar los servicios de forma eficiente, así como aplicar políticas específicas de control de tráfico y seguridad.

La VPC ha sido configurada con un rango de direcciones IP privadas amplio (10.0.0.0/16), lo que permite la escalabilidad del entorno. Dentro de esta red se han establecido una subred pública.

Para habilitar el acceso desde la subred pública, se ha asociado una Gateway de Internet y se ha definido una tabla de rutas que permite la salida hacia el exterior.

Para permitir la administración de las instancias que requieren ser accesibles para tareas de configuración y mantenimiento, como la instalación de herramientas especializadas como TheHive y MISP se ha habilitado el puerto SSH.

A continuación, se muestran capturas del proceso de creación de la infraestructura:

VPC > Sus VPC > Crear VPC

Crear VPC Información

Una VPC es una parte aislada de la nube de AWS que contiene objetos de AWS, como instancias de Amazon EC2.

Configuración de la VPC

Recursos que se van a crear Información
Cree únicamente el recurso de VPC o la VPC y otros recursos de red.

Solo la VPC VPC y más

Etiqueta de nombre - opcional
Crea una etiqueta con una clave de "Nombre" y el valor que usted especifique.

MyVPC

Bloque de CIDR IPv4 Información
 Entrada manual de CIDR IPv4
 Bloque de CIDR IPv4 asignado por IPAM

CIDR IPv4
10.0.0.0/16
El tamaño del bloque CIDR debe estar entre /16 y /28.

Bloque de CIDR IPv6 Información
 Sin bloque de CIDR IPv6
 Bloque de CIDR IPv6 asignado por IPAM
 Bloque de CIDR IPv6 proporcionado por Amazon
 CIDR IPv6 de mi propiedad

Tenencia Información
Predeterminado

Figura 12. Creación de la VPC.

VPC > Subredes > Crear subred

Crear subred Información

VPC

ID de la VPC
Cree subredes en esta VPC.
vpc-08dcab646683f5837 (MyVPC)

CIDR de VPC asociados

CIDR IPv4
10.0.0.0/16

Configuración de la subred

Especifique los bloques de CIDR y la zona de disponibilidad de la subred.

Subred 1 de 1

Nombre de la subred
Cree una etiqueta con una clave de "Nombre" y el valor que especifique.
PublicSubnet

El nombre puede tener un máximo de 256 caracteres.

Zona de disponibilidad Información
Elija la zona en la que residirá la subred o deje que Amazon elija una por usted.
Europa (Irlanda) / eu-west-1a

Bloque de CIDR de VPC IPv4 Información
Elija el bloque CIDR IPv4 de la VPC para la subred. El CIDR IPv4 de la subred debe estar dentro de este bloque.
10.0.0.0/16

Bloque de CIDR de la subred IPv4
10.0.1.0/24 256 IPs

Figura 13. Creación subred pública.

VPC > Tablas de enruteamiento > Crear tabla de enruteamiento

Crear tabla de enruteamiento Información

Una tabla de enruteamiento especifica cómo se envían los paquetes entre las subredes de la VPC, Internet y la conexión de la VPN.

Configuración de la tabla de enruteamiento

Nombre - optional
Cree una etiqueta con una clave de "Nombre" y el valor que especifique.

VPC
La VPC que se debe usar para esta tabla de enruteamiento.

vpc-08dcab646683f5837 (MyVPC)

Figura 14. Creación tabla de enruteamiento.

VPC > Tablas de enruteamiento > rtb-005b5063978c8c952 > Editar asociaciones de subredes

Editar asociaciones de subredes

Cambiar las subredes que están asociadas a esta tabla de enruteamiento.

Subredes disponibles (1/2)																				
<input type="checkbox"/> Filtrar asociaciones de subredes <table border="1"> <thead> <tr> <th>Nombre</th> <th>ID de subred</th> <th>CIDR IPv4</th> <th>CIDR IPv6</th> <th>ID de tabla de enruteamiento</th> </tr> </thead> <tbody> <tr> <td>PrivateSubnet</td> <td>subnet-0c3aedcbba430994e</td> <td>10.0.2.0/24</td> <td>-</td> <td>Principal (rtb-0c5fc58997b19a05)</td> </tr> <tr> <td><input checked="" type="checkbox"/> PublicSubnet</td> <td>subnet-05a1022a28e988340</td> <td>10.0.1.0/24</td> <td>-</td> <td>rtb-005b5063978c8c952 / PublicRout...</td> </tr> </tbody> </table>						Nombre	ID de subred	CIDR IPv4	CIDR IPv6	ID de tabla de enruteamiento	PrivateSubnet	subnet-0c3aedcbba430994e	10.0.2.0/24	-	Principal (rtb-0c5fc58997b19a05)	<input checked="" type="checkbox"/> PublicSubnet	subnet-05a1022a28e988340	10.0.1.0/24	-	rtb-005b5063978c8c952 / PublicRout...
Nombre	ID de subred	CIDR IPv4	CIDR IPv6	ID de tabla de enruteamiento																
PrivateSubnet	subnet-0c3aedcbba430994e	10.0.2.0/24	-	Principal (rtb-0c5fc58997b19a05)																
<input checked="" type="checkbox"/> PublicSubnet	subnet-05a1022a28e988340	10.0.1.0/24	-	rtb-005b5063978c8c952 / PublicRout...																
Subredes seleccionadas <input type="checkbox"/> subnet-05a1022a28e988340 / PublicSubnet <input type="button" value="X"/>																				

Figura 15. Asociación de la tabla de enruteamiento a la subred pública.

VPC > Internet gateways > Create internet gateway

Create internet gateway Info

An internet gateway is a virtual router that connects a VPC to the internet. To create a new internet gateway specify the name for the gateway below.

Internet gateway settings

Name tag
Creates a tag with a key of 'Name' and a value that you specify.

Tags - optional
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

Key	Value - optional
<input type="text" value="Name"/>	<input type="text" value="MyIGW"/> <input type="button" value="X"/> <input type="button" value="Remove"/>

You can add 49 more tags.

Figura 16. Creación de Gateway de Internet.

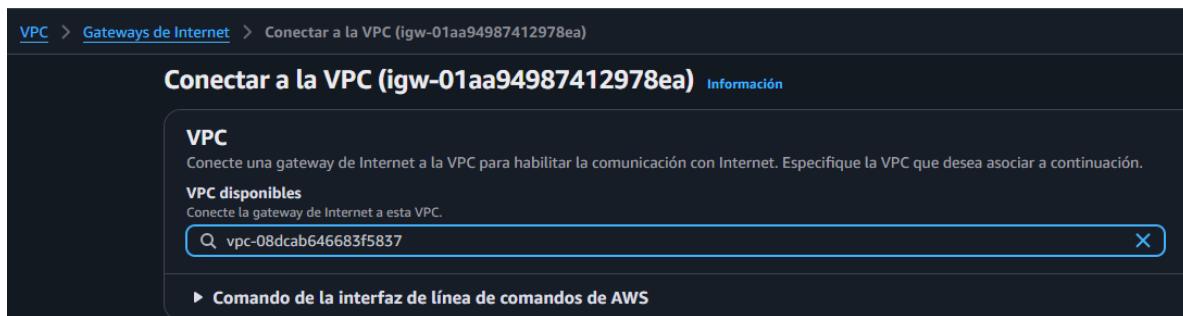


Figura 17. Asociación de la Gateway a la VPC.

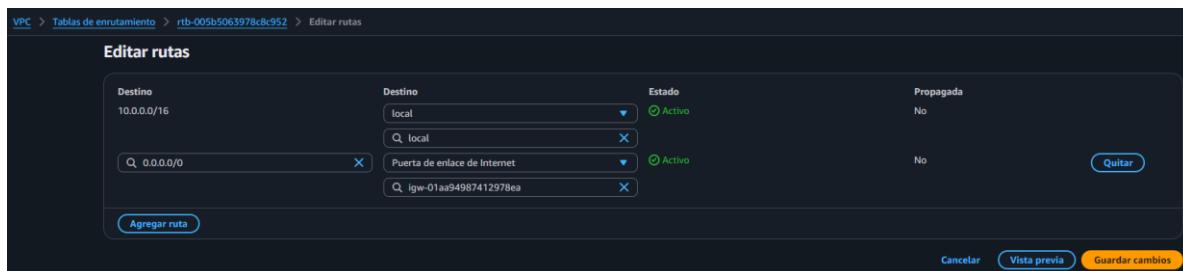


Figura 18. Configuración de las rutas de la tabla de enrutamiento.

3.2.2. Creación y configuración del SIEM

Para la implementación del SIEM se ha desplegado una instancia EC2 de tamaño t3.medium con sistema operativo Ubuntu 22.04, debido a que para los componentes principales de Wazuh (el gestor, el indexador y el panel web) se recomienda que las especificaciones más bajas sean 2 vCPU y 4GiB de RAM que se encuentran fuera de los recursos que permite el nivel gratuito de AWS. Esta elección garantiza un funcionamiento estable y permite un análisis y visualización de eventos en tiempo real fluidos.

La instancia se ha ubicado en la subred pública de la infraestructura. Para controlar el acceso a la instancia, se ha definido un grupo de seguridad que maximiza el control del tráfico. En concreto, este grupo permite las conexiones por el puerto 22 (SSH) exclusivamente desde la dirección IP local, habilita el acceso a Wazuh mediante el puerto 443 (HTTPS) para Wazuh tanto desde la IP local como desde la víctima, y permite todo el tráfico desde el SOAR.

A continuación, se muestra el paso a paso para la creación de la instancia:



Figura 19. Nombre y sistema operativo de la instancia.

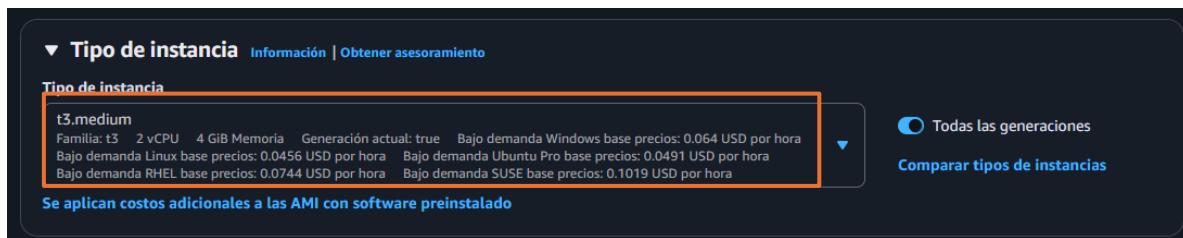


Figura 20. Selección de tipo de instancia.

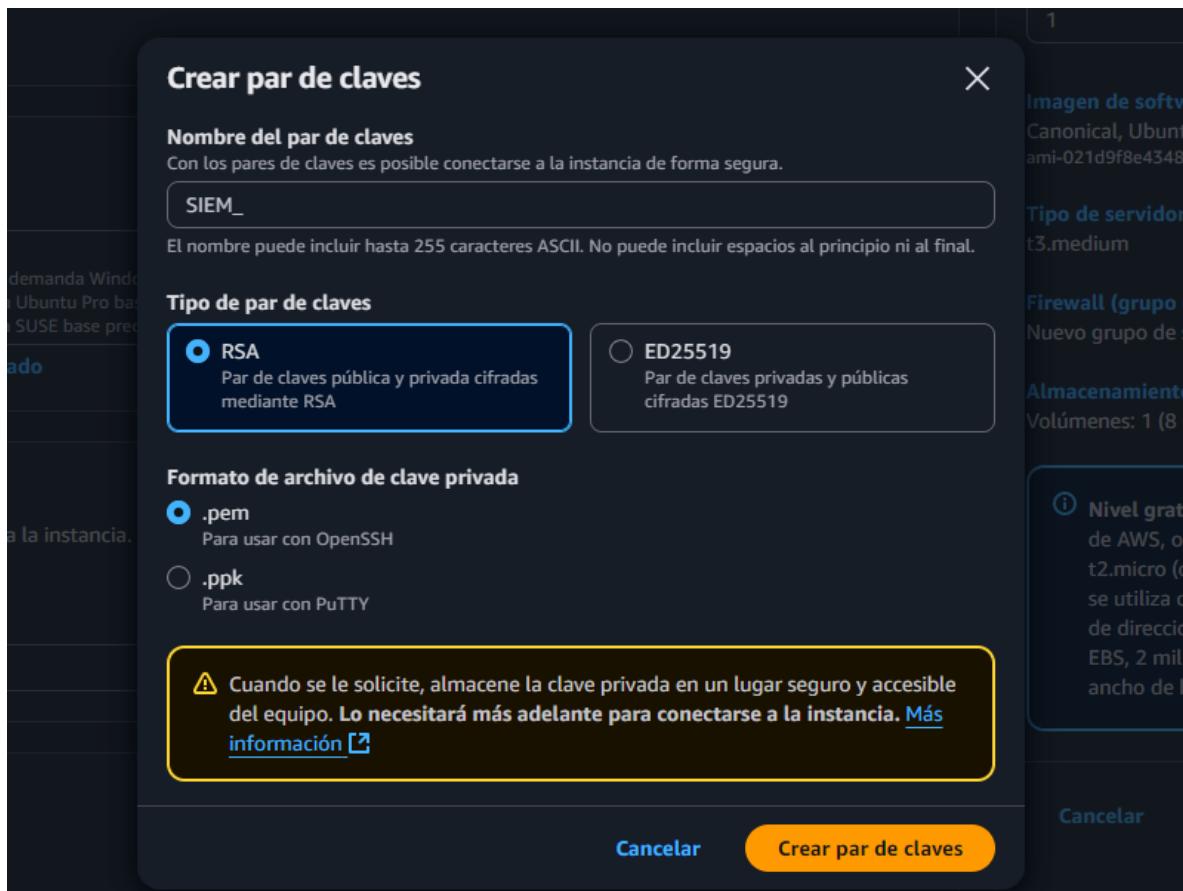


Figura 21. Creación par de claves.

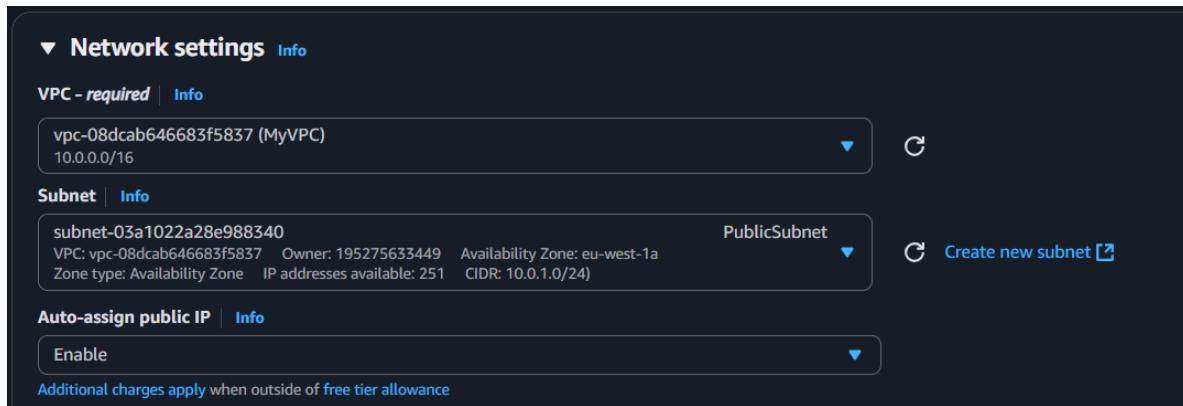


Figura 22. Configuración de red de la instancia.

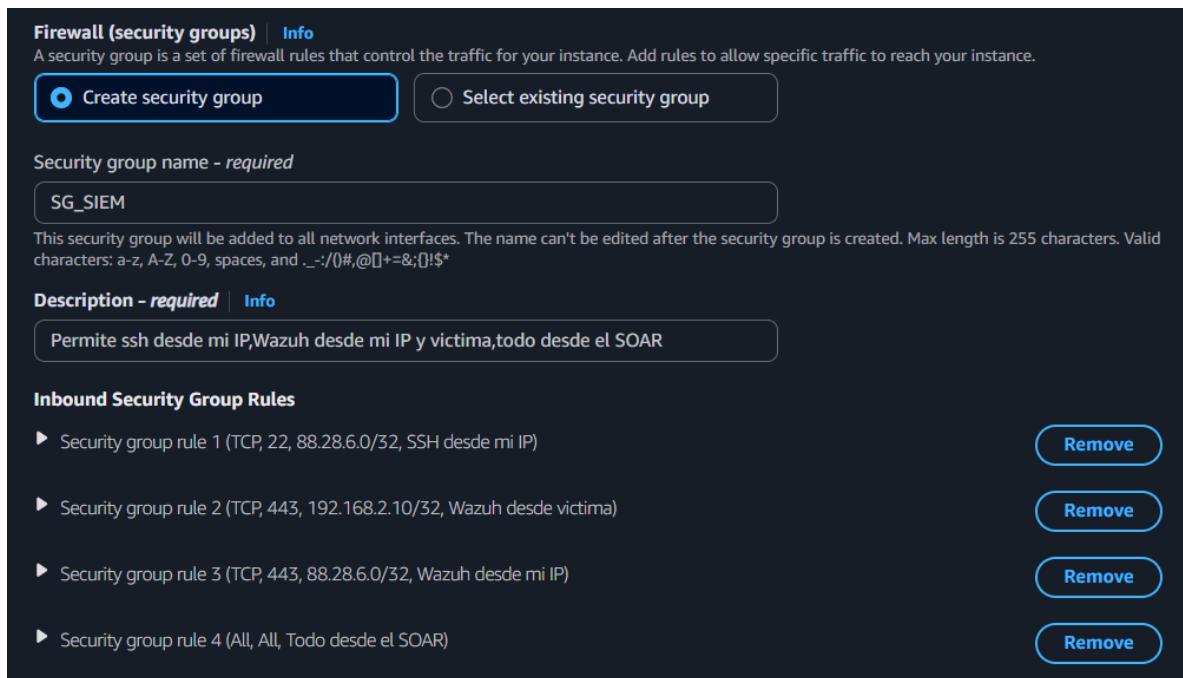


Figura 23. Creación del grupo de seguridad.

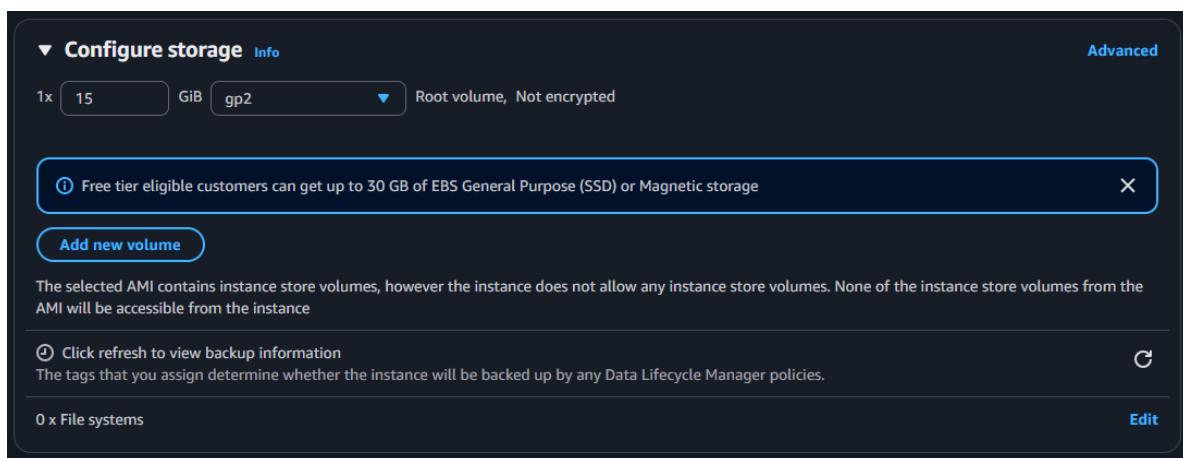


Figura 24. Configuración del almacenamiento.

Una vez configurada la instancia, se han instalado los tres componentes principales: el Wazuh Manager, encargado de la recolección y correlación de eventos; el Wazuh Indexer, basado en OpenSearch, que permite almacenar y consultar grandes volúmenes de logs; y el Wazuh Dashboard, que proporciona una interfaz gráfica interactiva desde la que se pueden visualizar alertas, gestionar agentes y configurar reglas de detección.

Para instalar los componentes hay que seguir los siguientes pasos:

Primero hay que conectarse a través de SSH a la instancia y arrancar como root:

```
sudo -i
```

Una vez arrancado como root, se puede empezar la instalación de Wazuh [8]. Para ello se ejecuta:

```
curl -s0 https://packages.wazuh.com/4.7/wazuh-install.sh&&sudo  
bash ./wazuh-install.sh -a -i
```

Transcurrido un tiempo, la instalación finaliza y se puede iniciar sesión en la interfaz gráfica de Wazuh buscando en el navegador https://IP_PUBLICA_SIEM.

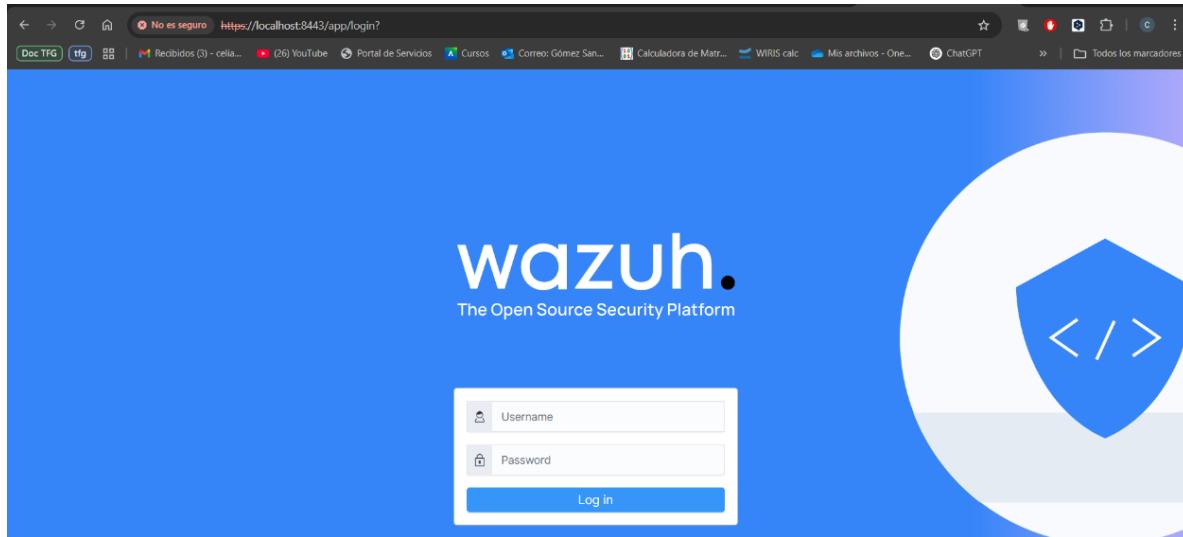


Figura 25. Pantalla de login de la interfaz gráfica de Wazuh

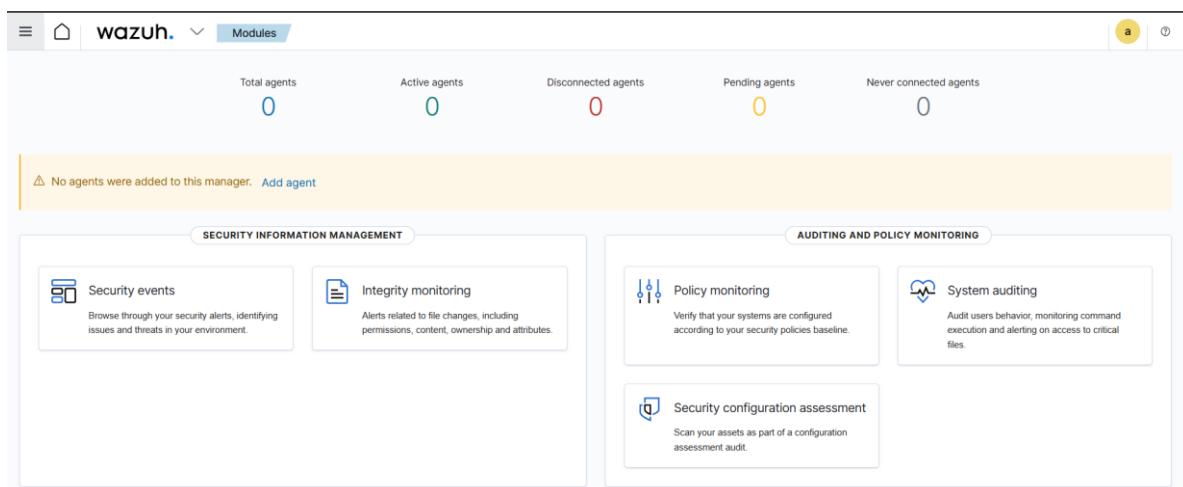


Figura 26. Pantalla principal de la interfaz gráfica de Wazuh.

3.2.3. Creación y configuración del SOAR

Para el despliegue del SOAR se ha utilizado también una instancia EC2 de tamaño t3.medium con sistema operativo Ubuntu 22.04. Esta configuración se debe a las necesidades mínimas de TheHive, MISP y Cortex. Además, al tratarse de servicios que requieren almacenamiento de datos estructurados

y archivos, se ha ampliado el volumen de disco de la instancia hasta 30 GiB, anticipando un uso creciente a medida que se acumulen evidencias e indicadores.

Durante la fase de configuración inicial, se ha desplegado en la subred pública y se ha definido un grupo de seguridad específico para esta instancia, limitando el tráfico como en el SIEM. En este caso, se han permitido las conexiones por el puerto 22 (SSH) exclusivamente desde la dirección IP local, se han habilitado también los puertos 9000,9001 y 443 (HTTPS) desde la dirección IP local para la conexión con TheHive, Cortex y MISP respectivamente, y se han abierto todas las conexiones con el SIEM.

A continuación, se muestra el paso a paso para la creación de la instancia:



Figura 27. Nombre y sistema operativo de la instancia.

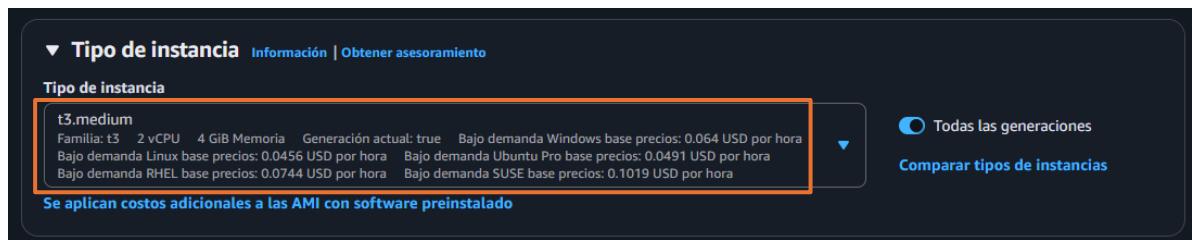


Figura 28. Selección de tipo de instancia.



Figura 29. Creación par de claves.

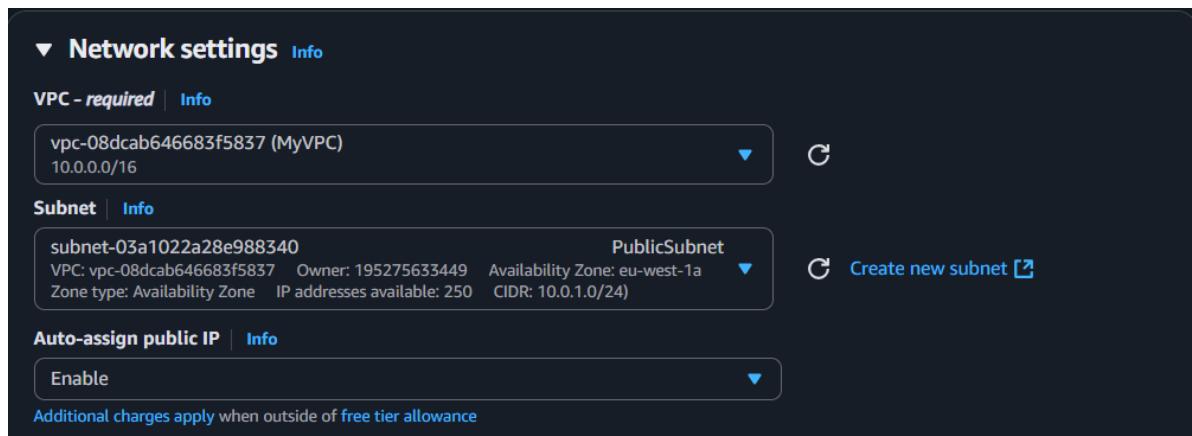


Figura 30. Configuración de red de la instancia.

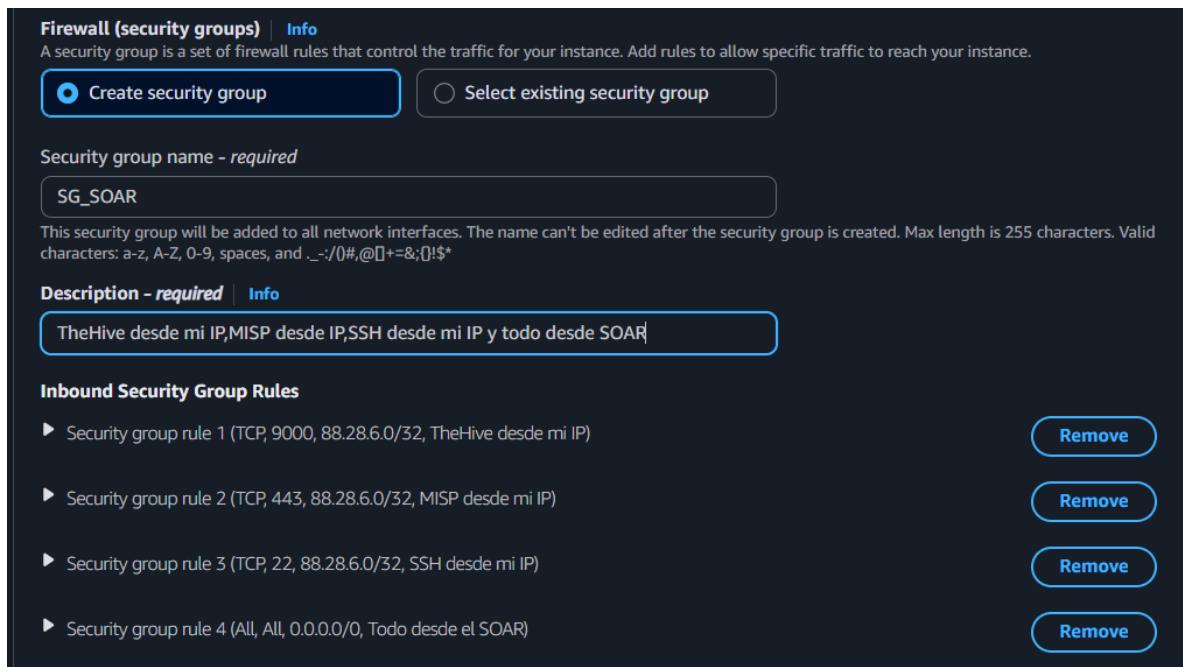


Figura 31. Creación del grupo de seguridad.

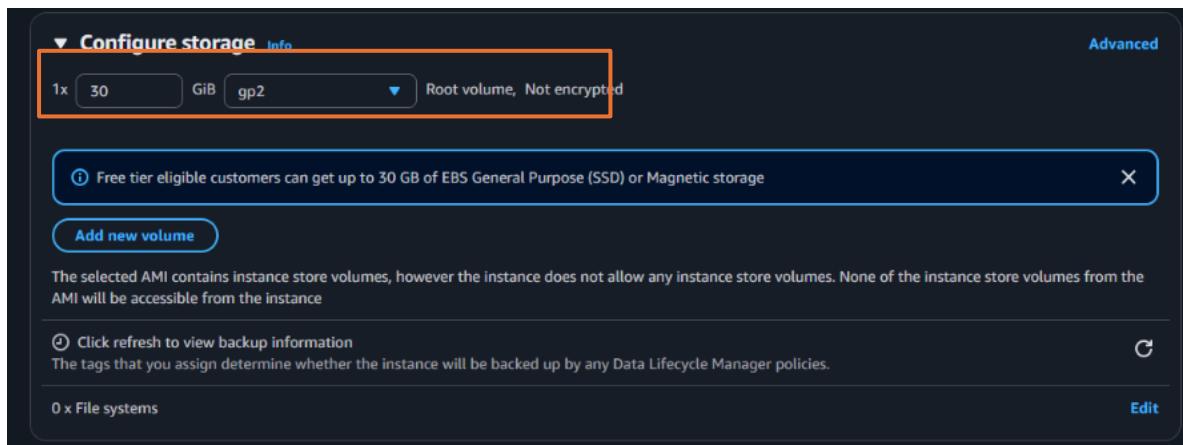


Figura 32. Configuración del almacenamiento.

Una vez desplegada la instancia, se procede a la instalación de los tres componentes del SOAR: TheHive, para la gestión de incidentes, Cortex para la respuesta a incidente y MISP, como analizador.

Para llevar a cabo la instalación de TheHive, se ha seguido el procedimiento descrito en la documentación oficial de StrangeBee [9], que contempla una instalación paso a paso de todos los componentes necesarios. Este proceso incluye la configuración de la base de datos Apache Cassandra para almacenar los casos, OpenSearch como motor de búsqueda, y el propio servicio de TheHive.

El acceso a la instancia se realiza a través de SSH y escalar los privilegios a root para poder instalar las dependencias y configurar los repositorios

necesarios:

```
sudo -i
```

El primer paso es instalar las herramientas y dependencias básicas para el correcto funcionamiento del sistema.

```
sudo apt update
apt install wget gnupg apt-transport-https git ca-certificates
ca-certificates-java curl software-properties-common python3-
pip lsb-release
```

Con las dependencias instaladas, hay que instalar el entorno de ejecución requerido por TheHive y por la base de datos, Java 11.

```
sudo apt update
sudo apt install openjdk-11-jdk -y
java -version
```

Transcurrido un tiempo, la instalación habrá finalizado. Es recomendable asegurarse de que las dependencias y el entorno se han instalado correctamente.

Completado este proceso, se procede con la instalación de Apache Cassandra. Primero se agregan las referencias del repositorio oficial de Cassandra al sistema.

```
echo "deb [trusted=yes]
https://apache.jfrog.io/artifactory/cassandra-deb 41x main" |
sudo tee /etc/apt/sources.list.d/cassandra.list
```

```
sudo apt update
sudo apt install cassandra -y
```

Tras la configuración, se puede iniciar y habilitar el servicio:

```
sudo systemctl enable cassandra
sudo systemctl start cassandra
```

Con el entorno preparado se procede a la instalación y despliegue de TheHive.

```
echo "deb [trusted=yes] https://deb.strangebee.com thehive-5.4
main" | sudo tee /etc/apt/sources.list.d/thehive.list
curl https://deb.strangebee.com/strangebee.gpg | sudo gpg --
dearmor -o /usr/share/keyrings/strangebee-archive-keyring.gpg
```

```
sudo apt update  
sudo apt install thehive -y
```

De nuevo, de la configuración predeterminada, solo se cambia el nombre del clúster en el archivo *application.conf*.

```
sed -i 's/cluster-name = thp/cluster-name = thehive/'  
/etc/thehive/application.conf
```

Por último, se habilita e inicia el sistema.

```
systemctl start thehive  
systemctl enable thehive
```

Con TheHive ya instalado, se puede acceder ya al dashboard desde internet.

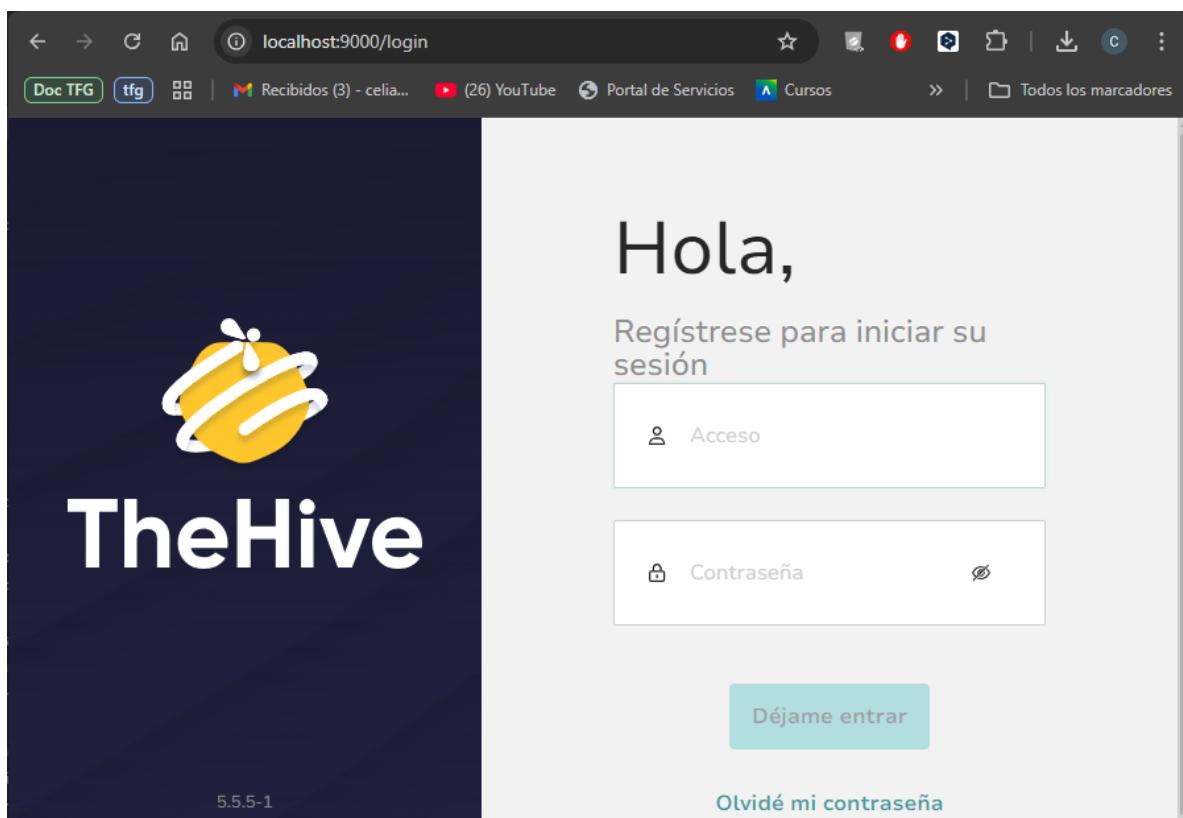


Figura 33. Pantalla de login del dashboard de TheHive.

Por defecto, los módulos de Cortex y MISP están activos en TheHive, aunque es necesario instalarlos y configurarlos manualmente. En este caso, se ha instalado MISP siguiendo las instrucciones de la documentación correspondiente [10].

En primer lugar, se ha instalado el cliente MySQL, el cual es fundamental para el instalador de MISP.

```
sudo apt-get install mysql-client -y
```

Posteriormente, se ha ejecutado un script oficial de instalación de MISP que automatiza la instalación de los principales componentes necesarios, como Apache, PHP o MariaDB.

```
wget  
https://raw.githubusercontent.com/MISP/MISP/2.4/INSTALL/INSTALL  
.sh
```

Una vez finalizada la instalación, se puede acceder a la interfaz gráfica de MISP desde el navegador con http://IP_PUBLICA_SOAR.

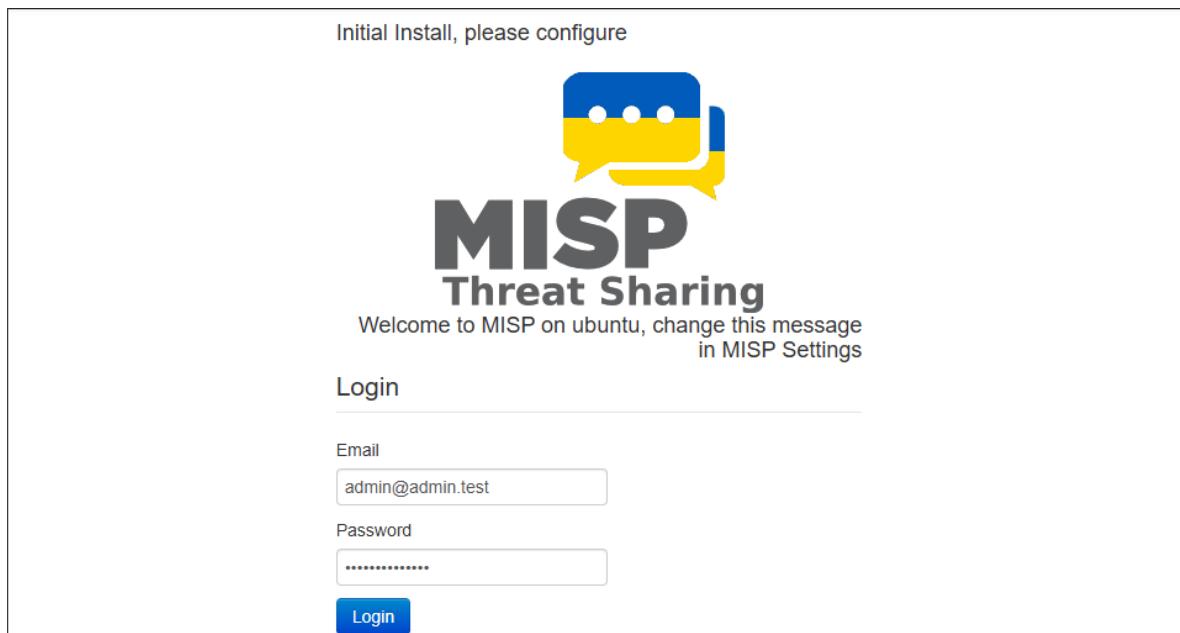


Figura 34. Pantalla principal de login de MISP.

El procedimiento de implementación del módulo Cortex se ha realizado siguiendo las directrices proporcionadas por la documentación oficial de StrangeeBee dedicada a este tema [11].

Para realizar una instalación segura y verificada, primero hay que preparar el entorno descargando el paquete de instalación junto con sus archivos de verificación desde la página web oficial:

```
wget -O /tmp/cortex_3.2.1-2_all.deb  
https://cortex.download.strangebee.com/3.2/deb/cortex_3.2.1-  
2_all.deb  
wget -O /tmp/cortex_3.2.1-2_all.deb.sha256  
https://cortex.download.strangebee.com/3.2/sha256/cortex_3.2.1-  
2_all.deb.sha256  
wget -O /tmp/cortex_3.2.1-2_all.deb.asc  
https://cortex.download.strangebee.com/3.2/asc/cortex_3.2.1-  
2_all.deb.asc
```

Luego de la descarga, se procede con la instalación:

```
sudo apt-get install /tmp/cortex_3.2.1-2_all.deb
```

El inicio de Cortex requiere establecer una clave secreta para la instancia en el archivo `/etc/cortex/secret.conf` e incluirla en el archivo de configuración de Cortex `/etc/cortex/application.conf`.

```
cat > /etc/cortex/secret.conf << _EOF_  
play.http.secret.key="$(cat /dev/urandom | tr -dc 'a-zA-Z0-9' |  
fold -w 64 | head -n 1)"  
_EOF_
```

Terminada la configuración, se puede iniciar el servicio y acceder a través de http://IP_PUBLICA_SOAR:9001/ donde hay que configurar en primer lugar el usuario administrador.

```
systemctl start cortex
```

Create administrator account

Login	<input type="text" value="Login"/>
Name	<input type="text" value="Name"/>
Password	<input type="password" value="Password"/>
<input type="button" value="Create"/>	

Figura 35. Creación usuario administrador Cortex.

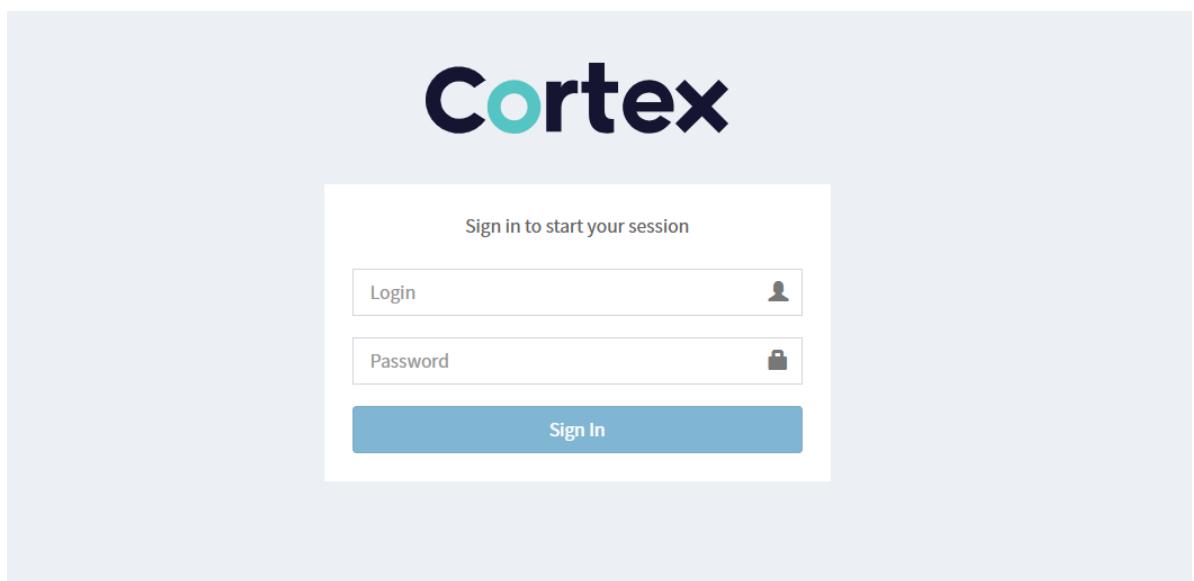


Figura 36. Pantalla login de Cortex.

Una vez dentro de la interfaz como administrador, hay que crear una organización y un usuario con rol *orgadmin* dentro de ella.

Create organization

Name *

Description *

Cancel

* Required field

Save

Figura 37. Creación de organización en Cortex.

Add user

Login *

Full name *

Roles *

Cancel

* Required field

Save user

Figura 38. Creación de usuario en Cortex

3.2.4. Creación y configuración de la víctima

Para el despliegue de la víctima se han utilizado dos máquinas virtuales creadas en VMWare, una con Windows 10 como estación de trabajo y otra con Windows Server 2019 como controlador de dominio.

3.2.4.1. Creación y configuración del Server

Para la instalación y configuración de la máquina Server se ha seguido la documentación correspondiente [12].

En primer lugar, se ha lanzado una nueva máquina virtual en VMWare mediante la imagen ISO de Windows Server 2019 descargada del sitio oficial de Windows [13]. Para su configuración, no es necesario utilizar una clave de producto y se recomienda instalar la versión Datacenter para tener menos restricciones. Para la capacidad del disco se deben seguir las recomendaciones, en este caso 60GB. Antes de terminar de crear la

máquina, es importante modificar el hardware para configurarlo de tal forma que esté conectada con la estación de trabajo a través de un segmento LAN.

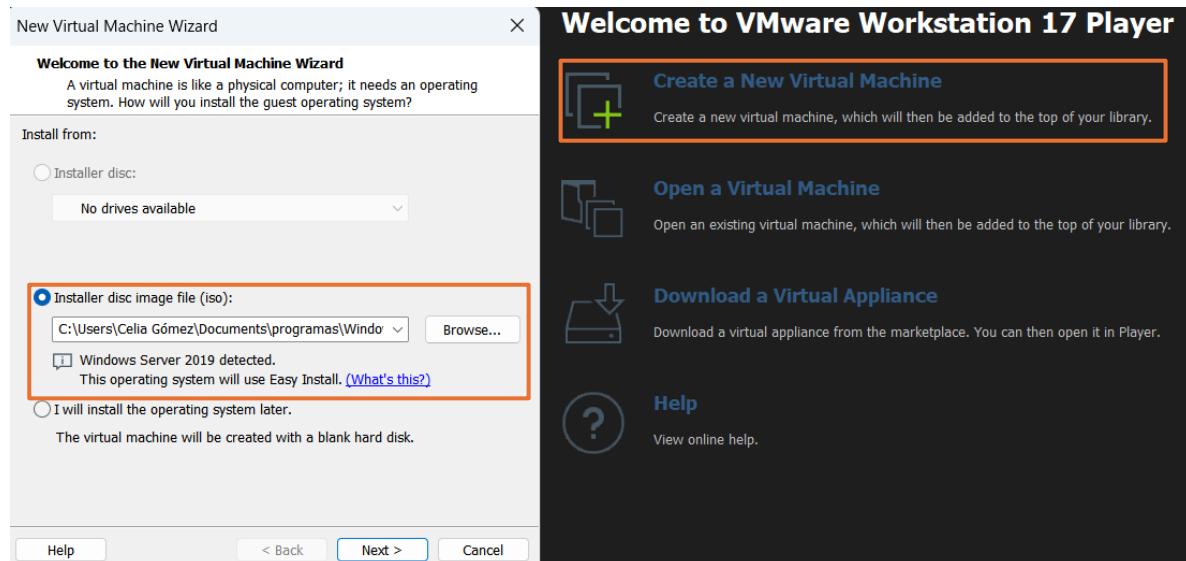


Figura 39. Creación máquina virtual Server.

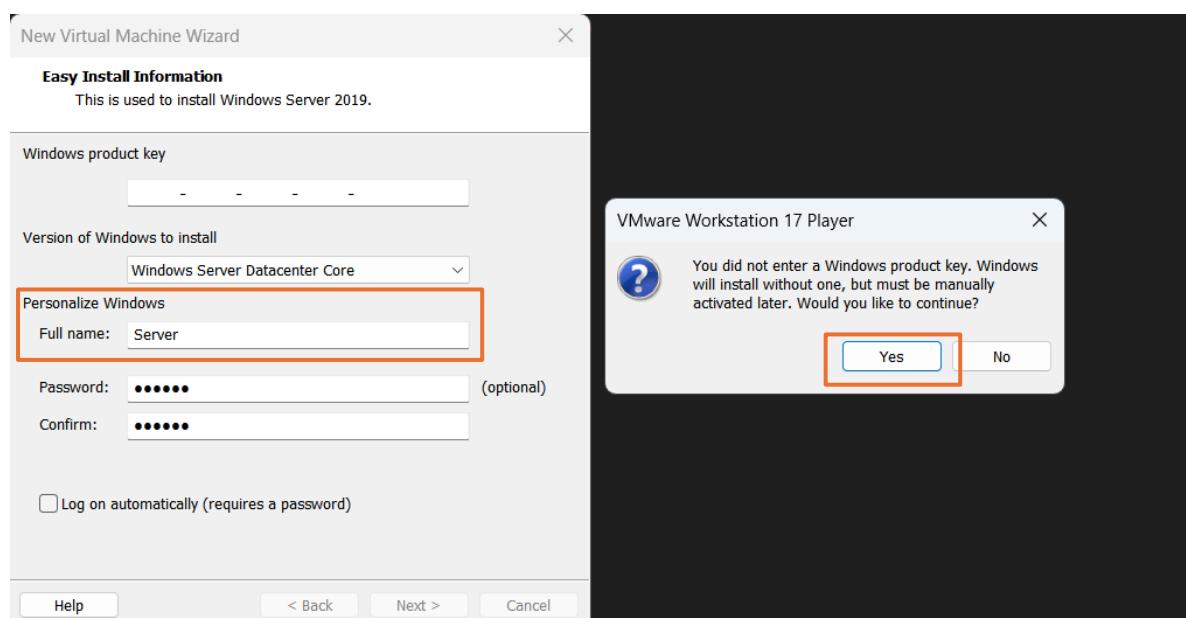


Figura 40. Configuración máquina Server.

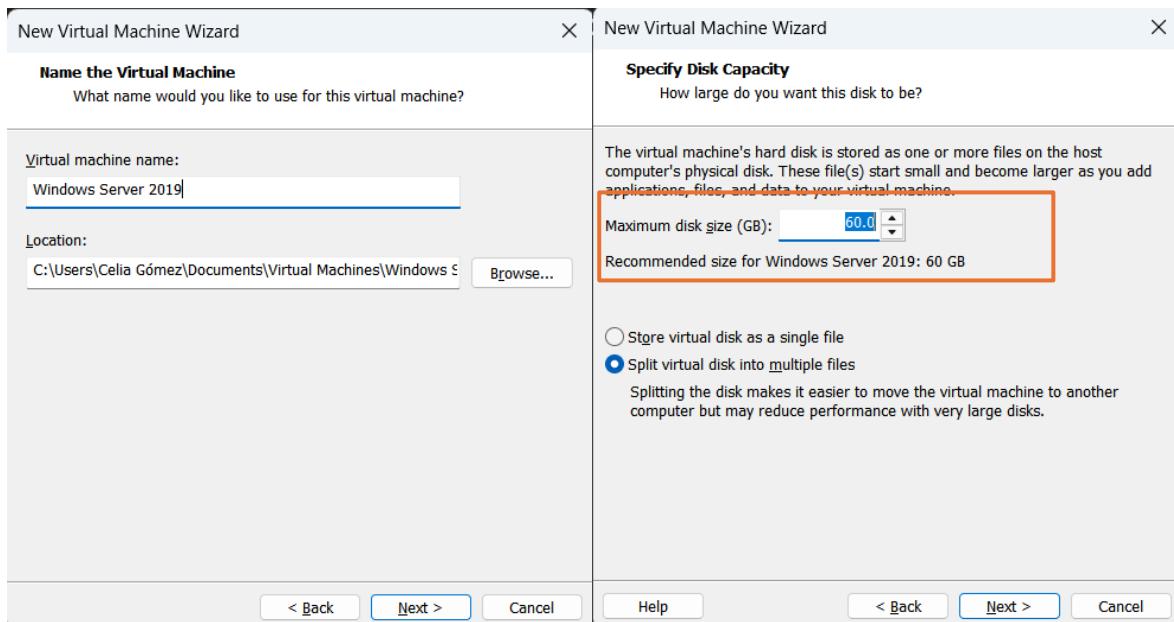


Figura 41. Configuración máquina Server parte 2.

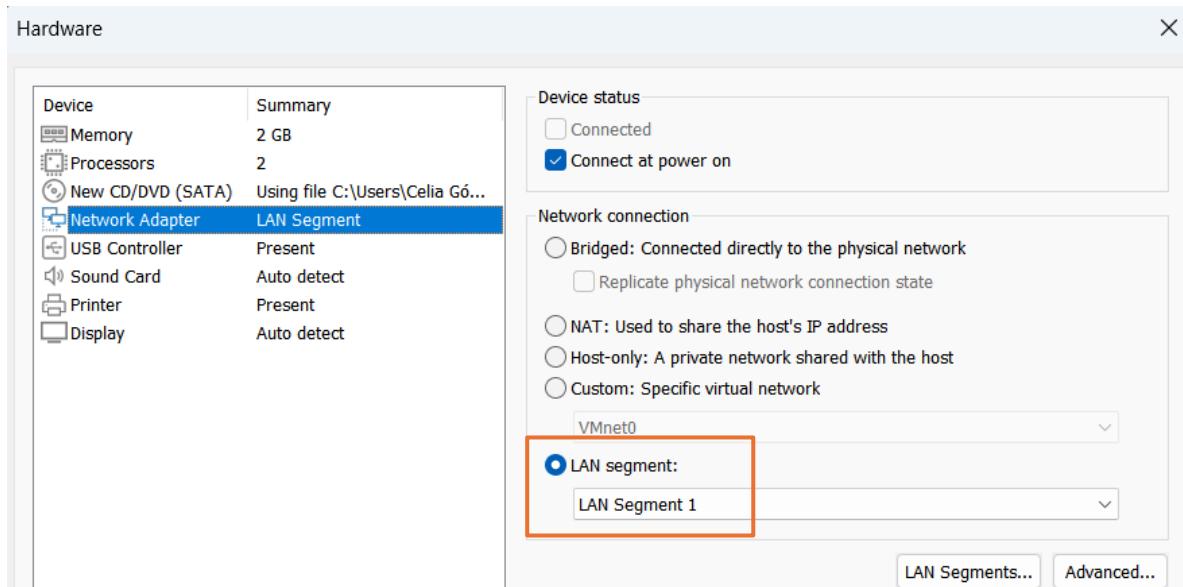


Figura 42. Configuración máquina Server parte 3.

Una vez creada, se inicia el proceso de instalación del sistema operativo donde se siguen las indicaciones del asistente de configuración. Es importante que en la selección de la versión de Server a instalar se seleccione de nuevo la opción de *Datacenter con experiencia de escritorio* para poder trabajar con interfaz gráfica.

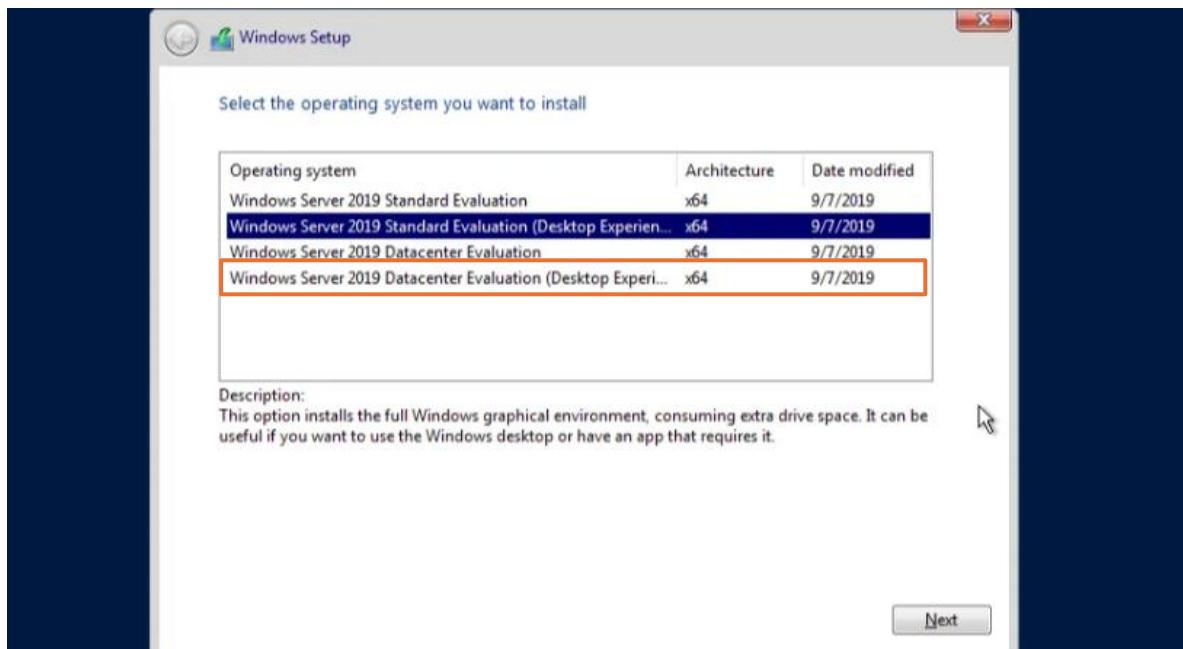


Figura 43. Selección de versión Windows Server 2019.

Para finalizar la instalación, se selecciona la partición 0 del disco.

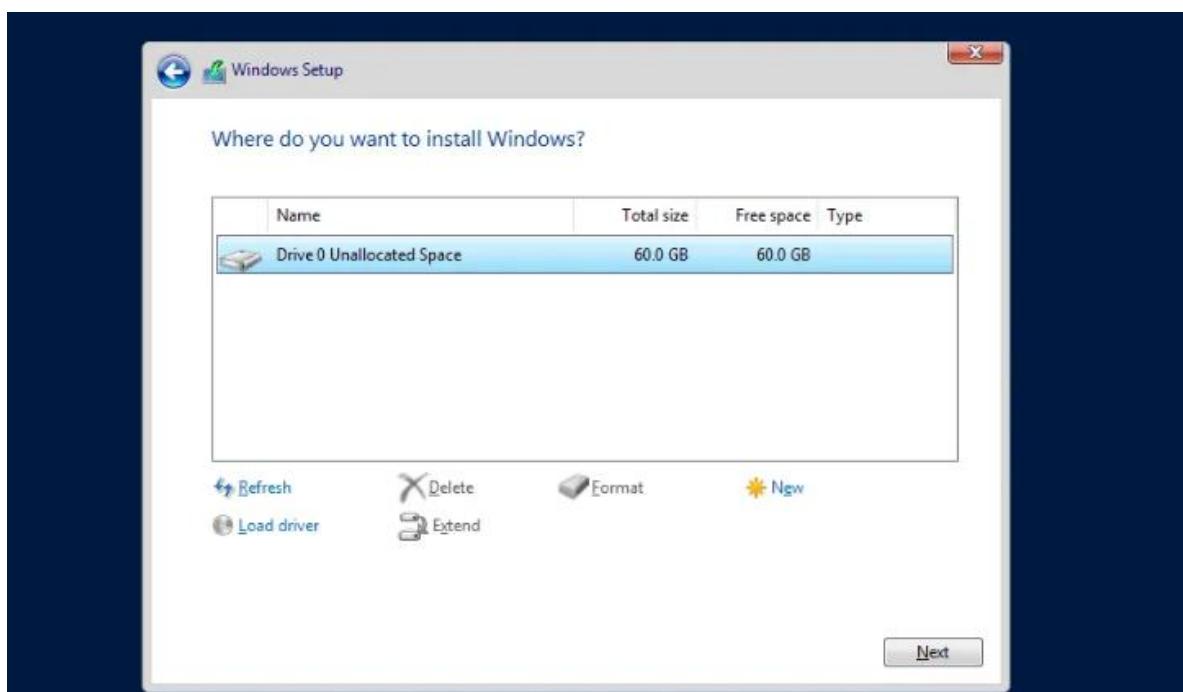


Figura 44. Ubicación del sistema operativo en el disco.

Durante la configuración inicial de Windows Server se crea la cuenta de administrador que tienes plenos derechos administrativos y es la que se usará siempre en esta máquina.

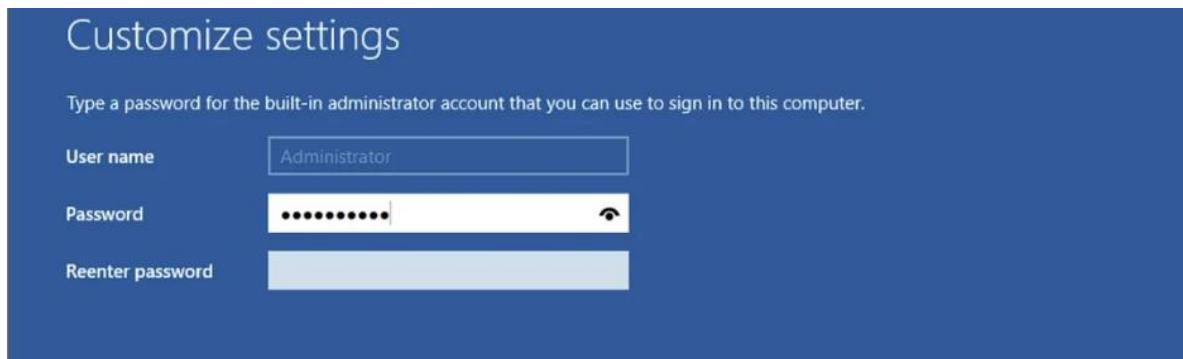


Figura 45. Creación usuario administrador.

Una vez finalizada la instalación e iniciada la sesión, se accede a la pantalla del Server Manager por defecto.

Este servicio permite configurar los servicios de Dominio de Active Directory (AD DS) para gestionar la gestión centralizada de usuarios, equipos y políticas dentro de un entorno de red. Para esta configuración, se ha seguido el manual correspondiente [14].

Para comenzar, desde el panel de inicio del Server Manager se selecciona la opción de Agregar roles y características.

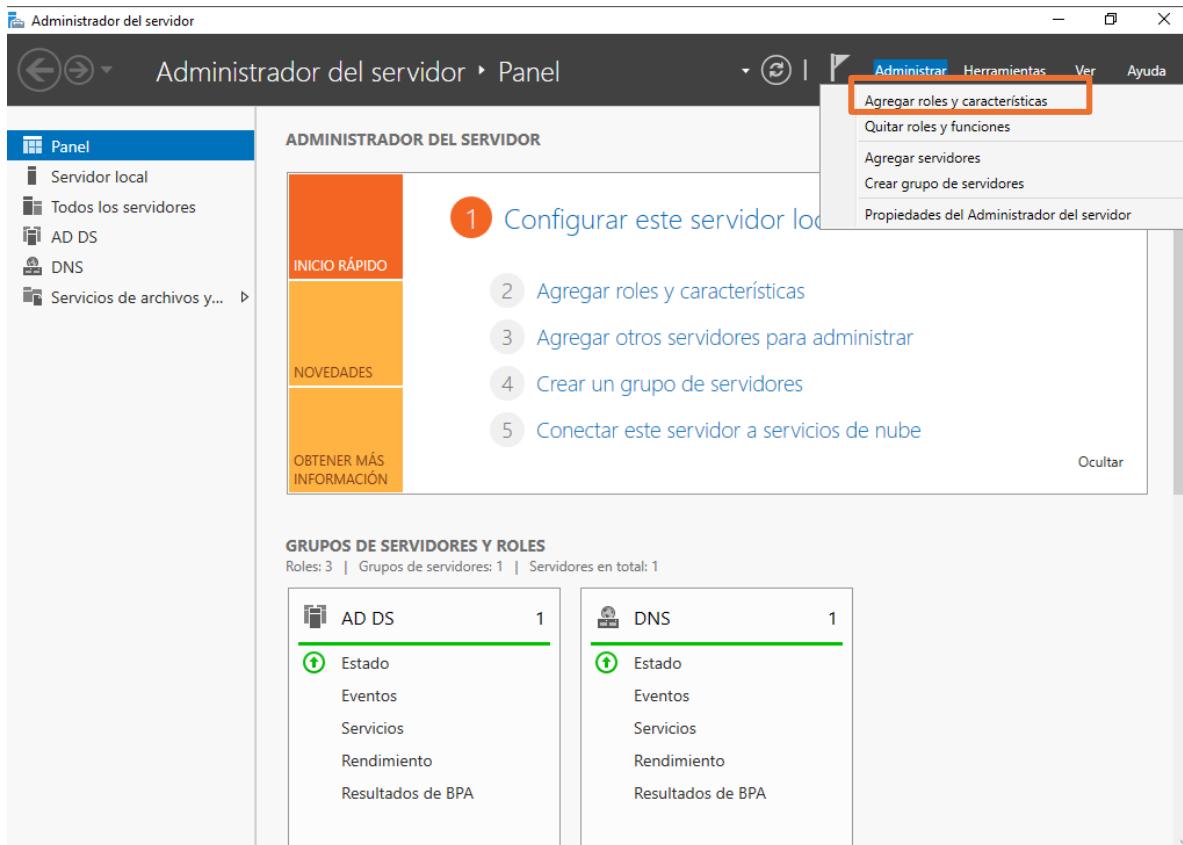


Figura 46. Panel de inicio Server Manager.

A continuación, se selecciona la instalación basada en roles o características, el servidor local como destino y el rol de AD DS.

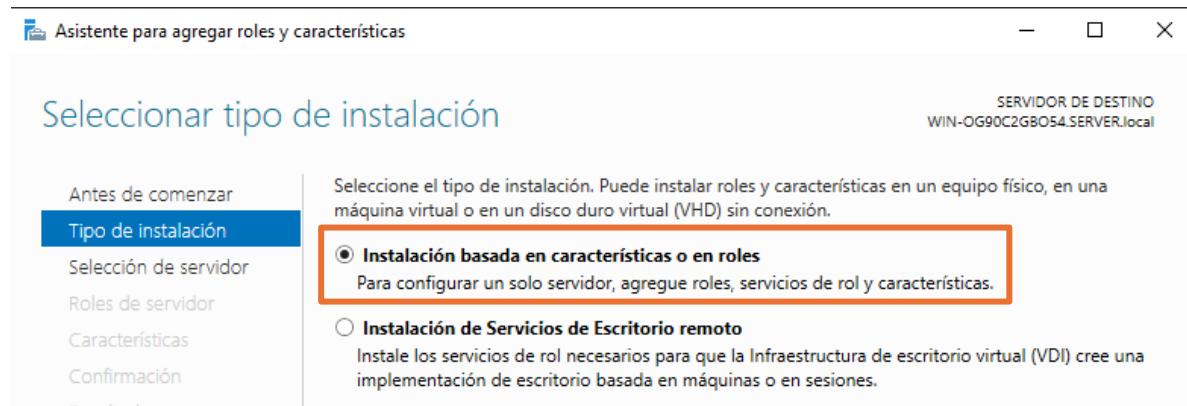


Figura 47. Tipo de instalación.



Figura 48. Servidor de destino.

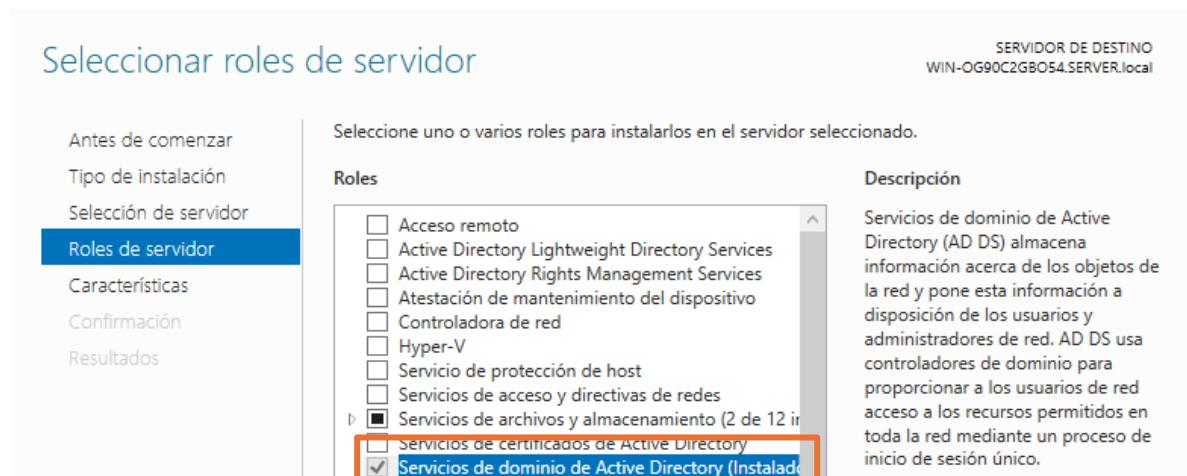


Figura 49. Roles del servidor.

Una vez instalado el rol, el sistema notifica la disponibilidad del Asistente de configuración de Active Directory mediante una alerta en la bandera de notificaciones del Administrador del servidor. Al hacer clic en dicha alerta,

se accede al asistente de configuración de AD DS. Una vez en el asistente, se agrega un nuevo bosque y se establecen el nivel funcional tanto del bosque como del dominio en la versión más reciente compatible (Windows Server 2016). Es importante comprobar que las opciones DNS y GC estén marcadas.

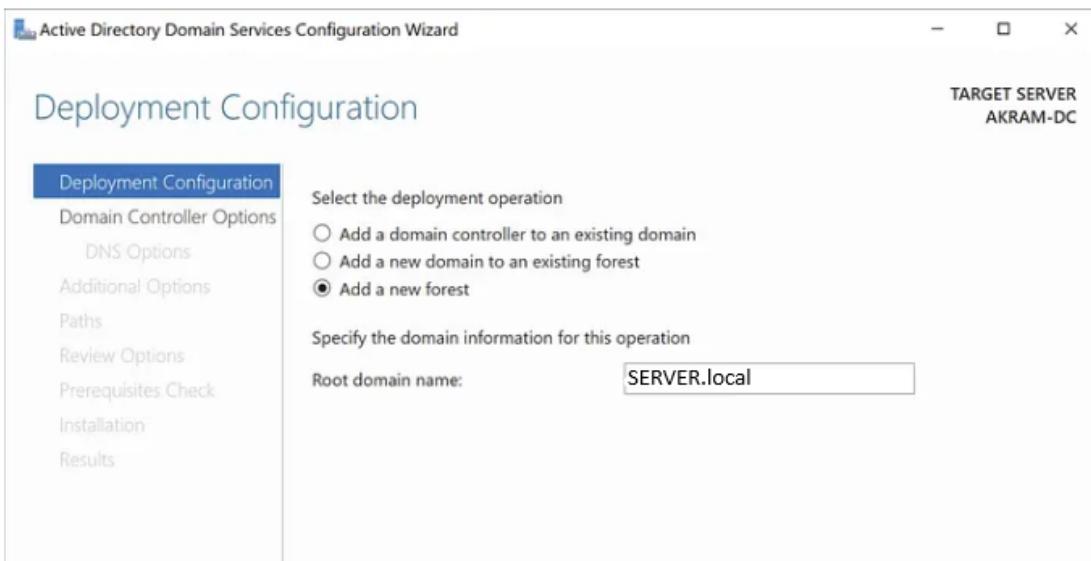


Figura 50. Creación de nuevo bosque.



Figura 51. Selección de opciones del controlador de dominio.

Tras completar estos pasos, el asistente finaliza la configuración y prepara al sistema para reiniciar aplicando así los cambios necesarios. Finalmente, se crea un usuario en el AD con el objetivo de disponer de una identidad gestionada dentro del dominio para la estación de trabajo.

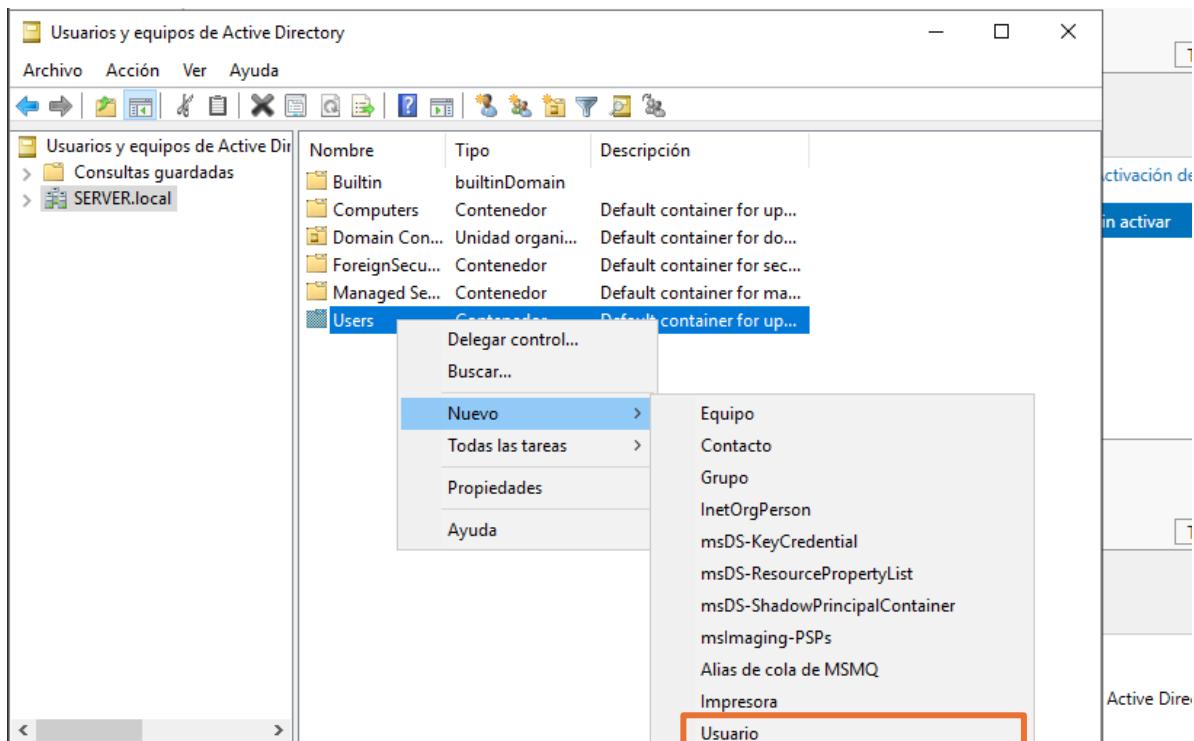


Figura 52. Creación nuevo usuario en el AD.

3.2.4.2. Creación y configuración de la estación de trabajo

La estación de trabajo con Windows 10 se ha configurado según la documentación correspondiente [15].

Primero, se ha lanzado una nueva máquina virtual en VMWare mediante la imagen ISO de Windows 10 Pro descargada del sitio oficial de Windows [16]. Para la capacidad del disco se deben seguir las recomendaciones, en este caso 60GB. Antes de terminar de crear la máquina, es importante modificar el hardware para configurarlo de tal forma que esté conectada con el servidor a través de un segmento LAN y la conexión a Internet esté en modo *Bridge*.

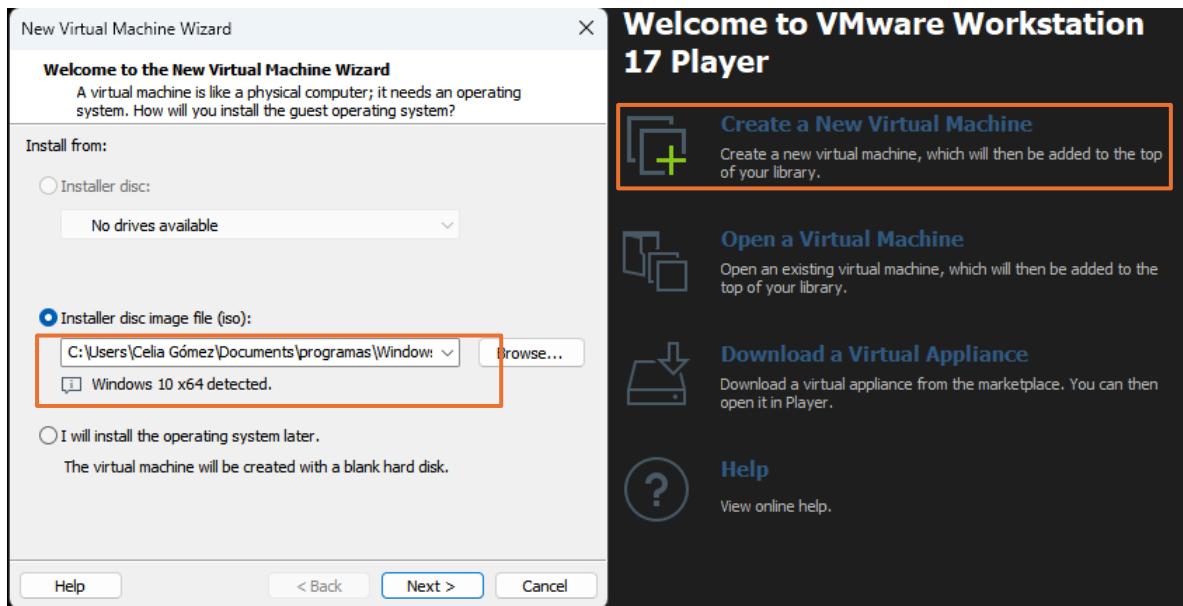


Figura 53. Creación máquina virtual de la estación de trabajo.

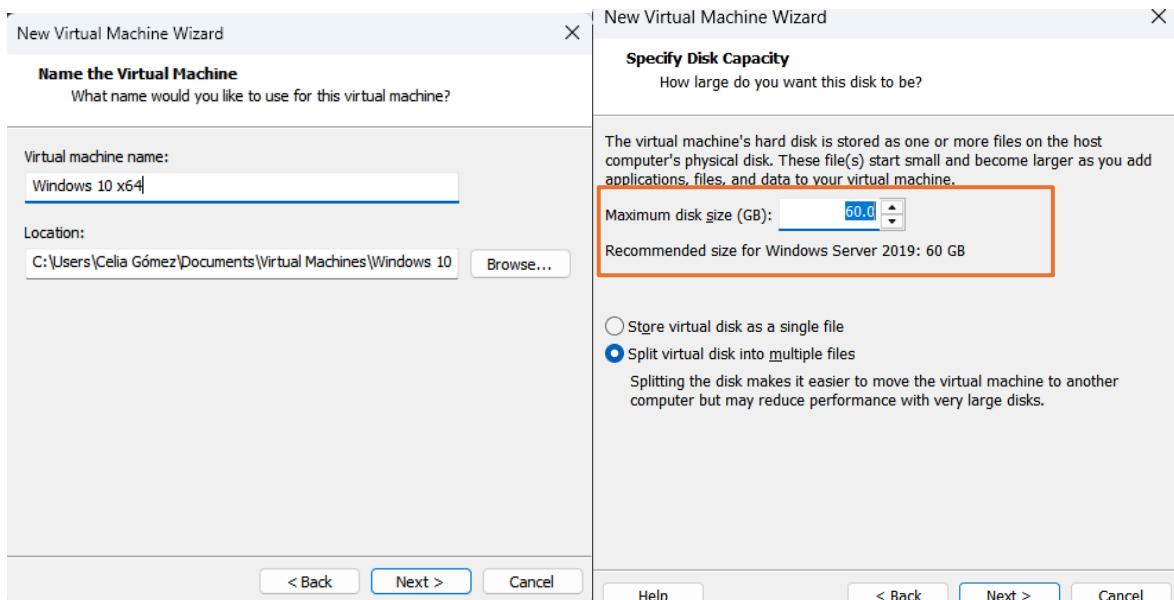


Figura 54. Configuración estación de trabajo.

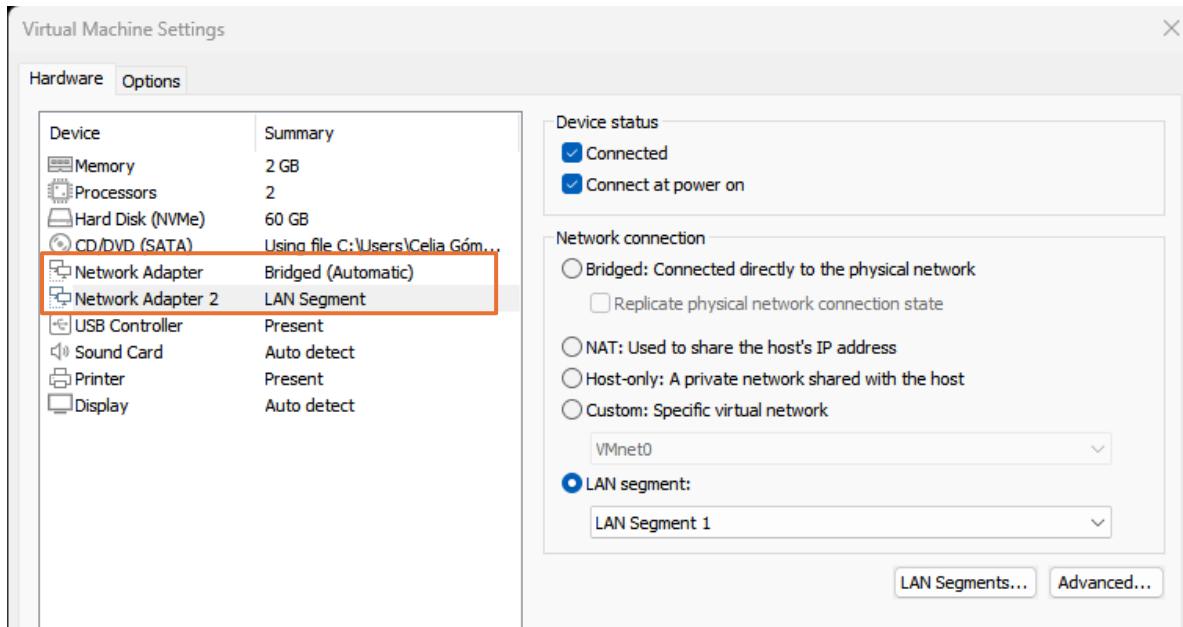


Figura 55. Configuración estación de trabajo parte 2.

Con la máquina ya creada, se puede iniciar el proceso de instalación del sistema operativo. No es necesario tener una clave de producto así que se continúa sin ella y se selecciona la instalación personalizada. Como con el servidor, se selecciona el disco duro virtual para instalar el SO. La configuración posterior consiste en ir siguiendo los pasos que te indica Windows y seleccionando las opciones que deseas.

Para simular ataques reales en un entorno controlado, facilitando la recolección y análisis de logs desde ambas máquinas para alimentar las herramientas del SOC y desarrollar capacidades de detección se ha unido al dominio y configurado como estación de trabajo típica de un usuario final. Para ello, hay que cambiar el nombre del ordenador y agregarle el dominio correspondiente como indica la documentación [14].

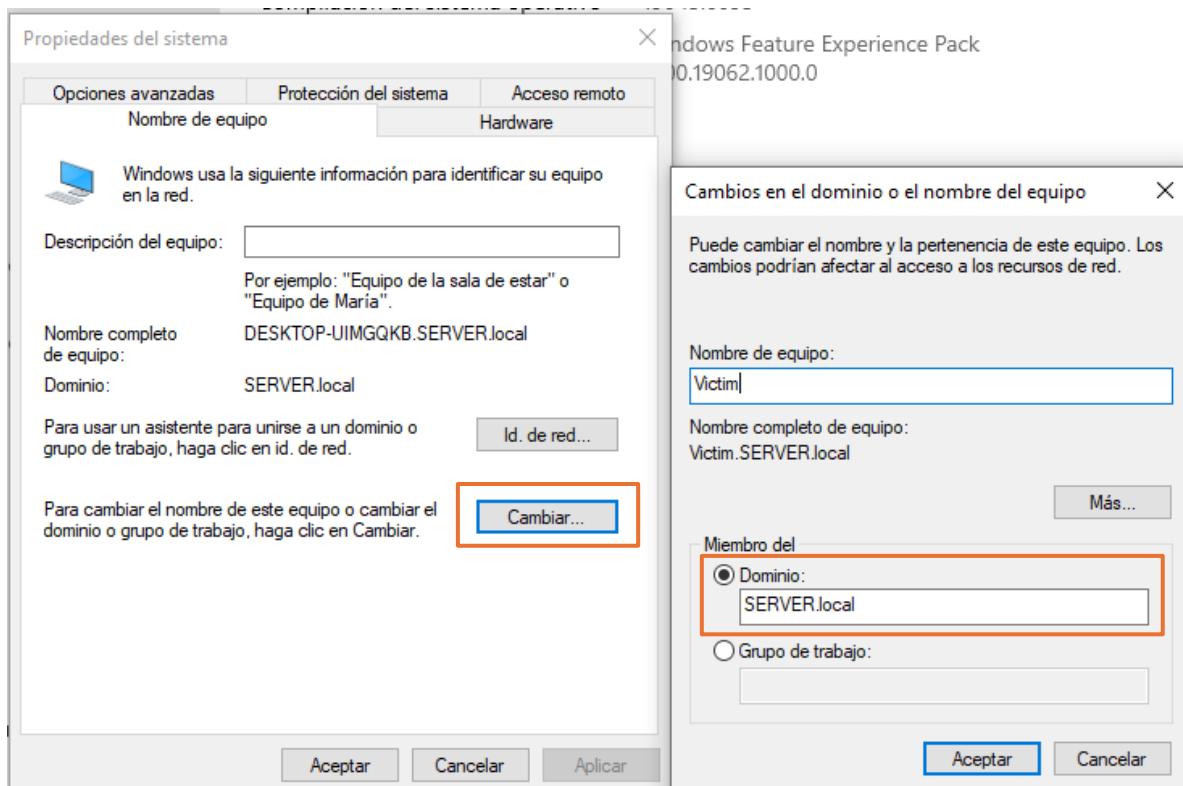


Figura 56. Cambio de nombre y agregación al dominio.

3.2.5. Creación y configuración del atacante

Para la implementación del entorno atacante se ha creado también una máquina virtual en VMWare utilizando esta vez una imagen preconfigurada de Kali Linux, descargada desde el sitio web oficial del proyecto Kali [17].

Una vez descargada la imagen en formato VMX, se ha importado a VMWare con la opción Open a Virtual Machine.

Antes de comenzar a trabajar con ella hay que verificar que la configuración de red a nivel hardware tiene dos adaptadores de red, uno configurado como Bridge para la conexión a internet y el segundo conectado a la red interna compartida con el servidor y la máquina de trabajo correspondiente al segmento LAN.

Al iniciar la máquina virtual, se carga el sistema operativo Kali Linux preinstalado, además el entorno está diseñado para proporcionar acceso rápido a herramientas de *pentesting* y análisis de seguridad.

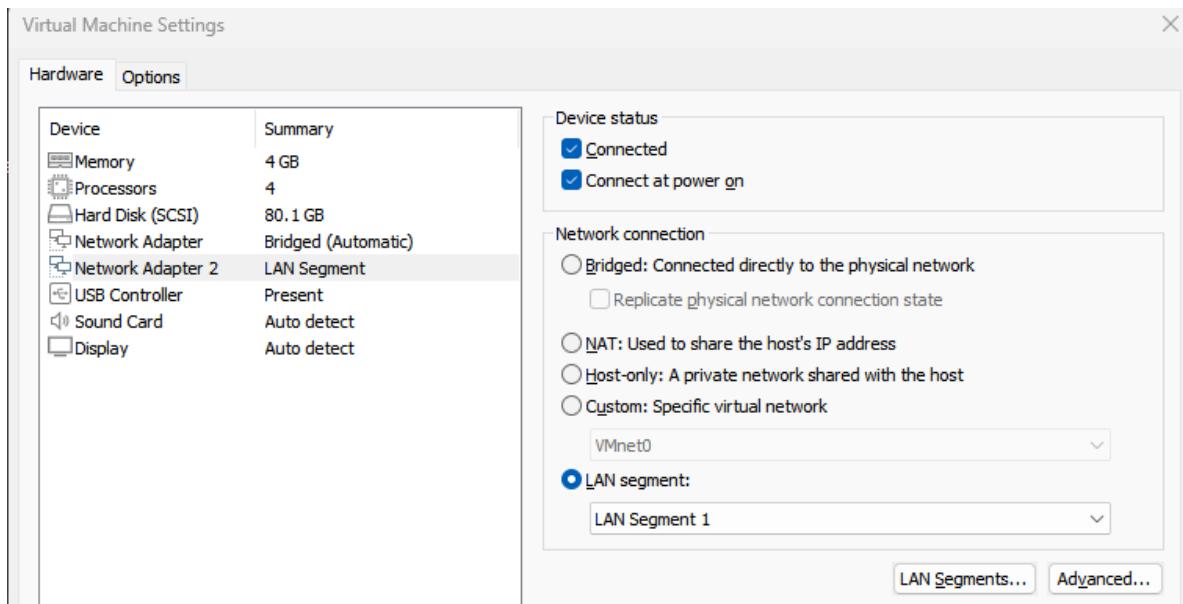


Figura 57. Configuración hardware máquina atacante.

3.3. Conectividad e interrelación entre componentes

3.3.1. Conexión de TheHive con MISP

La integración entre estos dos componentes del SOAR es imprescindible ya que permite recuperar automáticamente eventos MISP filtrados como alertas en TheHive. Para establecer esta conexión, se ha seguido la documentación pertinente [18].

En primer lugar, desde la interfaz web de MISP se crea una API Key para el usuario por defecto ya que se necesita para vincular MISP a TheHive.

ID	Org	Role	Email	NIDS SID	Last Login	Created	Last API Access	Actions
1	ORGNAME	admin	admin@admin.test	4000000	2025-08-12 00:17:09	2025-08-03 21:32:20	2025-08-03 21:32:20	

Figura 58. Creación clave de autorización MISP.

User admin@admin.test

ID	1
Email	admin@admin.test
Organisation	ORGNAME
Role	admin
TOTP	No
Email notifications	Event published notification: No Daily notifications: No Weekly notifications: No Monthly notifications: No
Contact alert enabled	No
Invited By	N/A
Org admin	
NIDS Start SID	400000
Terms accepted	No
Must change password	No
PGP key	No
Created	N/A
Last password change	2025-08-03 20:43:08
News read at	N/A
Disabled	No

[Download user profile for data portability](#) [Review user logs](#) [Review user logins](#)

[Auth keys](#)

Figura 59. Creación API KEY MISP parte 2.

Auth keys							
#	User	Auth Key	Expiration	Last used	Comment	Allowed IPs	Seen IPs
2	admin@admin.test	Uup8*****HAoR	Indefinite	Never	Initial auto-generated key		

[+ Add authentication key](#) [Filter](#)

Figura 60. Creación API KEY MISP parte 3.

Add auth key

Auth keys are used for API access. A user can have more than one authkey, so if you would like to use separate keys per tool that queries MISP, add additional keys. Use the comment field to make identifying your keys easier.

User: admin@admin.test

Comment: TheHive API Key

Allowed IPs:

Expiration (keep empty for indefinite): YYYY-MM-DD

Read only (it will unset all permissions. This should not be used for sync users)

Submit **Cancel**

Figura 61. Creación API KEY MISP parte 4.

Con la clave de autorización de MISP generada, se accede a la interfaz web TheHive a la sección de Platform Management, se selecciona la pestaña Connectors y, dentro de esta, la subpestaña MISP para comenzar la configuración.

Gestión de plataforma

Licencia Estado Marca Connectors Autenticación SMTP Puntos finales globales Servidores LDAP

Corteza MISP Email Intake

A General

* Intervalo: 20 minutes

Servidores +

Nombre del servidor Fechas

No data

Figura 62. Conexión servidor MISP.

A continuación, se agrega un servidor, ingresando un nombre identificativo para la conexión, la URL del servidor MISP y la clave API correspondiente al usuario con permisos adecuados, obtenida previamente desde el perfil del usuario en MISP.

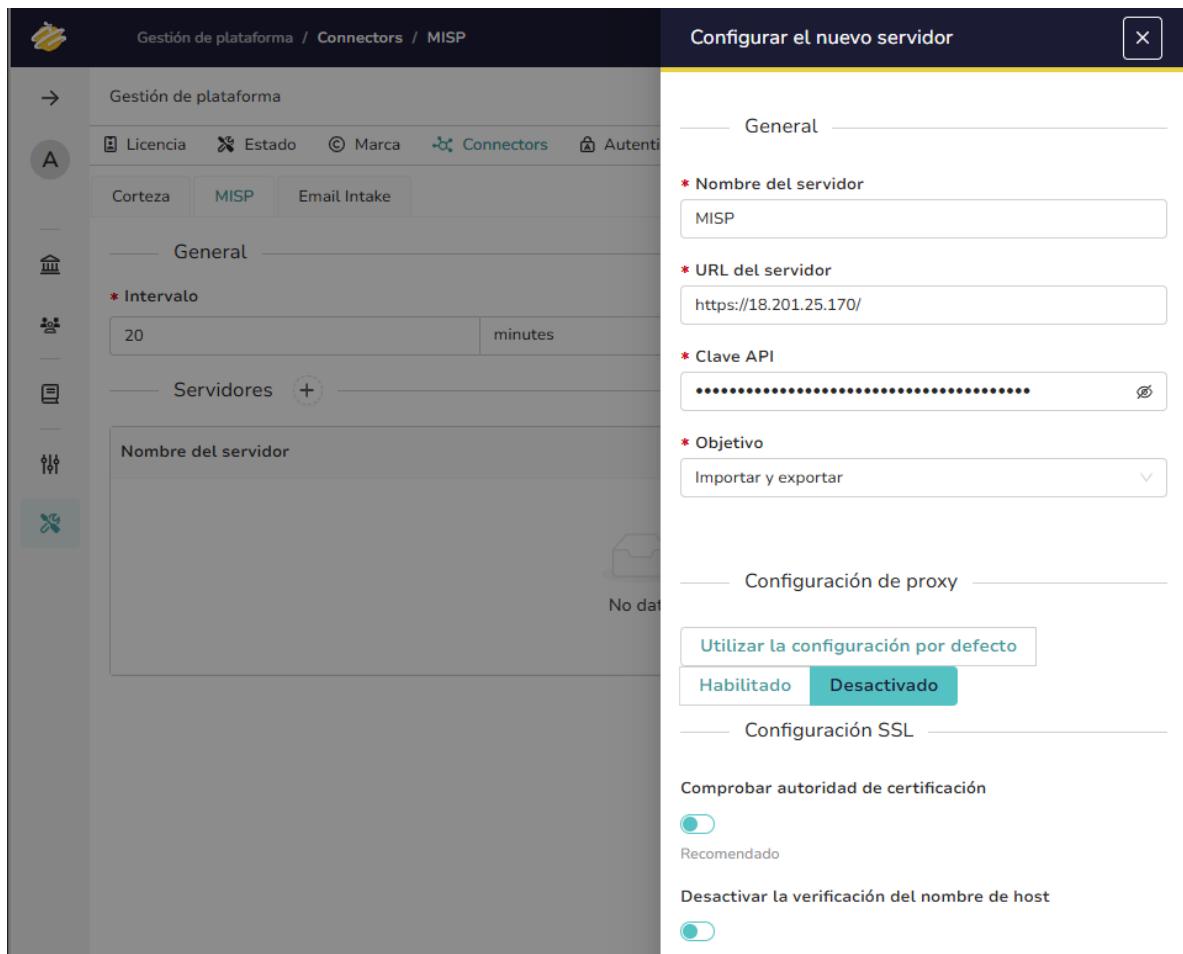


Figura 63. Conexión servidor MISP parte 2.

Es importante deshabilitar tanto el uso del proxy como de certificados SSL o la verificación del nombre del host.

3.3.2. Conexión de TheHive con Cortex

La integración entre estos componentes potencia la respuesta a incidentes de seguridad automatizando la respuesta. Para establecer esta conexión, se ha seguido un procedimiento similar a la integración con MISP.

En primer lugar, desde la interfaz web de Cortex se crea la API Key del usuario TheHive.

The screenshot shows the 'Organization: TheHive' page in TheHive. In the top right corner, there is a link to 'Back to list'. Below it, there is a search bar and a '50 / page' dropdown. The main area displays a table with columns: Status, User details, Password, and API Key. A specific row for 'thehive' is selected, showing details: Login: thehive, Full name: TheHive, Organization: TheHive, Roles: read, analyze, orgadmin. There are buttons for 'Edit password', 'Renew', 'Revoke' (which is highlighted with a red box), a copy icon, 'Edit', and 'Lock'.

Figura 64. Creación API Key Cortex.

Ya generada la API Key, se accede a la interfaz web de TheHive a la sección de Platform Management, se selecciona la pestaña Connectors y, dentro de esta, la subpestaña Corteza donde se añade un servidor como en MISP ingresando un nombre identificativo para la conexión, la URL del servidor Cortex y la clave API correspondiente al usuario, obtenida previamente.

The screenshot shows the 'Gestión de plataforma' section in TheHive. On the left, there is a sidebar with icons for 'Licencia', 'Estado', 'Marca', 'Autenticación', 'SMTP', 'Puntos finales globales', and 'Servidores LDAP'. The 'Connectors' tab is selected and highlighted with a red box. Below it, there are tabs for 'Corteza' (highlighted with a red box), 'MISP', and 'Email Intake'. Under 'Corteza', there are several configuration sections: 'General' (Max reintentos en caso de error: 3), 'Retraso de actualización' (5 seconds), 'Frecuencia de las verificaciones de estado' (1 minute), and 'Tiempo máximo que TheHive espera a que se complete el trabajo' (3 hours). At the bottom, there is a 'Servidores' section with a '+' button.

Figura 65. Conexión servidor Cortex.

Figura 66. Conexión servidor Cortex parte 2.

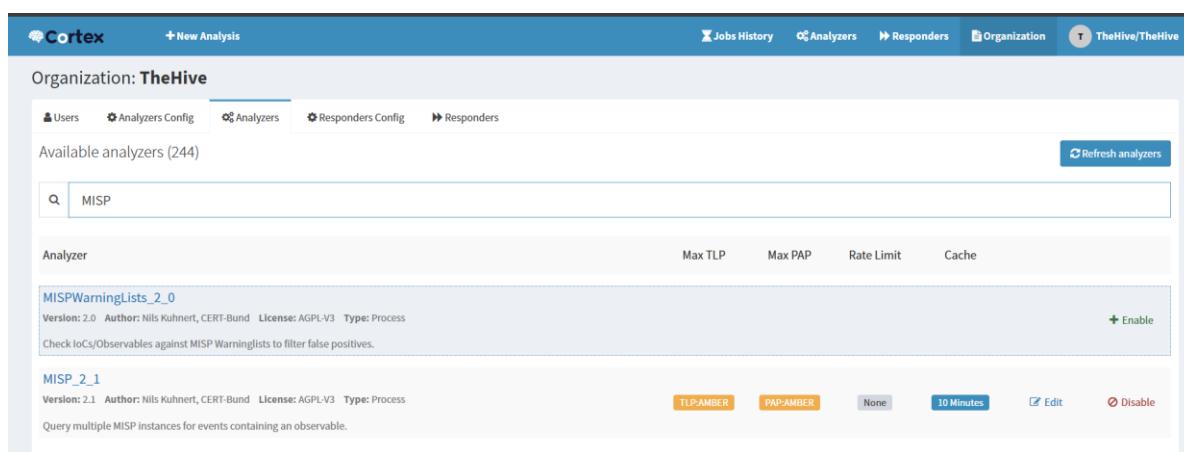
Una vez completada esta configuración, se verifica el correcto funcionamiento mediante el indicador visual que muestra el estado de conexión en la interfaz de TheHive.

Figura 67. Comprobación correcta integración de los conectores.

3.3.3. Conexión de MISP con Cortex

La integración entre MISP y Cortex se realiza mediante un enfoque que permite que TheHive analice las alertas y compare sus observables con la base de datos.

Para llevarlo a cabo, una vez ambos servicios estén operativos, se habilita el analizador correspondiente a MISP y se define su configuración en Cortex. Este archivo de parámetros contiene la dirección URL del servidor de MISP, la clave API que autoriza las consultas. La definición de estas variables permite que Cortex reconozca a MISP como fuente de inteligencia y lo invoque en cada análisis solicitado.



The screenshot shows the Cortex web interface. At the top, there's a navigation bar with links for 'Jobs History', 'Analyzers', 'Responders', 'Organization', and 'TheHive/TheHive'. Below the navigation is a section titled 'Organization: TheHive' with tabs for 'Users', 'Analyzers Config', 'Analyzers' (which is selected), 'Responders Config', and 'Responders'. A search bar at the top of this section contains the text 'MISP'. To the right of the search bar is a blue button labeled 'Refresh analyzers'. The main content area is titled 'Available analyzers (244)' and shows a list of analyzers. One item is highlighted: 'MISPAWarningLists_2_0' by Nils Kuhnert, CERT-Bund, version 2.0. It's described as a 'Process' type analyzer. Below the analyzer name, it says 'Version: 2.0 Author: Nils Kuhnert, CERT-Bund License: AGPL-V3 Type: Process'. There's also a note: 'Check IoCs/Observables against MISP Warninglists to filter false positives.' To the right of this entry is a blue '+' icon followed by the word 'Enable'. Further down the list is another entry: 'MISP_2_1' by Nils Kuhnert, CERT-Bund, version 2.1. It's also a 'Process' type analyzer. Its details are: 'Version: 2.1 Author: Nils Kuhnert, CERT-Bund License: AGPL-V3 Type: Process'. To its right are buttons for 'TLP:AMBER', 'PAP:AMBER', 'None', '10 Minutes', 'Edit', and 'Disable'.

Figura 68. Habilitación de MISP en Cortex.

Un paso imprescindible en este punto es la creación de un evento en MISP para dar significado a un indicador y convertirlo en un activo de conocimiento compartido.

En la descripción del evento se ha establecido la naturaleza de la amenaza, especificando su criticidad, el estado del análisis, la distribución de este evento y una breve información de lo que trata.

Add Event

Date: 2025-09-13 | Distribution: This community only

Threat Level: High | Analysis: Completed

Event Info: IP atacante

Extends Event: Event UUID or ID. Leave blank if not applicable.

Submit

Figura 69. Creación de un evento en MISP.

En este punto la dirección IP 192.168.2.50 pasa a incorporarse como atributo del evento. Además, se agrega un comentario claro que identifica esa dirección como perteneciente a un atacante.

Add Attribute

Category: Network activity | Type: ip-src

Distribution: Your organisation only

Value: 192.168.1.40

Contextual Comment: Esta IP pertenece a un atacante

Figura 70. Incorporación de un atributo al evento.

3.3.4. Conexión de TheHive con Wazuh

La comunicación entre estos dos servicios permite que las alertas generadas por Wazuh se transformen automáticamente en alertas dentro de TheHive. Para esta integración se ha seguido la guía oficial de Wazuh [19].

Para establecer esta integración, en primer lugar, hay que crear una API Key

al usuario por defecto de TheHive. Esta configuración inicial permite que Wazuh, a través de su integración, pueda enviar alertas directamente a TheHive mediante la API REST.



Figura 71. Creación clave API usuario TheHive.

Con el entorno de TheHive preparado, se procede con la instalación de Python3 y el módulo thehive4py en el servidor Wazuh.

```
sudo yum install python3
```

```
sudo /var/ossec/framework/python/bin/pip3 install  
thehive4py==1.8.1
```

Este módulo proporciona la interoperabilidad necesaria para enviar peticiones a TheHive desde scripts personalizados

Posteriormente, se guarda el script de integración proporcionado por la documentación oficial, encargado de gestionar la transformación de las alertas generadas por Wazuh en objetos compatibles con TheHive.

```
sudo nano /var/ossec/integrations/custom-w2thive.py
```

Además, hay que crear también el script bash que ejecutará correctamente el script creado en el paso anterior y cambiar el permiso y propiedad de ambos para garantizar que Wazuh tenga los permisos adecuados para acceder a ellos y ejecutarlos.

```
sudo chmod 755 /var/ossec/integrations/custom-w2thive.py  
sudo chmod 755 /var/ossec/integrations/custom-w2thive  
sudo chown root:wazuh /var/ossec/integrations/custom-w2thive.py  
sudo chown root:wazuh /var/ossec/integrations/custom-w2thive
```

Este script se vincula al gestor de alertas de Wazuh mediante una entrada en el archivo de configuración *ossec.conf*, lo que permite que las alertas cumplan un flujo directo hacia TheHive. Insertamos la dirección IP del servidor TheHive junto con la clave API generada anteriormente.

```
<integration>
  <name>custom-w2thive</name>
  <hook_url>http://TheHive_Server_IP:9000</hook_url>
  <api_key> API_KEY_THEHIVE </api_key>
  <alert_format>json</alert_format>
</integration>
```

Una vez configurado, se reinicia el servicio de Wazuh-manager para aplicar los cambios.

```
sudo systemctl restart wazuh-manager
```

3.3.5. Conexión de Wazuh con la estación de trabajo

Para implementar esta conexión, hay que ejecutar el agente de Wazuh en la estación para que se comunique con el servidor Wazuh enviando datos a tiempo real a través de un canal cifrado y autenticado.

Para la descarga e instalación del agente se ha seguido la guía oficial [20].

Usando la CMD como administrador para instalarlo, ejecutarlo e iniciararlo:

```
wazuh-agent-4.12.0-1.msi /q WAZUH_MANAGER="10.0.0.2"
NET START Wazuh
```

Capítulo 4

4. Resultados

4.1. Escenario 1. Ataque por fuerza bruta al puerto SSH

4.1.1. Roles y participantes:

- **Atacante:** VM Kali. Herramientas: Hydra para lanzar el ataque de fuerza bruta. Rol: *Red Team*.
- **Víctima:** VM Windows 10. Herramientas: Agente Wazuh, servicio ssh habilitado. Rol: Usuario Final.
- **SIEM:** EC2 SIEM. Herramientas: Wazuh. Rol: Monitorea, detecta la actividad sospechosa y envía la alerta al SOAR.
- **SOAR:** EC2 SOAR. Herramientas: TheHive y Cortex. Rol: Recibe la alerta, genera un caso automáticamente y ejecuta una respuesta a la alerta.

4.1.2. Objetivos y alcance

El objetivo es recrear un escenario realista en el que un atacante externo intenta acceder de manera no autorizada al sistema interno de un empleado mediante múltiples intentos fallidos de autenticación SSH, con el propósito de obtener credenciales válidas y comprometer la máquina de la víctima.

Con esta práctica se busca evidenciar cómo las capacidades del SOC permiten detectar comportamientos anómalos asociados a ataques de fuerza bruta, correlacionar los eventos de seguridad y activar un procedimiento automatizado de respuesta para la notificación del incidente.

El alcance de la simulación se limita a la fase de intrusión inicial, concretamente a los intentos reiterados de conexión al puerto SSH y la generación de la alerta correspondiente. A partir de este punto se valida la integración del SIEM, responsable de identificar el patrón de ataque, y el SOAR, que gestiona la creación de la alerta y ejecuta la acción de notificación. No se consideran fases posteriores de explotación debido a que el propósito central del estudio es evaluar la detección temprana y la capacidad de orquestación de respuesta del SOC.



Figura 72. Diagrama del ataque de fuerza bruta.



Figura 73. Diagrama de la actuación del SOC al ataque de fuerza bruta.

4.1.3. Metodología de ataque (MITRE ATT&CK)

- **Táctica**: Acceso inicial. Intento de obtener acceso a través de credenciales válidas.
- **Técnica**: T1110/T1110.001 (Brute Force: Password Guessing).
- **Cadena de ataque**:
 1. **Preparación**: Configuración de la herramienta de fuerza bruta en el atacante para apuntar al servicio SSH de la víctima.
 2. **Entrega**: Se lanzan múltiples intentos de autenticación al puerto 22 (SSH) con diferentes combinaciones de contraseña.
 3. **Explotación**: El sistema registra sucesivos fallos de autenticación en los logs.
 4. **Detección**: Wazuh genera alertas. Reglas ID: 100200, 100201 y 100201.
 5. **Orquestación**: TheHive crea un caso; Cortex lanza un *responder* que crea un ticket en el sistema de incidencias Jira.

4.1.4. Procedimiento paso a paso

4.1.4.1. Atacante

Para preparar al atacante en este escenario, se ha instalado la herramienta *Hydra*. Se trata de una herramienta para realizar ataques de fuerza bruta contra distintos servicios de autenticación de red como lo es SSH.

Para su instalación, en la terminal de Kali-Linux ejecutamos como root:

```
sudo apt install hydra -y
```

Una vez completada la instalación. Para este ataque es necesario un diccionario de contraseñas. En este caso, Kali ya tiene varios diccionarios, por lo que se ha optado por emplear el diccionario *rockyou* que tiene una colección de las contraseñas más utilizadas.

Para poder ejecutar el ataque con este diccionario, primero es necesario descomprimirlo:

```
gzip -d /usr/share/wordlists/rockyou.txt.gz
```

Con el diccionario ya disponible, y cuando el resto del laboratorio esté preparado, se lanza el ataque:

```
hydra -l testuser -P /usr/share/wordlists/rockyou.txt  
ssh://192.168.2.21
```

Esto generará muchos intentos seguidos lo cual es perfecto para que Wazuh dispare la alerta de *brute force*.

4.1.4.2. Víctima

El primer paso en la víctima es instalar y habilitar el servidor de OpenSSH desde PowerShell con privilegios de administrador. Para instarlo ejecutar:

```
Add-WindowsCapability -Online -Name OpenSSH.Server~~~~0.0.1.0
```

Una vez instalado, el servicio se inicia y configura para que arranque automáticamente:

```
Start-Service sshd  
Set-Service -Name sshd -StartupType 'Automatic'
```

Para permitir conexiones entrantes se abre el puerto 22 en el firewall:

```
New-NetFirewallRule -Name sshd -DisplayName 'OpenSSH Server (sshd)' -Enabled True -Direction Inbound -Protocol TCP -Action Allow -LocalPort 22
```

Finalmente, se crea un usuario con una contraseña débil que será el objetivo del ataque. Debemos asegurarnos de que esta contraseña esté en el diccionario que se utiliza en el ataque para que funcione.

```
net user testuser Password1 /add
```

4.1.4.3. SIEM

Se han configurado en Wazuh 3 reglas de correlación para este ataque en el archivo */var/ossec/etc/rules/local_rules.xml*. La primera para indicar el fallo de sesión con el ID 100200, la segunda con ID 100201 para informar de un posible ataque por fuerza bruta tras 5 intentos fallidos de inicio de sesión en el puerto SSH en un periodo inferior a un minuto. Por último, una alerta por inicio de sesión correcto tras los fallos con ID 100202.

Cuando el atacante ejecuta múltiples intentos de autenticación fallida al servicio SSH de la máquina víctima, dichos intentos quedan registrados en los logs del sistema. El agente de Wazuh instalado en la víctima recolecta estos eventos y los transmite al manager para su análisis.

En el manager, los registros son procesados mediante decodificadores e identifican los campos relevantes de cada intento fallido. Una vez normalizada la información, los eventos pasan por el motor de reglas de correlación de Wazuh y se activan las reglas configuradas.

El evento, ya convertido en alerta, se almacena e indexa en Elasticsearch, lo que permite su consulta y representación en el dashboard de Wazuh. Este proceso ocurre de manera automática y transparente para el usuario final.

Finalmente, con la alerta accesible en la interfaz, se pueden revisar los detalles completos del ataque. A su vez, la alerta es enviada a TheHive a través del conector configurado, donde se crea un caso para su gestión en el SOAR.

```

<!-- Detección de fuerza bruta en Windows Security (Event ID 4625) -->
<group name="windows,authentication,bruteforce,">

  <!-- 1) Fallo de inicio de sesión (4625) -->
  <rule id="100200" level="5">
    <!-- Aplica a eventos de Windows EventChannel -->
    <if_group>windows</if_group>
    <!-- Event ID 4625 -->
    <field name="win.system.providerName">Microsoft-Windows-Security-Auditing$</field>
    <field name="win.system.eventID">^4625$</field>
    <!-- Códigos típicos de fallo: usuario/contraseña incorrectos -->
    <options>no_full_log</options>
    <description>Windows: Logon failure (4625) - Unknown user or bad password.</description>
    <mitre>
      <id>T1110</id>
      <id>T1110.001</id>
    </mitre>
    <group>windows,authentication_failed,attack,</group>
  </rule>

  <!-- 2) Fuerza bruta: >=5 fallos desde la misma IP en 60s -->
  <rule id="100201" level="10" frequency="5" timeframe="60" ignore="60">
    <if_matched_sid>100200</if_matched_sid>
    <field name="win.eventdata.processName">\ssh\exe$</field>
    <!-- Y al tipo de logon usado por OpenSSH en Windows (8) -->
    <field name="win.eventdata.logonType">8$</field>
    <!-- Usa el mismo origen de red del evento de Windows -->
    <same_field>win.eventdata.targetUserName</same_field>
    <description>Windows: SSH/Logon brute force suspected (>=5 failures in 60s from same IP).</description>
    <mitre>
      <id>T1110</id>
      <id>T1110.001</id>
    </mitre>
    <group>windows,authentication_failures,attack,bruteforce,</group>
  </rule>

  <!-- 3) Éxito tras los fallos (opcional, eleva criticidad):
        si tras el patrón de fuerza bruta hay un 4624 desde la misma IP en 120s -->
  <rule id="100202" level="12" timeframe="120">
    <if_matched_sid>100201</if_matched_sid>
    <if_group>windows</if_group>
    <field name="win.system.eventID">^4624$</field>
    <field name="win.eventdata.logonType">8$</field>
    <same_field>win.eventdata.targetUserName</same_field>
    <description>Windows: Possible credential stuffing success after brute force (4624 after multiple 4625).</description>
    <mitre>
      <id>T1110</id>
      <id>T1078</id> <!-- Valid Accounts, si hay éxito -->
    </mitre>
    <group>windows,authentication_success,attack,bruteforce,</group>
  </rule>
</group>

```

Figura 74. Reglas de correlación fuerza bruta.

Security Alerts						
Time	Technique(s)	Tactic(s)	Description	Level	Rule ID	
> Sep 13, 2025 @ 03:20:49.933	T1110 T1078	Credential Access, Defense Evasion, Persistence, Privilege Escalation, Initial Access	Windows: Possible credential stuffing success after brute force (4624 after multiple 4625).	12	100202	
> Sep 13, 2025 @ 03:19:26.677	T1110 T1110.001	Credential Access	Windows: SSH/Logon brute force suspected (>=5 failures in 60s from same IP).	10	100201	
> Sep 13, 2025 @ 03:19:25.235	T1110 T1110.001	Credential Access	Windows: Logon failure (4625) - Unknown user or bad password.	5	100200	

Figura 75. Dashboard de Wazuh con alertas por ataque de fuerza bruta.

4.1.4.4. SOAR

El primer paso en el SOAR, antes de recibir la alerta, consiste en crear el *responder* que se activa en Cortex para este tipo de incidentes.

La respuesta a un ataque por fuerza bruta al puerto SSH es abrir un ticket en el sistema de incidencias Jira. Para ello se ha creado en una carpeta de */opt/Cortex-Analyzers/responders* un *responder* específico. Este

responder está formado por 3 componentes, un archivo JSON que define la configuración y metadatos del *responder* (nombre, descripción, parámetros de entrada), un archivo PY que contiene la lógica principal en Python encargada de ejecutar la creación del ticket, y un archivo *requirements.txt* donde se especifican las dependencias necesarias para que el *responder* funcione correctamente. El código fuente se puede encontrar en el anexo correspondiente.

```
ubuntu@ip-10-0-1-224:/opt/Cortex-Analyzers/responders/JiraCreateIssue$ ls
JiraCreateIssue.json  jira_create_issue.py  requirements.txt
```

Figura 76. Componentes *responder* fuerza bruta.

Estado	Gravedad	Título	# Caso	Escribe	Fuente	Referencia	Detalles	Cesionario	fechas
New	M	Windows: Possible credential stuffing success after brute force (4624 after multiple 4625).	001b06	wazuh_alert	wazuh		Observables 1 TTP 0		O. 13/09/2025 03:20 C. 13/09/2025 03:20 U. 13/09/2025 03:20
New	M	Windows: SSH/Logon brute force suspected (>5 failures in 60s from same IP).	f4539e	wazuh_alert	wazuh		Observables 1 TTP 0		O. 13/09/2025 03:19 C. 13/09/2025 03:19 U. 13/09/2025 03:19
New	M	Windows: Logon failure (4625) - Unknown user or bad password.	a87283	wazuh_alert	wazuh		Observables 1 TTP 0		O. 13/09/2025 03:19 C. 13/09/2025 03:19 U. 13/09/2025 03:19

Figura 77. Alertas creadas en TheHive por fuerza bruta.

Una vez TheHive ha recibido la alerta, el sistema puede activar de manera automática el *responder* previamente configurado. Aunque en la edición *Community* esta acción debe ejecutarse manualmente, en este escenario, el *responder* ejecuta un script que a través de la API del sistema de incidencias *Jira* crea un ticket en el tablero *TFG* para notificar a los analistas de la intrusión y que puedan tomar medidas.

Estado	Gravedad	Título	# Caso	Escribe	Fuente	Referencia	Detalles	Cesionario	fechas
New	M	Windows: Possible credential stuffing success after brute force (4624 after multiple 4625).	001b06	wazuh_alert	wazuh		Observables 1 TTP 0		O. 13/09/2025 03:20
New	M	Windows: SSH/Logon brute force suspected (>5 failures in 60s from same IP).	f4539e	wazuh_alert	wazuh		Observables 1 TTP 0		O. 13/09/2025 03:19
New	M	Windows: Logon failure (4625) - Unknown user or bad password.	a87283	wazuh_alert	wazuh		Observables 1 TTP 0		O. 13/09/2025 03:19

- Assign to me
- Iniciar
- Cerrar
- Ignorar nuevas actualizaciones
- New case from alert
- Merge alert into case
- Respondedores

Figura 78. Ejecución manual del *responder* paso 1.



Figura 79. Ejecución manual del *responder* paso 2.

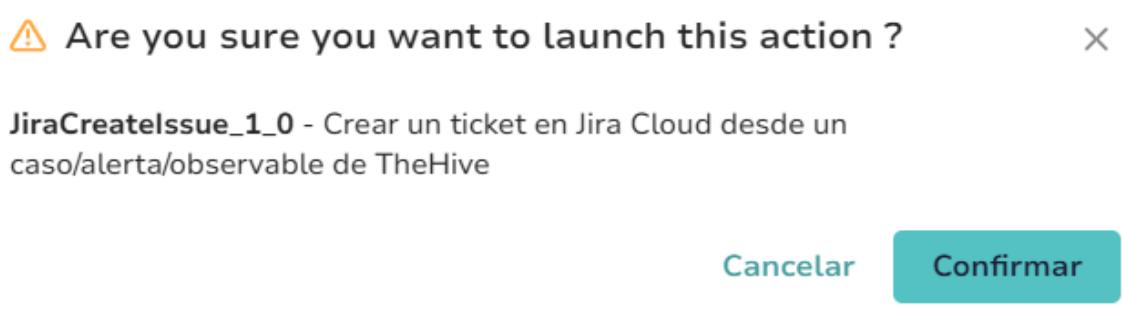


Figura 80. Ejecución manual del *responder* paso 3.

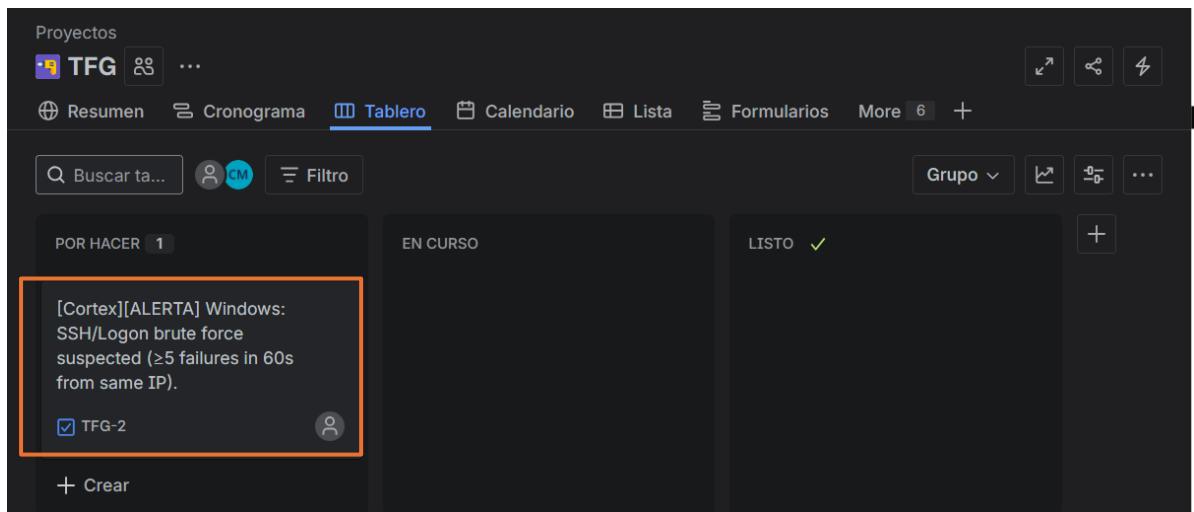


Figura 81. Ticket creado en Jira.

Por otro lado, para añadir contexto al incidente y evaluar con mayor precisión su relevancia, con el analizador MISP se pueden consultar si los observables coinciden con amenazas previamente reportadas en la base de datos.

Windows: Possible credential stuffing success after brute force (4624 after multiple 4625).

General Observables (2) TTPs (0) Attachments Casos similares Alertas similares Respondedores Historia

dataType: any(ip) X

Banderas Tipo de datos Valor / Nombre de archivo

TLP:AMBER ip 192[.]168[.]1[.]40 Ninguno No hay informes disponibles

TLP:AMBER ip 192[.]168[.]2[.]50 Ninguno No hay informes disponibles

Ejecutar analizadores

Figura 82. Ejecución manual del analizador MISP paso 1.

Analizador

Ip Analyzer Seleccionar todo / Deseleccionar todo

MISP_2_1 [Cortex]

Ejecutar analizadores seleccionados

Figura 83. Ejecución manual del analizador MISP paso 2.

TLP:AMBER ip 192[.]168[.]2[.]50 Ninguno S. 13/09/2025 04:47 C. 13/09/2025 04:47

MISP:Search="1 event(s)"

Figura 84. Coincidencia con un evento del analizador.

The screenshot shows a dark-themed user interface for TheHive. At the top, a header bar has the text "Reporte de análisis" on the left and a red "X" icon on the right. Below the header, there is a button labeled "Mostrar resultado bruto". A search bar contains the text "- 1 results". Underneath the search bar, a section titled "1 - IP atacante" is displayed. This section contains several pieces of information: "EventID: 1", "Event info: IP atacante", "UUID: 753b140c-7b53-4c37-b375-70b982146a2c", and "From: ORGNAME".

Figura 85. Reporte del análisis del observable.

Finalmente, con la respuesta aplicada y los observables analizados, la alerta queda actualizada en TheHive, proporcionando un registro detallado del ataque, la detección y la acción de mitigación realizada. Esto asegura la trazabilidad del incidente y permite medir la efectividad de la integración entre SIEM y SOAR en el proceso de defensa.

4.1.5. Resultados experimentales

Los experimentos realizados para simular un ataque de fuerza bruta al puerto SSH demostraron la efectividad del SOC en todo el ciclo de detección y respuesta. Durante la ejecución del ataque con Hydra desde la máquina atacante, la víctima registra múltiples intentos fallidos de autenticación que son recogidos por el agente de Wazuh. El manager procesa estos registros y genera una alerta clasificada como intento de fuerza bruta sobre SSH. La alerta se almacena e indexa correctamente, apareciendo en el panel de análisis con la información necesaria.

La integración con el SOAR funciona de manera fluida. La alerta emitida por Wazuh se transmite y da lugar a la creación automática de una alerta en TheHive. El caso incluye información clave, lo que permite el seguimiento del incidente. Además, su clasificación dentro de la táctica de Credential Access (MITRE ATT&CK T1110 – Brute Force) facilita la categorización inmediata de la amenaza.

En la fase de respuesta, el *responder* de Cortex se ejecuta desde la alerta de TheHive y crea automáticamente un ticket en Jira. La acción se queda registrada tanto en Cortex como en el propio caso de TheHive, garantizando la trazabilidad y la transparencia del proceso.

The screenshot shows the Cortex interface with the 'Jobs History' tab selected. There are filters for Data Types (0), Job Type (2), Analyzers (0), and Observable (Search for observable data). The results table has columns for Status, Job details, TLP, and PAP. One entry is shown: '[thehive:alert] [wazuh:f4539e] Windows: SSH/Logon brute force suspected (>5 failures in 60s from same IP).', with a 'Success' status, 'Responder: JiraCreateIssue_1_0', 'Date: a few seconds ago', 'User: TheHive/thehive', and TLP/PAP values of 'IP-AMBER'. Action buttons include View and Delete.

Figura 86. Ejecución exitosa del *responder* registrada en Cortex.

The screenshot shows the TheHive activity log. It contains three entries:

- A green info icon: 'Actualizado por Wazuh' (Updated by Wazuh) at 13/09/2025 03:34. Details: 'JiraCreateIssue_1_0 [terminado]' (Status Success), 'Windows: SSH/Logon b...'.
- A green info icon: 'Actualizado por Wazuh' (Updated by Wazuh) at 13/09/2025 03:34. Details: '[terminado]' (Status InProgress), 'Windows: SSH/Logon b...'.
- An orange plus icon: 'Creado por Wazuh' (Created by Wazuh) at 13/09/2025 03:34. Details: 'Status Started', 'CortexId Cortex', 'StartDate 13/09/2025 03:34', 'Windows: SSH/Logon b...'.
- An orange plus icon: 'Creado por Wazuh' (Created by Wazuh) at 13/09/2025 03:20.

Figura 87. Registro de la actividad en TheHive.

Los tiempos de detección y respuesta resultaron adecuados al escenario: la alerta se generó tras los primeros intentos reiterados, la alerta se generó en TheHive sin mayor retardo y la ejecución del *responder* creó de forma inmediata el ticket. Las pruebas se repitieron con resultados consistentes, confirmando la reproducibilidad del flujo. Con ello se comprobó que la combinación de Wazuh y TheHive, junto con Cortex, permite no solo detectar de manera temprana un ataque de fuerza bruta, sino también *responder* de forma efectiva.

4.2. Escenario 2. Phishing con robo de credenciales

4.2.1. Roles y participantes:

- **Atacante:** VM Kali. Herramientas: Zphisher para clonar landing, servidor en localhost, envío de correo con enlace. Rol: *Red Team*.
- **Víctima:** VM Windows 10. Herramientas: Agente Wazuh, navegador para acceso al correo electrónico. Rol: Usuario Final
- **SIEM:** EC2 SIEM. Herramientas: Wazuh. Rol: Monitorea, detecta la actividad sospechosa y envía la alerta al SOAR.

- **SOAR:** EC2 SOAR. Herramientas: TheHive y Cortex. Rol: Recibe la alerta, genera un caso automáticamente y ejecuta una respuesta a la alerta.

4.2.2. Objetivos y alcance

El objetivo es reproducir una situación realista en la que el atacante logra introducir en la red local de la empresa una página web fraudulenta que simula la página de inicio de sesión de Microsoft, con el fin de recopilar credenciales, y en la que un empleado accede a través de un correo electrónico aparentemente legítimo para cambiar la contraseña.

Con esta práctica se pretende demostrar como las herramientas del SOC permiten identificar la amenaza, correlacionar la información y activar un flujo de respuesta para la gestión del incidente.

El alcance de la simulación está delimitado a la fase inicial del ataque, concretamente cuando la víctima interactúa con la página fraudulenta y se genera la alerta de seguridad. A partir de ahí, se pone a prueba la integración entre el SIEM, encargado de detectar la actividad sospechosa, y el SOAR, que centraliza la gestión del caso y lanza la acción de respuesta. No se abordan fases posteriores del ataque ya que la intención es concentrar el ejercicio en la detección temprana y la capacidad del SOC de coordinar la respuesta.

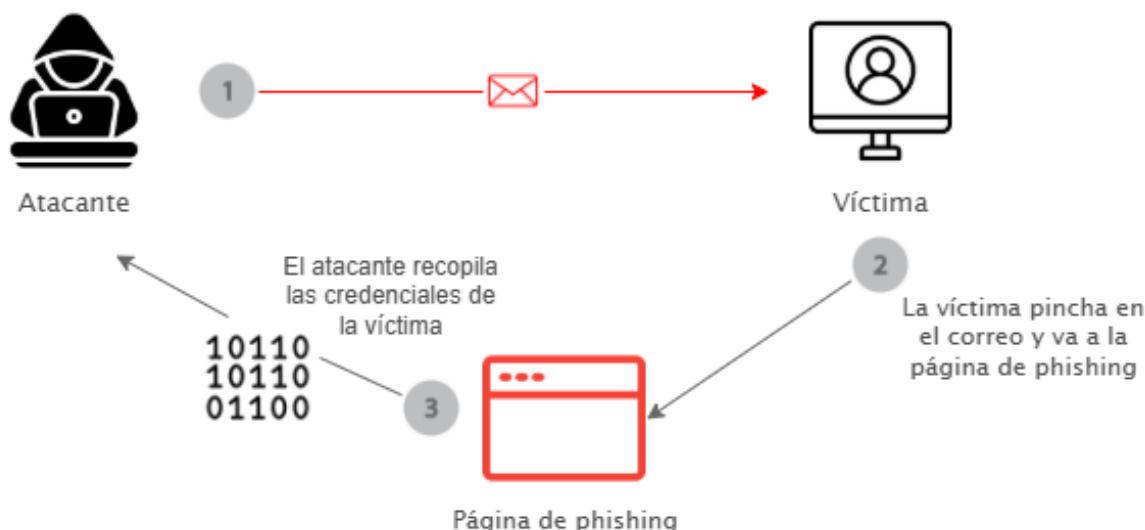


Figura 88. Diagrama del ataque de Phishing.



Figura 89. Diagrama respuesta del ataque de Phishing.

4.2.3. Metodología de ataque (MITRE ATT&CK)

- **Táctica:** Acceso inicial. Penetrar en el sistema o red de destino.
- **Técnica:** T1566 / T1566.002 (*Phishing: Link*).
- **Cadena de ataque:**
 1. **Preparación:** clonado de web legítima con *Zphisher*, hosting en la misma LAN. Preparación de plantilla HTML del correo electrónico con redirección a la página fraudulenta.
 2. **Entrega:** envío de correo electrónico creado.
 3. **Explotación:** visita del enlace y envío de credenciales.
 4. **Detección:** Wazuh genera alerta. Regla ID: 100111; Nivel: 12.
 5. **Orquestación:** TheHive crea un caso; Cortex lanza *responder* que envía correo de aviso.

4.2.4. Procedimiento paso a paso

4.2.4.1. Atacante

Para este escenario se ha instalado en la máquina atacante la herramienta *Zphisher*. Se trata de una herramienta de phishing automatizado que cuenta con plantillas prediseñadas para un propósito académico. Las instrucciones para la instalación y el uso de la herramienta se encuentran disponibles en su repositorio oficial de GitHub [21].

Para la instalación, en la terminal de Kali-Linux ejecutamos como root:

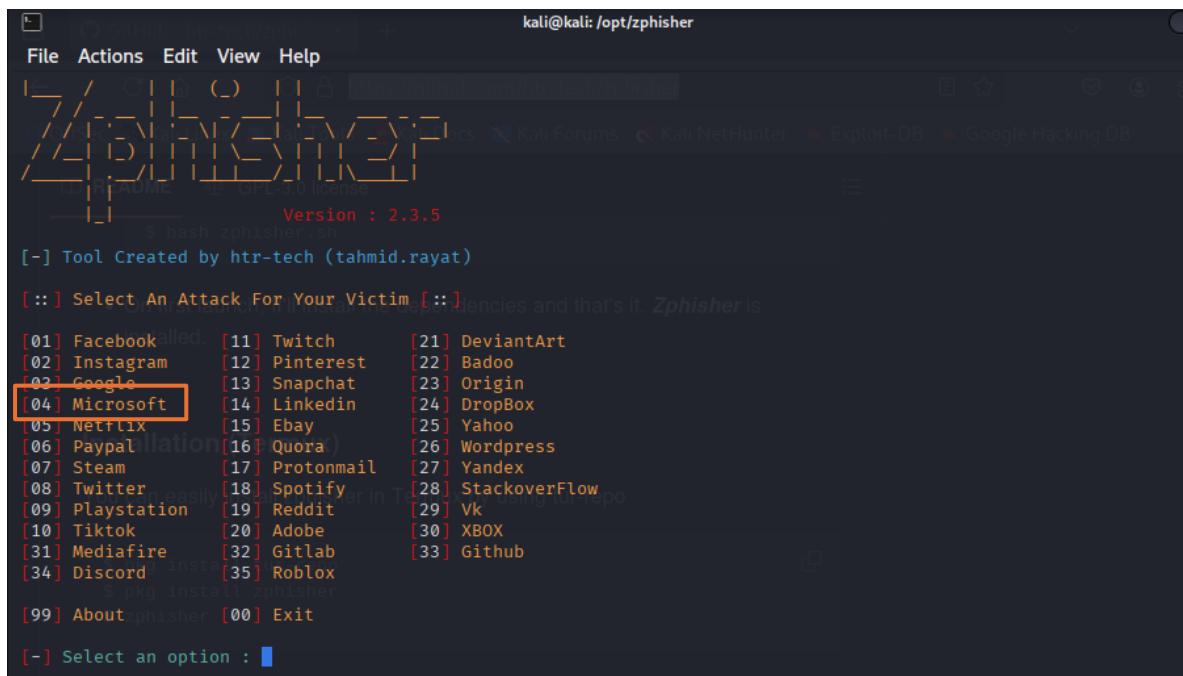
```

sudo git clone --depth=1 https://github.com/htr-tech/zphisher.git
sudo cd zphisher
sudo bash zphisher.sh

```

Una vez completada la instalación de las dependencias, se puede proceder a la creación del ataque. El procedimiento inicial consiste en la selección de la plantilla. En este caso, se ha optado por emplear la plantilla de *Microsoft*, ya que permite obtener las credenciales de correo del empleado y acceder

a información confidencial a partir de las mismas.



The screenshot shows a terminal window titled 'kali@kali: /opt/zphisher'. The menu lists various attack targets, with 'Microsoft' highlighted by a red box. The menu ends with '[99] About zphisher [00] Exit' and a prompt '[-] Select an option : █'.

```
kali@kali: /opt/zphisher
[ - ] Select an option : █
```

[::] Select An Attack For Your Victim [::]encies and that's it. *Zphisher* is

- [01] Facebook
- [02] Instagram
- [03] Google
- [04] Microsoft**
- [05] Netflix
- [06] Paypal
- [07] Steam
- [08] Twitter
- [09] Playstation
- [10] Tiktok
- [31] Mediafire
- [34] Discord
- [99] About zphisher [00] Exit

[01] Facebook [11] Twitch [21] DeviantArt
[02] Instagram [12] Pinterest [22] Badoo
[03] Google [13] Snapchat [23] Origin
[04] Microsoft [14] Linkedin [24] DropBox
[05] Netflix [15] Ebay [25] Yahoo
[06] Paypal [16] Quora [26] Wordpress
[07] Steam [17] Protonmail [27] Yandex
[08] Twitter [18] Spotify [28] StackOverflow
[09] Playstation [19] Reddit [29] Vk
[10] Tiktok [20] Adobe [30] XBOX
[31] Mediafire [32] Gitlab [33] Github
[34] Discord [35] Roblox

Figura 90. Menú de selección de ataque Zphisher.

El siguiente paso es seleccionar donde se quiere alojar el ataque. Se ha decidido alojarlo en el servidor local, ya que, Cloudflared detecta que es una plantilla de Zphisher y bloquea el envío de credenciales. Por defecto, Zphisher establece la conexión por la dirección IP 127.0.0.1; sin embargo, en este caso, la dirección IP a la que puede llegar la víctima es la 192.168.2.50, lo que requiere la apertura de un puente. Para llevarlo a cabo, es necesario relanzar el servidor para que apunte a todas las interfaces. Este método facilita el acceso a la página desde la red local compartida por la víctima y el atacante.



The screenshot shows a terminal window titled 'kali@kali: /opt/zphisher'. The menu lists hosting options, with 'localhost' highlighted by a red box. The menu ends with '[-] Select a port forwarding service : 1' and '[?] Do You Want A Custom Port [y/N]: █'.

```
kali@kali: /opt/zphisher
[ - ] Select a port forwarding service : 1
[?] Do You Want A Custom Port [y/N]: █
```

[01] Localhost [02] Cloudflared [Auto Detects] [03] LocalXpose [NEW! Max 15Min]

Figura 91. Selección de alojamiento y puerto.

```

[-] Successfully Hosted at : http://127.0.0.1:8080
[-] Waiting for Login Info, Ctrl + C to exit ... ^C
[!] Program Interrupted.

└─(kali㉿kali)-[~/opt/zphisher]
$ sudo php -S 0.0.0.0:8080 -t /opt/zphisher/.server/www

```

Figura 92. Relanzamiento del servidor.

Por otro lado, se ha desarrollado el código HTML del correo electrónico que actuará como vector de entrada del ataque. El mensaje se ha diseñado con apariencia legítima, transmitiendo cierta urgencia y un motivo aparentemente convincente que incrementa la probabilidad de que la víctima acceda al enlace fraudulento. Asimismo, se ha cuidado la elección del remitente, de manera que su nombre pueda inducir a confusión y reforzar la credibilidad del correo.

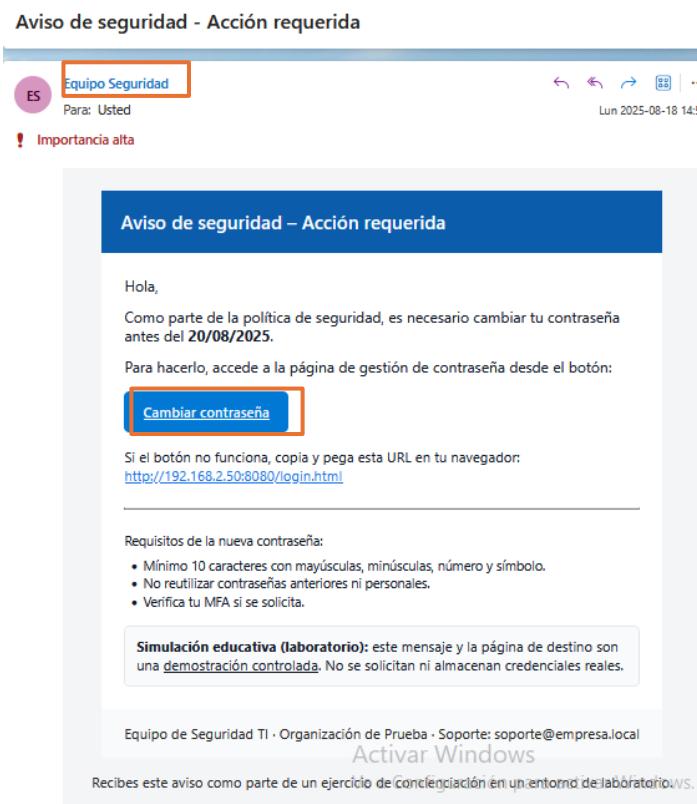


Figura 93. Apariencia final del correo desarrollado en HTML.

4.2.4.2. Víctima

La víctima debe abrir el correo, pulsar el botón e introducir las credenciales.

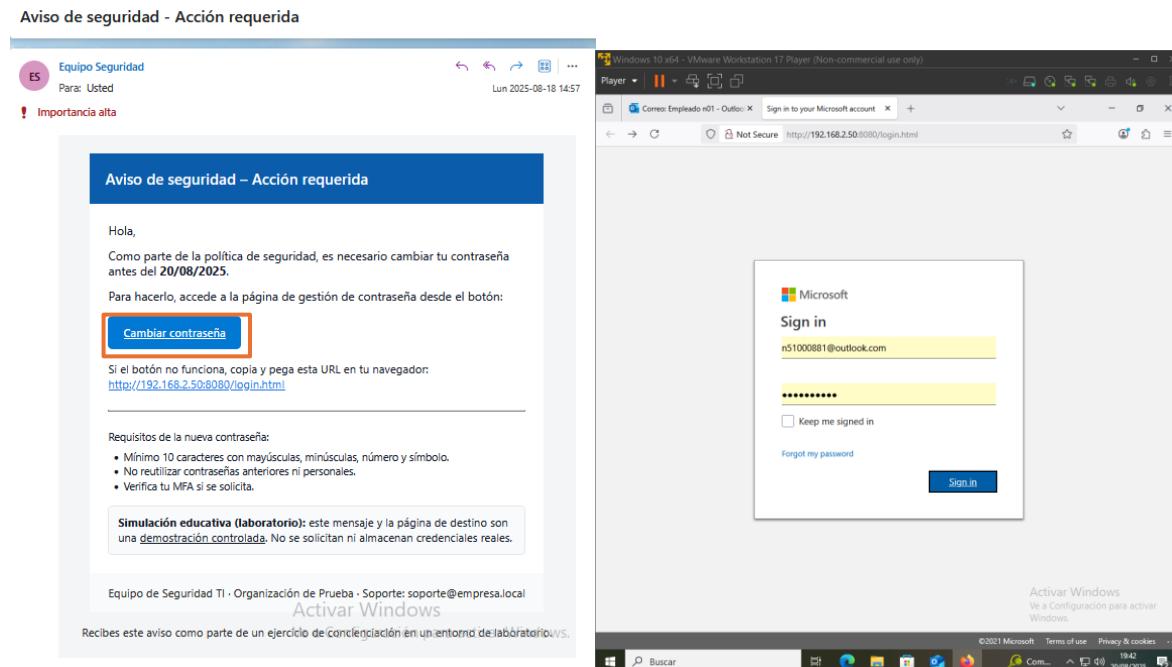


Figura 94. Pasos a seguir por la víctima.

4.2.4.3. SIEM

Como en el primer escenario, el primer paso es configurar las reglas de correlación para el ataque en el archivo *local_rules.xml*. En este caso solo se ha creado una que indica la conexión a la dirección IP 192.168.2.50 por el puerto 8080 que es donde está alojada la página web fraudulenta.

```
<!-- Detección de Phishing -->
<group name="windows,sysmon,phishing,">
    <!-- Detección: conexión a 192.168.2.50:8080 -->
    <rule id="100111" level="12">
        <if_group>sysmon_event</if_group>
        <field name="win.eventdata.DestinationIp">192.168.2.50</field>
        <field name="win.eventdata.DestinationPort">8080</field>
        <description>Phishing detected: connection to 192.168.2.50:8080</description>
        <mitre>
            <id>T1566</id>
            <id>T1566.002</id>
        </mitre>
        <options>no_full_log</options>
        <group>phishing,windows,sysmon,network,</group>
    </rule>
</group>
```

Figura 95. Regla de correlación para detección de phishing.

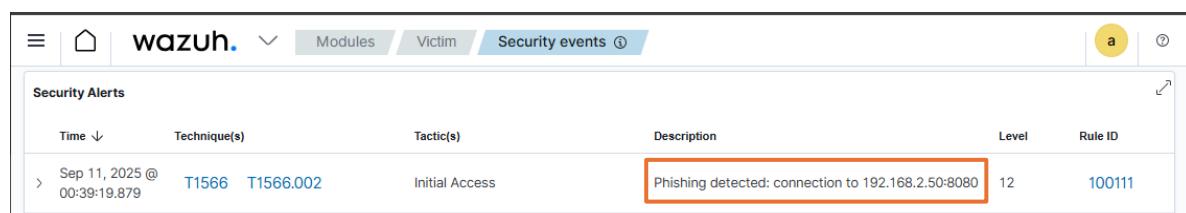
Cuando la víctima accede a la página fraudulenta, la actividad queda registrada en los logs del sistema y enviada al servidor de Wazuh.

Tras la configuración de las reglas, el siguiente paso consiste en recolectar los registros mediante el agente instalado en la máquina víctima, que se encarga de monitorizar los eventos del sistema y transmitirlos al *manager*.

Posteriormente, el manager de Wazuh procesa estos registros a través de sus decodificadores, los cuales extraen y normalizan la información relevante. Una vez normalizados los eventos, se someten las reglas de correlación y se dispara la regla configurada para este ataque.

Para visualizar la alerta en el dashboard de Wazuh, primero se almacena e indexa en Elasticsearch. Este proceso es completamente transparente al usuario.

Finalmente, con la alerta disponible en la interfaz, se pueden ver todos los detalles del evento para poder analizarla en detalle. A su vez, Wazuh genera la notificación que se envía a TheHive a través del conector habilitado para poder gestionarlo desde el SOAR.



The screenshot shows the Wazuh dashboard interface. At the top, there are navigation icons and a search bar labeled 'wazuh'. Below the search bar, there are tabs for 'Modules', 'Victim', and 'Security events'. The 'Security events' tab is active, indicated by a blue background. Under this tab, there is a table titled 'Security Alerts'. The table has columns: 'Time', 'Technique(s)', 'Tactic(s)', 'Description', 'Level', and 'Rule ID'. One row in the table is highlighted with a red border. The details for this row are: Time: Sep 11, 2025 @ 00:39:19.879; Technique(s): T1566 T1566.002; Tactic(s): Initial Access; Description: Phishing detected: connection to 192.168.2.50:8080; Level: 12; Rule ID: 100111. The 'Description' cell contains the text 'Phishing detected: connection to 192.168.2.50:8080'.

Figura 96. Dashboard de Wazuh con alerta por phishing.

4.2.4.4. SOAR

Como respuesta al phishing se ha creado un *responder* específico en una carpeta de `/opt/Cortex-Analyzers/responders`. Este *responder* está formado por 3 componentes, un archivo JSON que define la configuración y metadatos del *responder* (nombre, descripción, parámetros de entrada), un archivo PY que contiene la lógica principal en Python encargada de enviar un correo electrónico avisando del ataque, y un archivo `requirements.txt` donde se especifican las dependencias necesarias para que el *responder* funcione correctamente. El código fuente de estos archivos se puede encontrar en el anexo correspondiente.

```
ubuntu@ip-10-0-1-224:/opt/Cortex-Analyzers/responders/MailerPhishing$ ls  
requirements.txt  sendEmail.json  sendEmail.py
```

Figura 97. Componentes *responder* phishing.

Una vez la alerta llega al SOAR, el primer paso tiene lugar en TheHive, que recibe la notificación y crea de manera automática un caso asociado al evento. Este caso incluye información esencial para poder iniciar la investigación.

The screenshot shows the TheHive interface with a search bar at the top. Below it is a table of alerts. One specific alert is highlighted with a yellow border. This alert is titled "Phishing detected: connection to 192.168.2.50:8080" and has a timestamp of "New hace unos segundos". The alert details show "wazuh_alert" as the writer, "wazuh" as the source, and "rule=100111" as the reference. The observables section lists "agent_id=001", "agent_name=Victim", and "agent_ip=192.168.1.46". The status is "New" and the priority is "M".

Figura 98. Alerta creada en TheHive por phishing.

El segundo paso corresponde a la respuesta al incidente. Para ello, en TheHive se ejecuta manualmente el *responder* configurado para enviar un correo electrónico de alerta al equipo de seguridad. Esta notificación permite que se puedan tomar medidas de contención lo antes posible.

This screenshot shows the same TheHive interface as Figure 98, but with a context menu open over the highlighted alert. The menu includes options like "Assign to me", "Iniciar", "Cerrar", "Ignorar nuevas actualizaciones", "New case from alert", "Merge alert into case", and "Respondedores". The "Respondedores" option is highlighted with a red box.

Figura 99. Ejecución manual del *responder* paso 1.

This screenshot shows a modal dialog titled "Run actions on current alert". At the top, there are tabs for "All", "Responders (1)", and "Functions (0)". The "Responders" tab is selected. Below the tabs, a list shows a single responder named "SendEmail_1_0" with the description "Responder que envía un email de alerta (phishing) vía SendGrid". A checkmark is next to this responder. At the bottom right of the dialog are two buttons: "Cancelar" and "Launch action(s)".

Figura 100. Ejecución manual del *responder* paso 2.

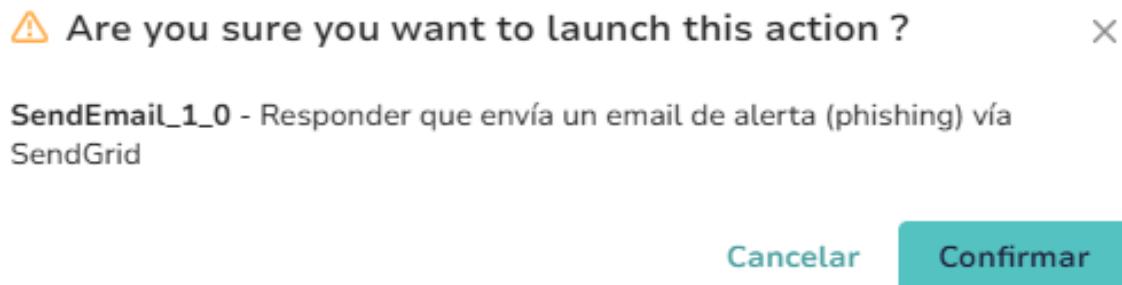


Figura 101. Ejecución manual del *responder* paso 3.

[Alerta Phishing] Caso en TheHive

celiagomezsm@gmail.com via sendgrid.net
to me

01:22 (0 minutes ago)

español inglés (americano)

Translate to English

Se ha detectado una alerta de phishing en el equipo Victim en la fecha 11-09-25--00:39:25. Ha saltado la regla 100111.
Se adjunta fichero JSON con más información.

[Message clipped] [View entire message](#)

One attachment • Scanned by Gmail

wazuh_alert_~124...

Figura 102. Correo de aviso de phishing.

Por otro lado, para más contexto del incidente y evaluar con mayor precisión su relevancia, se analizan desde MISP sus observables para posibles coincidencias con información de la base de datos.

The screenshot shows the MISP web interface. On the left, there's a sidebar with event details: ID ~41296024, created by Wazuh on 13/09/2025 04:39. It has severity set to MEDIUM and TLP set to AMBER. A dropdown menu is open on the right, containing options like 'Borrar', 'Ejecutar analizadores' (which is highlighted with a red box), 'Respondedores', and 'Copiar datos'. The main area shows a table of observables, with one row selected:

Banderas	Tipo de datos	Valor / Nombre de archivo	fecha
<input type="checkbox"/> TLP:AMBER	ip	192[.]168[.]2[.]50	S. 13/09/2025 04:39
<input type="checkbox"/> PAP:AMBER	Ninguno	No hay informes disponibles	C. 13/09/2025 04:39
<input type="checkbox"/> TLP:AMBER	ip	192[.]168[.]2[.]21	S. 13/09/2025 04:39

Figura 103. Ejecución manual del analizador MISP paso 1.

This screenshot shows a modal dialog titled 'Analizador'. At the top, it says 'Ip Analyzer' with buttons for 'Seleccionar todo' and 'Deseleccionar todo'. Below is a list box containing a single item: 'MISP_2_1 [Cortex]' with a checked checkbox. At the bottom are 'Cancelar' and 'Ejecutar analizadores seleccionados' buttons.

Figura 104. Ejecución manual del analizador MISP paso 2.

This screenshot shows the MISP event details page for event ID ~41296024. It displays an observable table with one row. The first column has an empty checkbox. The second column shows 'TLP:AMBER' and 'PAP:AMBER' status. The third column shows an 'ip' observable with value '192[.]168[.]2[.]50'. The fourth column shows dates: 'S. 13/09/2025 04:47' and 'C. 13/09/2025 04:47'. The fifth column contains a '...' button. At the bottom of the table, there's an orange button labeled 'MISP:Search="1 event(s)"'.

Figura 105. Coincidencia con un evento del analizador.

The screenshot shows a dark-themed user interface for TheHive. At the top, a black bar contains the text "Reporte de análisis" and a red "X" button. Below this, a light blue header bar says "Mostrar resultado bruto". A white search bar indicates "- 1 results". The main content area has a light gray background. It displays a single event entry with a blue header "1 - IP atacante". Inside, there are four entries: "EventID: 1", "Event info: IP atacante", "UUID: 753b140c-7b53-4c37-b375-70b982146a2c", and "From: ORGNAME".

Figura 106. Reporte del análisis del observable.

En conclusión, el caso queda documentado en TheHive con el historial de acciones realizadas, los resultados del análisis en MISP y la evidencia de la respuesta efectuada con Cortex. De este modo, el SOAR no solo centraliza la gestión del incidente, sino que también proporciona trazabilidad completa y garantiza la coordinación entre detección, análisis y respuesta.

4.2.5. Resultados experimentales

Los ensayos realizados en el laboratorio confirmaron el correcto funcionamiento del flujo de detección y respuesta ante *phishing*. Tras el acceso de la víctima a la página fraudulenta alojada en la red interna, se registró la actividad de la víctima en el agente de Wazuh y el *manager* generó una alerta de alta criticidad mapeada a MITRE ATT&CK (T1566/T1566.002). La alerta se indexó y quedó visible en el panel de análisis, con los campos clave correctamente decodificados.

La integración entre el SIEM y el SOAR respondió conforme a lo esperado. La alerta emitida por Wazuh fue transmitida a TheHive, que creó automáticamente una alerta con los observables asociados.

En la fase de respuesta, el responder de Cortex se ejecutó desde el caso de TheHive y envió un correo de aviso al buzón designado. El mensaje se entregó correctamente y contenía el mensaje programado y un archivo JSON adjunto con toda la información.

The screenshot shows the Cortex interface with the 'Jobs History' tab selected. The main title is 'Jobs History (5)'. Below it is a search bar with filters for 'Data Types (0)', 'Job Type (2)', 'Analyzers (0)', 'Observable' (containing 'Search for observable data'), and 'Status' (with dropdowns for 'Select'). There is also a 'Search' button, a 'Clear' button, and a 'Pagination' dropdown set to '50 / page'. The results table has columns for 'Status', 'Job details', 'TLP', and 'PAP'. One result is shown: '[thehive:alert] [wazuh:9583e9] Phishing detected: connection to 192.168.2.50:8080'. It includes a 'Success' status indicator, 'Responder: SendEmail_1_0', 'Date: 3 minutes ago', 'User: TheHive/thehive', and buttons for 'TLP:AMBER', 'PAP:AMBER', 'View', and 'Delete'.

Figura 107. Ejecución exitosa del *responder* registrada en Cortex.

La trazabilidad en TheHive permitió verificar la secuencia completa desde la detección hasta la notificación.

En términos cualitativos, los tiempos fueron coherentes con un entorno de laboratorio: la alerta apareció de forma prácticamente inmediata tras la interacción de la víctima, el caso en TheHive se generó sin retardo apreciable y la notificación por correo se recibió poco después de la ejecución del *responder*. Las evidencias recopiladas demuestran que la cadena de detección, orquestación y notificación funcionó extremo a extremo sin errores.

The screenshot shows the TheHive activity log. The top bar says 'Transmisión en vivo'. The first entry is for 'MISP_2_1 [terminado]' updated by Wazuh on 13/09/2025 05:05. It shows an observable of 192[.]168[.]2[.]50, a success status, and an end date of 13/09/2025 05:05. A note below says 'Phishing detected: c...'. The second entry is for the same case, updated by Wazuh on 13/09/2025 05:04, showing it is still in progress. It also shows the observable, status, end date, and the same note about phishing detection.

Figura 108. Registro de la actividad en TheHive.

Finalmente, la reproducibilidad del ejercicio quedó verificada al repetir el flujo con la misma topología y configuración, obteniéndose resultados consistentes.

Capítulo 5

5. Conclusiones y líneas futuras

5.1. Conclusiones

El presente estudio ha demostrado la viabilidad de implementar un Centro de Operaciones de Seguridad (SOC) adaptado a las necesidades y limitaciones de las PYMEs y SOHOs mediante el uso de tecnologías open-source y una infraestructura cloud escalable. A lo largo del desarrollo, se ha evidenciado que herramientas como Wazuh, TheHive, Cortex y MISP permiten cubrir el ciclo completo de detección, análisis y respuesta ante incidentes sin necesidad de grandes inversiones económicas ni de equipos altamente especializados.

Los escenarios experimentales de fuerza bruta SSH y phishing con robo de credenciales validaron con éxito la capacidad del laboratorio para detectar amenazas en fases tempranas, generar alertas correlacionadas con el marco MITRE ATT&CK y activar mecanismos de respuesta efectivos. Esto confirma que un SOC simplificado puede ser una solución realista y replicable para reforzar la seguridad en organizaciones con recursos limitados, reduciendo así la brecha existente frente a grandes corporaciones.

Asimismo, se ha comprobado que la integración entre SIEM y SOAR es clave para garantizar no solo la detección temprana, sino también la trazabilidad, orquestación y automatización de la respuesta, mejorando significativamente los tiempos de reacción y la resiliencia de la organización ante incidentes cibernéticos.

En conclusión, el trabajo aporta una solución práctica y económica que puede servir como base para que las pequeñas empresas fortalezcan su postura de ciberseguridad y aseguren la continuidad de su negocio frente al creciente panorama de amenazas digitales.

5.2. Líneas futuras

De cara a la evolución del proyecto, una primera línea de mejora se centra en la automatización avanzada de respuestas. La implantación de mecanismos como *webhooks* o *playbooks* adicionales permitiría automatizar la ejecución de los *responders* en la versión *Community* de TheHive, lo que reduciría de forma significativa los tiempos de notificación y la dependencia de acciones manuales en escenarios de alta carga operativa.

Otra posible ampliación del trabajo es la incorporación de escenarios de ataque más complejos, que incluyan pruebas con ransomware o ataques de denegación de servicio (DoS/DDoS). Esto permitiría evaluar la capacidad del SOC para responder ante incidentes de mayor impacto.

Finalmente, resulta de gran interés la integración de técnicas de *machine learning*. La aplicación de algoritmos de detección de anomalías y análisis de comportamiento complementaría el enfoque actual basado en reglas de Wazuh, mejorando la capacidad de identificar amenazas desconocidas o de tipo *zero-day*.

Bibliografía

- [1] INCIBE. (2021). *Mantenimiento remoto seguro*. Instituto Nacional de Ciberseguridad. https://assets.pubpub.org/epsxt3by/REMOTE%20MAINTENANCE-eBook-final_es-31717666519716.pdf
- [2] Kaspersky. (2024). *Ciberseguridad en las PYMEs: necesaria ante las crecientes amenazas*. SecureList. <https://securelist.lat/smb-threat-report-2024/98854/>
- [3] Verizon. (2023). *Small business cybersecurity and data breaches: What you need to know*. Verizon Business. <https://www.verizon.com/business/resources/articles/small-business-cybersecurity-and-data-breaches/>
- [4] Daemon4. (s. f.). *5 principales tipos de ciberataques a PYMEs y cómo prevenirlas*. Daemon4. <https://www.daemon4.com/empresa/noticias/5-principales-tipos-de-ciberataques-a-PYMEs-y-como-prevenirlas/>
- [5] INCIBE. (2023). *Las principales vulnerabilidades de una PYME en materia de ciberseguridad*. Instituto Nacional de Ciberseguridad. INCIBE. <https://www.incibe.es/empresas/blog/las-principales-vulnerabilidades-de-una-pyme-en-materia-de-ciberseguridad>
- [6] Sophos. (2024). *The State of Ransomware 2024*. Sophos. <https://assets.sophos.com/X24WTUEQ/at/9brgj5n44hqvgsp5f5bqcps/sophos-state-of-ransomware-2024-wp.pdf>
- [7] Hiscox. (2023). *Informe de Ciberpreparación 2023: ¿Cómo de preparadas están las PYMEs?* Hiscox España. <https://www.hiscox.es/sites/spain/files/2023-10/22594%20-%20Cyber%20Readiness%20Report%202023%20-%20Spanish.pdf>
- [8] Cambara, F. R. (2023). *Cómo instalar Wazuh 2023 en Ubuntu 22.04 + instalación de agentes*. Medium. <https://medium.com/@fransk.roman.cambara/como-instalar-wazuh-2023-en-ubuntu-22-04-instalaci%C3%B3n-de-agentes-50cae9e23b35>
- [9] StrangeBee. (s.f.). *TheHive: Step-by-step installation guide*. StrangeBee. <https://docs.strangebee.com/thehive/installation/step-by-step-installation-guide/>
- [10] Shimazz. (2023). *MISP Installation Manual*. GitHub. <https://github.com/shimazz/wazuh/blob/744c72844f02c7ea2d53cde465386fce70430b13/MSPI%20Installation.pdf>
- [11] StrangeBee. (s. f.). *Cortex installation and configuration: Step-by-step guide*. StrangeBee. <https://docs.strangebee.com/cortex/installation-and-configuration/step-by-step-guide/#cortex-installation-and-configuration>
- [12] Certifried IT. (2022). *How to install Windows Server 2019 in a VM with VMware*. Medium. <https://medium.com/certifried-it/how-to-install-windows-server-2019-in-a-vm-with-vmware-b20b80a8bcd5>
- [13] Microsoft. (s. f.). *Windows Server 2019: Soluciones modernas para entornos híbridos*. Microsoft. <https://info.microsoft.com/ww-landing-windows-server-2019.html?lcid=es-es>

- [14] Talibi, A. (2023). *Build your own lab SOC: A step-by-step guide to creating a SOC from scratch (Part 2)*. Medium. <https://medium.com/@akramtalibi1902/build-your-own-lab-soc-a-step-by-step-guide-to-creating-a-soc-from-scratch-part-2-56e17f04d055>
- [15] Irritt. (2023). Setting up a Windows 10 on VMware for your cybersecurity lab. Medium. <https://irritt.medium.com/setting-up-a-windows-10-on-vmware-for-your-cybersecurity-lab-524438f7617d>
- [16] Microsoft. (s. f.). *Descargar Windows 10*. Microsoft. <https://www.microsoft.com/es-es/software-download/windows10>
- [17] Offensive Security. (s. f.). *Get Kali: Kali virtual machines*. Kali Linux. <https://www.kali.org/get-kali/#kali-virtual-machines>
- [18] Talibi, A. (2023). *Build your own lab SOC: A step-by-step guide to creating a SOC from scratch (Part 3)*. Medium. <https://medium.com/@akramtalibi1902/build-your-own-lab-soc-a-step-by-step-guide-to-creating-a-soc-from-scratch-part-3-6698687fe37d>
- [19] Wazuh. (2023). *Using Wazuh and TheHive for threat protection and incident response*. Wazuh. <https://wazuh.com/blog/using-wazuh-and-thehive-for-threat-protection-and-incident-response>
- [20] Wazuh. (s. f.). *Windows agent package*. Wazuh. <https://documentation.wazuh.com/current/installation-guide/wazuh-agent/wazuh-agent-package-windows.html>
- [21] HTR-TECH. (s. f.). *Zphisher*. GitHub. <https://github.com/htr-tech/zphisher>

Anexos

Presupuesto

El presupuesto del proyecto recoge de manera detallada los recursos invertidos en el desarrollo del laboratorio SOC y en la investigación y redacción del documento.

Se distinguen dos componentes principales: los servicios cloud y la mano de obra técnica.

Por un lado, los gastos asociados a AWS reflejan una evolución a lo largo de los meses de pruebas que se puede observar de forma gráfica en la tabla de la figura 109. En el mes de junio el coste fue prácticamente simbólico ($\approx 0,30$ €) debido al uso casi exclusivo de la capa gratuita (*Free Tier*). Sin embargo, en julio el importe ascendió considerablemente a aproximadamente 35€ por la creación de una red privada con *VPN*, *NAT Gateway*, endpoints y *Session Manager*, lo que generó costes adicionales en infraestructura. Esta configuración llevó a rediseñar la arquitectura hacia un despliegue en red pública por su elevado impacto económico. En agosto, los costes se redujeron con la nueva arquitectura de red pública a alrededor de 4,30 €, asociados casi en su totalidad al funcionamiento de instancias EC2 t3.medium. Finalmente, en septiembre los costes fueron de aproximadamente 2€ gracias a la optimización de recursos y la finalización de pruebas. En total, el gasto directo en servicios de AWS durante todo el proyecto fue de cerca de 40€, concentrado casi en su totalidad en el mes de julio.

La mano de obra técnica representa la partida más relevante del presupuesto, puesto que el trabajo requirió una dedicación intensiva tanto en la configuración del laboratorio como en la redacción del documento. El montaje y despliegue del laboratorio implicó aproximadamente 80 horas de trabajo. A ello se suman unas 350 horas dedicadas a la investigación y redacción del documento.

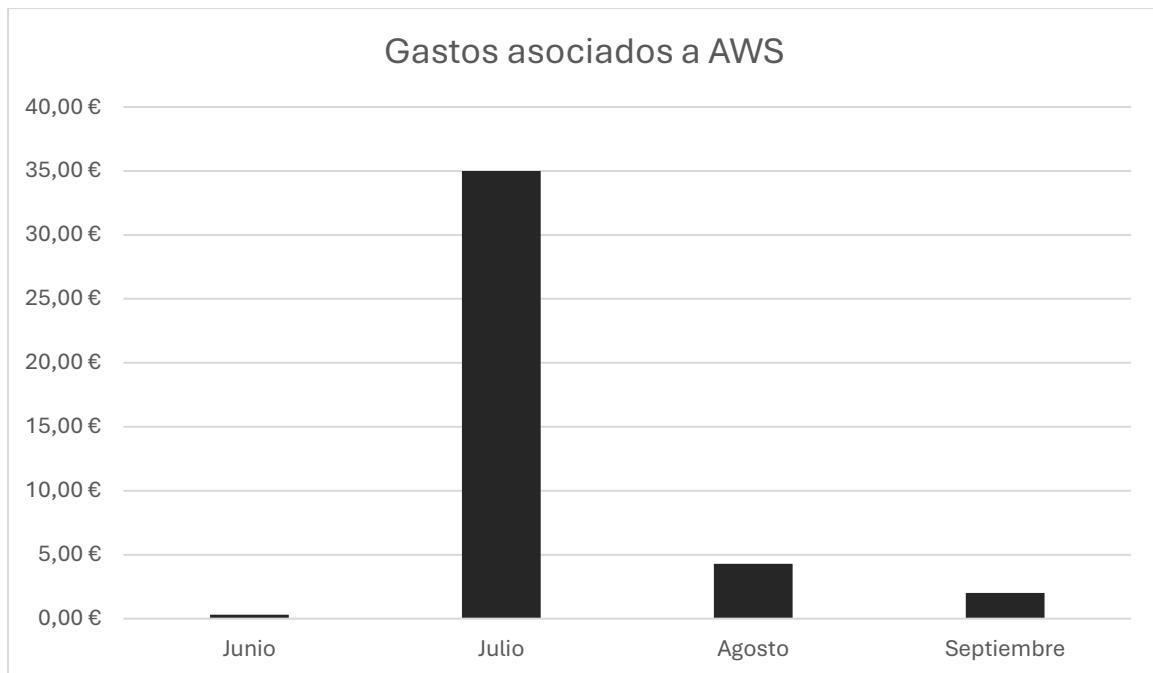


Figura 109. Evolución de los gastos asociados a AWS durante la investigación.

Concepto	Coste en horas de trabajo
Montaje y despliegue del laboratorio	80
Investigación y redacción del documento	350

Figura 110. Costes en horas de trabajo del proyecto.

AWS bill summary

Service provider	Total in invoice currency	Total in USD
Amazon Web Services EMEA SARL	USD 0.34	USD 0.34
Grand total:		USD 0.34

Figura 111. Recibo mes de junio.

AWS bill summary

Service provider	Total in invoice currency	Total in USD
Amazon Web Services EMEA SARL	USD 38.78	USD 38.78
Grand total:		USD 38.78

Figura 112. Recibo mes de julio.

	Billing period Aug 1 - Aug 31, 2025	Account ID 195275633449	Bill status Issued	Bill issued Sep 1, 2025	Date printed Sep 5, 2025
-----------------------------------------------------------------------------------	----------------------------------------	----------------------------	----------------------------------------------------------	----------------------------	-----------------------------

AWS bill summary

Service provider	Total in invoice currency	Total in USD
Amazon Web Services EMEA SARL	USD 4.74	USD 4.74
Grand total:		USD 4.74

Figura 113. Recibo mes de agosto.

	Billing period Sep 1 - Sep 30, 2025	Account ID 195275633449	Bill status Pending	Date printed Sep 13, 2025
-----------------------------------------------------------------------------------	----------------------------------------	----------------------------	----------------------------------------------------------	------------------------------

AWS estimated bill summary

Service provider	Total in USD
Amazon Web Services EMEA SARL	USD 2.01
Estimated grand total:	USD 2.01

Figura 114. Recibo mes de septiembre.

Código fuente

Responder ataque fuerza bruta

/opt/Cortex-Analyzers/responders/JiraCreateIssue/JiraCreateIssue.json

```
{  
    "name": "JiraCreateIssue",  
    "version": "1.0",  
    "author": "Celia Gómez",  
    "url": "/opt/Cortex-Analyzers/responders/",  
    "license": "MIT",  
    "description": "Crear un ticket en Jira Cloud desde una alerta de TheHive",  
    "dataTypeList": ["thehive:case", "thehive:alert", "thehive:case_artifact"],  
    "command": "JiraCreateIssue/jira_create_issue.py",  
    "baseConfig": "JiraCreateIssue",  
    "configurationItems": [  
        {  
            "name": "jira_url",  
            "description": "Base URL de Jira Cloud (https://tfgsoc2025.atlassian.net)",  
            "type": "string", "multi": false, "required": true  
        },  
        {  
            "name": "email",  
            "description": "Email de tu cuenta Atlassian",  
            "type": "string", "multi": false, "required": true  
        },  
        {  
            "name": "api_token",  
            "description": "API token creado en id.atlassian.com",  
            "type": "string", "multi": false, "required": true  
        },  
        {  
            "name": "project_key",  
            "description": "Project Key de Jira",  
            "type": "string", "multi": false, "required": true  
        },  
        {  
            "name": "label",  
            "description": "Label a aplicar",  
            "type": "string", "multi": false, "required": false, "defaultValue": "from-cortex"  
        },  
        {  
            "name": "check_tlp",  
            "description": "Enforce TLP máximo",  
            "type": "boolean", "multi": false, "required": false, "defaultValue": false  
        },  
        {  
            "name": "max_tlp",  
            "description": "TLP máximo aceptado",  
            "type": "number", "multi": false, "required": false, "defaultValue": 2  
        }  
    ]  
}
```

/opt/Cortex-Analyzers/responders/JiraCreateIssue/requirements.txt

```
cortexutils>=2.0.0  
requests>=2.31.0
```

/opt/Cortex-Analyzers/responders/JiraCreateIssue/jira_create_issue.py

```
#!/usr/bin/env python3

import json
import requests
from requests.auth import HTTPBasicAuth
from cortexutils.responder import Responder

# Responder que crea un issue en Jira
class JiraCreateIssue(Responder):
    def __init__(self):
        super().__init__()

        # Se configuran en Cortex los parámetros básicos
        self.jira_url = self.get_param('config.jira_url', None, 'Falta
config.jira_url')
        self.email = self.get_param('config.email', None, 'Falta config.email')
        self.api_token = self.get_param('config.api_token', None, 'Falta
config.api_token')
        self.project_key = self.get_param('config.project_key', 'TFG', None) # Proyecto
destino
        self.label = self.get_param('config.label', 'from-cortex', None) # Etiqueta
opcional

    # Autenticación para la API de Jira (email + token)
    def _auth(self):
        return HTTPBasicAuth(self.email, self.api_token)

    # Cabeceras para enviar JSON
    def _headers(self):
        return {"Accept": "application/json", "Content-Type": "application/json"}

    # Construye un título y descripción simples a partir de los datos
    def build_text(self, data):
        dt = self.get_param('dataType', '')
        title = 'Cortex → Jira: nuevo evento'
        desc_lines = []

        dt == 'thehive:alert':
            title = f"[Cortex][ALERTA] {data.get('title') or data.get('id')}"
            desc_lines.append(f"Fuente: {data.get('source')}")
            desc_lines.append(f"Severidad: {data.get('severity')}")
            desc_lines.append(f"Descripción:\n{data.get('description') or ''}")

        return title[:254], "\n".join([l for l in desc_lines if l is not None])

    # Convierte el texto a formato ADF requerido por Jira Cloud
    def adf_description(self, text):
        return {
            "type": "doc",
            "version": 1,
            "content": [
                {
                    "type": "paragraph",
                    "content": [{"type": "text", "text": text}]
                }
            ]
        }

    def run(self):
        super().run()

        # Datos recibidos de Cortex
        data = self.get_param('data', None, None)
        if not isinstance(data, dict):
            self.error('La entrada debe ser un objeto JSON')
```

```

# Título y descripción
summary, desc_text = self.build_text(data)

# Cuerpo del issue que se mandará a Jira
payload = {
    "fields": {
        "project": {"key": self.project_key},
        "summary": summary,
        "issuetype": {"name": "Task"}, # siempre crea un Task
        "description": self.adf_description(desc_text),
        "labels": [self.label] if self.label else []
    }
}

# URL del endpoint de creación de issues en Jira
url = f"{self.jira_url.rstrip('/')}/rest/api/3/issue"

try:
    resp = requests.post(url, json=payload,
                          headers=self._headers(),
                          auth=self._auth(),
                          timeout=20)
except requests.RequestException as e:
    self.error(f"Error de red al llamar a Jira: {e}")

if 200 <= resp.status_code < 300:
    body = resp.json()
    key = body.get("key")
    link = f"{self.jira_url.rstrip('/')}/browse/{key}"
    self.report({"message": f"Issue creado en Jira: {key}", "key": key, "url": link, "id": body.get("id")})
else:
    try:
        err = resp.json()
    except Exception:
        err = {"error": resp.text}
    self.error(f"Error {resp.status_code} de Jira: {json.dumps(err, ensure_ascii=False)}")

if __name__ == '__main__':
    JiraCreateIssue().run()

```

Responder phishing

/opt/Cortex-Analyzers/responders/MailerPhishing/sendEmail.json

```
{  
    "name": "SendEmail",  
    "version": "1.0",  
    "author": "Celia",  
    "url": "/opt/Cortex-Analyzers/responders",  
    "license": "AGPL-V3",  
    "description": "Responder que envía un email de alerta (phishing) vía SendGrid",  
    "dataTypeList": ["thehive:alert", "thehive:case"],  
    "command": "MailerPhishing/sendEmail.py",  
    "baseConfig": "MailerPhishing",  
    "configurationItems": [  
        {  
            "name": "api_key",  
            "description": "API Key de SendGrid",  
            "type": "string",  
            "multi": false,  
            "required": true  
        },  
        {  
            "name": "from",  
            "description": "Remitente (email verificado en SendGrid)",  
            "type": "string",  
            "multi": false,  
            "required": true  
        },  
        {  
            "name": "to",  
            "description": "Destino del aviso",  
            "type": "string",  
            "multi": false,  
            "required": true,  
            "defaultValue": "celiagomezsm@gmail.com"  
        }  
    ]  
}
```

/opt/Cortex-Analyzers/responders/MailerPhishing/requirements.txt

```
cortexutils  
sendgrid>=6,<7
```

/opt/Cortex-Analyzers/responders/MailerPhishing/sendEmail.py

```
#!/usr/bin/env python3  
import sys  
import os  
import json  
import base64  
from datetime import datetime  
  
from sendgrid import SendGridAPIClient  
from sendgrid.helpers.mail import (  
    Mail, Attachment, FileContent, FileName, FileType, Disposition  
)  
  
def read_input():  
    return json.load(sys.stdin)
```

```

def fmt_date_ms(ms):
    """Convierte al formato dd-mm-yy--HH:MM:SS (UTC)."""
    if not ms:
        return "desconocida"
    dt = datetime.utcfromtimestamp(ms / 1000.0)
    return dt.strftime("%d-%m-%y--%H:%M:%S")

def parse_tags(tags_list):
    """Parseado de los tags"""
    res = {}
    for t in tags_list or []:
        if "=" in t:
            k, v = t.split("=", 1)
            res[k.strip()] = v.strip()
    return res

def build_body(data):
    """Genera el cuerpo del correo con agent_name, regla y fecha."""
    tags = parse_tags(data.get("tags", []))
    agent_name = tags.get("agent_name", "desconocido")
    rule = tags.get("rule", "N/A")
    fecha = fmt_date_ms(data.get("date"))
    return (
        f"Se ha detectado una alerta de phishing en el equipo {agent_name} "
        f"en la fecha {fecha}. Ha saltado la regla {rule}.\n"
        "Se adjunta fichero JSON con más información."
    )

def make_json_attachment(payload, filename="wazuh_alert.json"):
    """Crea el adjunto JSON en base64 para SendGrid."""
    raw = json.dumps(payload, ensure_ascii=False, indent=2).encode("utf-8")
    b64 = base64.b64encode(raw).decode("ascii")
    attachment = Attachment()
    attachment.file_content = FileContent(b64)
    attachment.file_type = FileType("application/json")
    attachment.file_name = FileName(filename)
    attachment.disposition = D

```



Universidad
de Alcalá

2025