SOC Doméstico

Juan Marvin Fernández García
Grado de Ingeniería de Tecnologías y Servicios de Telecomunicación
Administración de redes y sistemas operativos

Mario Prieto Vega
David Bañeres Besora / Montse Serra Vizern

06/2023

Licencias alternativas (elegir alguna de las siguientes y sustituir la de la página anterior)

A) Creative Commons:

Esta obra está sujeta a una licencia de Reconocimiento-NoComercial-SinObraDerivada 3.0 España de Creative Commons

Esta obra está sujeta a una licencia de Reconocimiento-NoComercial-CompartirIgual 3.0 España de Creative Commons

Esta obra está sujeta a una licencia de Reconocimiento-NoComercial 3.0 España de Creative Commons

Esta obra está sujeta a una licencia de Reconocimiento-SinObraDerivada 3.0 España de Creative Commons

Esta obra está sujeta a una licencia de Reconocimiento-CompartirIgual 3.0 España de Creative Commons

Esta obra está sujeta a una licencia de Reconocimiento 3.0 España de Creative Commons

B) GNU Free Documentation License (GNU FDL)

Copyright © AÑO TU-NOMBRE.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free

C) Copyright

FICHA DEL TRABAJO FINAL

Título del trabajo:	*SOC Doméstico*
Nombre del autor:	*Juan Marvin Fernández García*
Nombre del consultor/a:	*Mario Prieto Vega*
Nombre del PRA:	*David Bañeres Besora / Montse Serra Vizern*
Fecha de entrega (mm/aaaa):	06/2023
Titulación::	*Grado de Ingeniería de Tecnologías y Servicios de Telecomunicación*
Área del Trabajo Final:	*Administración de redes y sistemas operativos*
Idioma del trabajo:	*Castellano*
Palabras clave	*SOC, open-source*

Resumen del Trabajo (máximo 250 palabras): *Con la finalidad, contexto de aplicación, metodología, resultados y conclusiones del trabajo.*

La creciente presencia de dispositivos conectados en los hogares y en el entorno laboral plantea desafíos significativos en términos de seguridad y protección de datos. La falta de seguridad en las redes domésticas expone la información sensible a posibles ataques externos.

Los firewalls actúan como una primera línea de defensa, pero pueden ser insuficientes ante amenazas complejas. Los Centros de Operaciones de Seguridad (SOC) permiten supervisar y gestionar la seguridad de la infraestructura, pero su adquisición suele ser costosa y orientada a grandes empresas. Sin embargo, el uso de herramientas de código abierto puede ofrecer funcionalidades personalizadas, permitiendo una implementación más accesible y adaptable.

Como propuesta, este proyecto, se ha buscado ofrecer una solución adaptada a los recursos y limitaciones de una red doméstica para proteger la integridad, confidencialidad y disponibilidad de los dispositivos y la información generada en los hogares.

La arquitectura SOC propuesta ha sido diseñada para su uso en un entorno doméstico, donde la arquitectura de este consistente de 4 módulos: Detección de intrusiones, Gestión de alertas, Almacenamiento y Análisis. Las

herramientas empleadas en el proyecto se limitan a software *open-source* gratuito.

El SOC propuesto ha demostrado tener un gran potencial para su implementación en entornos domésticos. Del mismo modo, la automatización de los procesos permite que el usuario pueda hacer uso del sistema sin la necesidad de contar con profundos conocimientos de ciberseguridad.

Abstract (in English, 250 words or less):

The increasing presence of connected devices in homes and work environments poses significant security and data protection challenges. The lack of security in home networks exposes sensitive information to potential external attacks.

Firewalls act as a first line of defence but may be insufficient against complex threats. Security Operations Centres (SOC) allow for the monitoring and managing of infrastructure security, but their acquisition is often costly and targeted toward large companies. However, the use of open-source tools can offer customized functionalities, enabling a more accessible and adaptable implementation.

As a proposed solution, this project aims to provide a solution tailored to the resources and limitations of a home network to protect the integrity, confidentiality, and availability of devices and generated information in households.

The proposed SOC architecture has been designed for use in a home environment, consisting of four modules: Intrusion Detection, Alert Management, Storage, and Analysis. The tools used in the project are limited to free open-source software.

The proposed SOC has demonstrated outstanding potential for implementation in home environments. Additionally, process automation allows users to utilize the system without the need for extensive knowledge of cybersecurity.

Índice

Lista de figuras

Lista de tablas

1 Introducción

1.1 Contexto y justificación del Trabajo

La tecnología se ha incorporado como parte imprescindible en el desarrollo de las tareas laborales, donde hasta las pequeñas empresas cuentan con diversos dispositivos como base de las operaciones del negocio. Del mismo modo, el uso de la tecnología dentro de los hogares ha incrementado drásticamente.

Es común ver que cada uno de los integrantes cuente con más de un dispositivo personal conectado a la red. A estos se suman los asistentes virtuales, televisores, altavoces e incluso juguetes infantiles, pero también se encuentran con mayor frecuencia los dispositivos domóticos. Es decir, aquellos dispositivos del hogar que se conectan automáticamente entre sí a través de una red concreta sin intervención humana para lograr una automatización de la vivienda inteligente (Xataka, 2022).

La conectividad de los dispositivos hace que los datos que recogen se registren y se manden a la red en la que gestionan por sí mismos la energía, la seguridad y la comunicación del hogar. Los dispositivos de esta red recopilan la información necesaria, la procesa y ordena en función de la petición del usuario.

La conectividad de los dispositivos en muchos casos está conectada a la red wifi del hogar, por el cual a su vez accede a internet. Donde en la mayoría de los casos no existe una codificación lo suficientemente segura. Esto hace que el acceso a la red interna sea un proceso sencillo, quedando más expuesta frente a agresores externos.

El tipo de información sensible que tratan y la enorme cantidad de datos que se almacenan en estos dispositivos, unidad a la escasa seguridad en la conexión a la red interna, provoca una situación de riesgo en la privacidad y la protección de los datos, donde los datos privados siendo susceptible de quedar en control de terceros.

En primer momento se recurre al cortafuegos o *firewall.* Este es un elemento que filtrar el tráfico de la red basándose en unas reglas, permitiendo aquellas conexiones previamente autorizadas. Los cortafuegos pueden ser un elemento de *software*, como el que incorporan la mayoría de sistemas operativos, pero también pueden ser un dispositivo *hardware* específico (Learning, 2022).

La principal problemática de los *firewall*s tradicionales es que solo son capaces de proteger de los troyanos básicos, sin embargo, la complejidad de las amenazas ha evolucionado, y el uso exclusivo de este modelo puede ser insuficiente para otros tipos de amenazas. Es por ello, por lo que las organizaciones suelen aplicar elementos de seguridad más elaborados (INCIBE, 2020).

Debido a la gran variedad de elementos y requisitos para la ciberseguridad, el equipo responsable de garantizar la seguridad de la información se apoya en una entidad llamada Centro de Operaciones de Seguridad, en sus siglas en inglés SOC. El SOC es una plataforma que permite la supervisión y administración de la seguridad de la infraestructura IT. Al detectar una anomalía, el SOC, escala y determina la naturaleza de la amenaza para poder resolverla con la mayor celeridad posible (IMF, 2020; ORACLE, 2023).

La adquisición de un SOC es compleja y costosa, haciendo que por lo general este oriento a grandes empresas, dado que la mejora de la detección de incidentes de seguridad por medio de la supervisión ininterrumpida de la actividad de los datos de la red tiene grandes costos asociados tanto de equipo como de personal. Esto hace que su adquisición sea inasumible por particulares o pequeñas empresas (Crowdstrike, 2022).

Se han creado herramienta con buenas prestaciones que apoyan a la solución de mejorar y personalizar los distintos mecanismos de una arquitectura SOC. Del mismo modo, el uso de código libre permitirá reducir drásticamente los costes de implementación y mantenimiento, dado que se desarrolla de manera descentralizada y colaborativa, las propias comunidades son las encargadas de su desarrollo y no un solo autor o empresa (Red Hat, 2023).

De este modo, utilizando modelos de desarrollo de *software open-source*, se diseñará una arquitectura de ciberseguridad adaptando la estructura y complejidad de un SOC a las necesidades y limitaciones de una pequeña empresa o red doméstica. Otorgando de este modo la posibilidad de controlar el acceso y seguridad de su red sin gran inversión de recursos (Park, 2022).

1.2 Objetivos del Trabajo

"Adaptar el SOC a los recursos y dimensiones de una red doméstica."

Dentro del alcance del proyecto y de los objetivos conseguidos podemos destacar:

- Análisis del funcionamiento de un SOC, así como las diferentes estructuras que la componen.

- Análisis de herramientas *open-source*.

- Definición de un entorno integrado de SOC doméstico basado en herramientas *open-source*.

- Diseño de la arquitectura SOC más favorable para uso doméstico.

1.3 Enfoque y método seguido

En este trabajo se adaptó la estructura SOC a un entorno doméstico, para ello se afrontó por medio de una arquitectura modular:

- Módulo de Gestión
- Módulo de Almacenamiento
- Módulo de Análisis
- Módulo de Detección

Valorando que un SOC muestra una estructura compleja y amplia en la que interviene diversas áreas, la elección de una arquitectura modular es la estrategia apropiada que permite reducir la complejidad gestionando los módulos por separado, al mismo tiempo que facilita el diseño personalizado y agiliza su implantación.

1.4 Planificación del Trabajo

Este apartado tiene la intención de describir la planificación del proyecto, en el que se detallara la distribución temporal de cada uno de los hitos del proyecto, junto a una breve descripción. Al final del apartado, mediante un diagrama Gantt, se mostrará la distribución temporal y la interdependencia de las tareas.

Proyecto: SOC Doméstico	**Fecha Inicio**: 03/03/2023
Hito 1: Estudio de la tecnología SOC	**Fecha Fin**: 19/03/2023
Descripción: Análisis de las distintas tecnologías y la viabilidad de interacción entre ellas para la implementación de SOC.	**Task1.1**: Identificación de las principales funciones de un SOC. **Task1.2**: Análisis de las distintas tecnologías.

Tabla 1: Hito 1 - Estudio de la tecnología SOC

Proyecto: SOC Doméstico	**Fecha Inicio**: 20/03/2023
Hito 2: Implementación del entorno de pruebas	**Fecha Fin**: 02/04/2023

Descripción: Se llevará a cabo todas las tareas relacionadas con la creación del entorno necesario para el desarrollo del proyecto.	**Task2.1**: Instalación de la VM. **Task2.2**: Creación del entorno Cloud en AWS.

Tabla 2: Hito 2: Implementación del entorno de pruebas

Proyecto: SOC Doméstico	**Fecha Inicio**: 03/04/2023
Hito 3: Creación del SOC doméstico	**Fecha Fin**: 21/05/2023
Descripción: Realización de la implementación de los diferentes módulos de la arquitectura SOC.	**Task3.1**: Implementación del módulo de gestión. **Task3.2**: Implementación del módulo de Análisis. **Task3.3**: Implementación del módulo de Detección. **Task3.4**: Implementación del módulo de Almacenamiento.

Tabla 3: Hito 3: Creación del SOC domestico

Proyecto: SOC Doméstico	**Fecha Inicio**: 22/05/2023
Hito 4: Evaluación de los resultados	**Fecha Fin**: 31/05/2023
Descripción: Se analizarán los resultados obtenidos, se hará retrospectiva de lo aprendido y se describirá los posibles casos de aplicación y propuestas de futuro a abordar.	**Task4.1**: Retrospectiva y evaluación de los resultados. **Task4.2**: Lecciones aprendidas y propuestas de futuro.

Tabla 4: Hito 4: Evaluación de los resultados

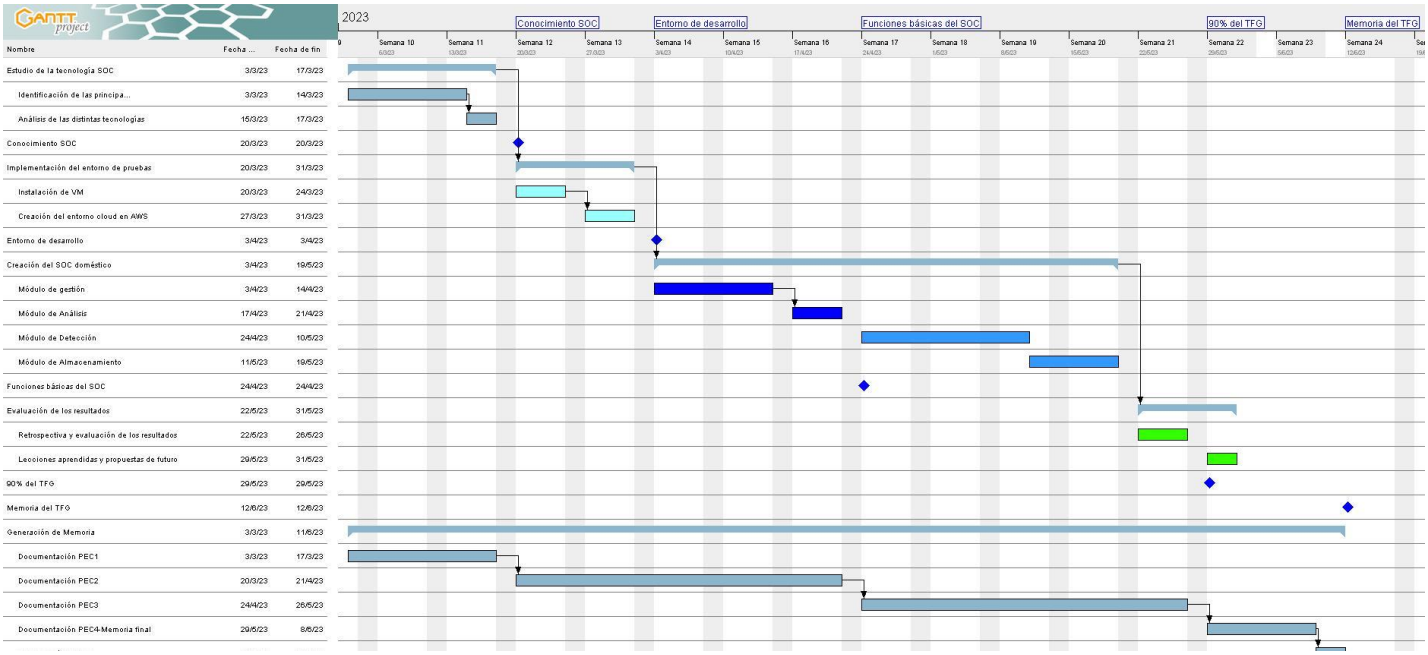

Ilustración 1: Diagrama de Gantt planificación de proyecto.

Código	Descripción	Causa	Probabilidad	Impacto
R01	Fallos derivados de la configuración de los parámetros de gestión de TheHive	Fallo de Configuración	Media	Alto
R02	Fallos derivados de la configuración de los parámetros de gestión de Cortex	Fallo de Configuración	Baja	Alto
R03	Fallos de la compatibilidad entre los diferentes programas	Fallo de Compatibilidad	Media	Medio
R04	Fallos derivados de la configuración del Módulo de Detección	Fallo de Configuración	Media	Medio
R05	Dificultades en la resolución de ciertos hitos por inexperiencia sobre las tecnologías trabajadas	Falta de Experiencia	Media	Alto

Tabla 5: Riesgos

Código	Acción	Tipo	Riesgo post Mitigación
A1R01	Se consultará con la comunidad de la tecnología en referencia	Correctora	Bajo
A1R02	Se consultará con la comunidad de la tecnología en referencia	Correctora	Bajo
A1R03	Se buscarán programas alternativos	Mitigadora	Bajo

A1R04	Se desestimará la implantación del módulo que menos afecte a la entrega del producto mínimo viable	Mitigadora	Medio
A1R05	Se usarán métodos de autoformación y se consultará con la comunidad de la tecnología en referencia	Correctora	Medio

Tabla 6: Plan de contingencia

1.5 Breve sumario de productos obtenidos

Los hitos alcanzados a la finalización del proyecto son:

- Conocimiento de la estructura SOC.
- Conocimiento de las distintas herramientas SOC *open-source*
- Diseño de un entorno SOC integrado en una red doméstica.
- Implementación de los diferentes módulos de la arquitectura SOC.
    - Módulo de Gestión
    - Módulo de Almacenamiento
    - Módulo de Análisis
    - Módulo de Detección
- Análisis y descripción de los posibles casos de uso de la herramienta.

1.6 Breve descripción de los otros capítulos de la memoria

Descripción de elementos de un SOC:
Este capítulo introducirá los conocimientos de los distintos elementos de un SOC los cuales serán útiles para la compresión de los capítulos posteriores con mayor facilidad. Del mismo modo se explicarán las herramientas *open-source* que existen en el mercado.

Diseño y arquitectura propuesta:
Este capítulo se hará la propuesta de la arquitectura SOC para implementarse en un entorno doméstico haciendo uso exclusivo de herramientas *open-source*.

Configuración de las herramientas:
Este capítulo alberga los detalles de la instalación, configuración e integración de las distintas herramientas necesarias que permiten al lector replicar el proyecto.

Resultados:

Este capítulo aborda el flujo de trabajo y el empleo de las herramientas de la arquitectura implementada.

Discusión:
En este capítulo se realiza un análisis crítico de los resultados obtenidos.

Conclusiones:
En este capítulo se tratará el grado de madurez y aprendizaje obtenido tras la ejecución del proyecto, así mismo, se abordarán los posibles casos de uso como propuesta de trabajo futuro a abordar para mejorar las limitaciones del modelo.

2 Descripción de los elementos de un SOC

Un SOC (Centro de Operaciones de Seguridad) se refiere al equipo responsable de garantizar la seguridad de la información. Este equipo está compuesto por analistas e ingenieros de seguridad, así como por gerentes que supervisan las operaciones de seguridad (IBM, s. f.-a).

El SOC es una central de seguridad informática que se encarga de prevenir, monitorear y controlar la seguridad en una red determinada. Sus servicios van desde el diagnóstico de vulnerabilidades hasta la recuperación de desastres, pasando por la respuesta a incidentes, neutralización de ataques, programas de prevención, administración de riesgos y alertas de antivirus informáticos (IMF, 2020).

El principal objetivo de un SOC es detectar, analizar y corregir incidentes de ciberseguridad utilizando soluciones tecnológicas y enfoques diferentes. De esta manera, garantiza que los posibles incidentes de seguridad se identifiquen, analicen, defiendan, investiguen e informen adecuadamente (ORACLE, 2023).

2.1 Roles principales en un equipo SOC

Un equipo SOC está compuesto por varios roles que trabajan juntos para proteger la seguridad de la empresa. Estos roles pueden variar según la organización, los roles más comunes son (Check Point, 2023; IBM, s. f.-a; Tecnógrafos, 2022):

Analista de Alertas (Alert Analyst):

Es el encargado de revisar y analizar las alertas de seguridad generadas por herramientas de detección de intrusiones y otras soluciones de seguridad. Son esencialmente los primeros en responder a los incidentes de ciberseguridad, es común nombrarlos técnicos de nivel 1. Es muy importante la eficacia de los involucrados en este nivel, puesto que deben saber filtrar lo que es un falso positivo de una incidencia real.

Investigador de incidentes (Incident Responder):

Es el encargado de investigar, clasificar y responder a los incidentes de seguridad detectados. También es responsable de coordinar la respuesta a incidentes y trabajar con otros miembros del equipo para mitigar la amenaza. Se consideran técnicos de nivel 2. Este analista debe identificar que incidencias de seguridad que puedan poner en peligro la organización y cuáles no, evaluando por ello el nivel de riesgo que representan.

Analista de Seguridad (Security Analyst):

Es el encargado de monitorizar los sistemas y detectar cualquier actividad sospechosa. También se encarga de analizar los datos recopilados para identificar amenazas y tomar medidas para mitigarlas. Se consideran técnicos de nivel 3. Este es el analista más común donde dependiendo de la experiencia y habilidad puede adaptarse a la mayoría de roles del equipo SOC.

Ilustración 2: Roles de un equipo SOC.

Investigador de Seguridad (Security Investigator):

Se encarga de recopilar y examinar pruebas para determinar la naturaleza y el alcance de los incidentes, y luego tomar medidas para remediar el problema y prevenir futuros ataques. Los Investigadores de Seguridad también están involucrados en la detección temprana de posibles amenazas y vulnerabilidades, y pueden utilizar herramientas y técnicas avanzadas de análisis forense para identificar patrones sospechosos. Además, pueden trabajar en la educación y concientización de los empleados sobre las mejores prácticas de seguridad informática para minimizar los riesgos de seguridad en la organización.

Cazadores de amenazas (Threat Hunters):

Es un analista altamente especializado y experimentado en seguridad cibernética que buscan de forma proactiva las amenazas y vulnerabilidades en los sistemas de una organización que aún no han sido detectadas. Para mitigarlas antes de que se conviertan en un problema grave, de esta forma ayudan a reducir el tiempo de respuesta y minimizar el impacto en la organización.

Arquitecto de seguridad (Security Architect):

Es el encargado de diseñar e implementar la arquitectura de seguridad de una organización. El objetivo principal de un Security Architect es asegurarse de que la infraestructura y los sistemas de la organización estén diseñados de manera segura y sean resistentes a las amenazas cibernéticas.

Gestor de Inteligencia de Amenazas (Threat Intelligence Manager):

Es el encargado dirigir el equipo de inteligencia de amenazas de una organización, recopilando y analizando información de amenazas para identificar patrones y tendencias que puedan ser útiles para prevenir y responder a futuros ataques. También es responsable de desarrollar y liderar la estrategia de inteligencia de amenazas de la organización y trabajar en estrecha colaboración con otros equipos de seguridad para garantizar una respuesta coordinada a las amenazas cibernéticas

Estos roles pueden variar según la organización y sus necesidades específicas, pero todos trabajan de manera coordinada para proteger la seguridad de la empresa y mitigar cualquier amenaza de seguridad.

2.2 Flujo de Trabajo

Los equipos SOC funcionan siguiendo el mismo flujo de trabajo, esto permite que cada miembro del equipo sepa cuál es su misión, y pueda centrarse en realizar sus tareas particulares con la finalidad de cumplir el objetivo general de un equipo SOC. Existen varias propuestas, pero el estándar que responde a las necesidades que demanda una organización de un equipo SOC sigue los siguientes pasos (González, 2021; kaspersky, 2023; Tecnógrafos, 2022; Zoho Corporation, 2023):

1. La recolección de logs y eventos generados por la operación diaria de la infraestructura informática de la organización, incluyendo sistemas operativos, middleware, software específico, *switches*, *routers*, *firewalls*, NAC (Control de Acceso a la Red), servidores, almacenamiento, entre otros.

2. La información recolectada es analizada mediante herramientas que definen un patrón base de comportamiento y buscan anomalías o eventos no previstos. Estas herramientas aplican heurísticas para detectar eventos extraños dentro del comportamiento base. Con la ayuda de estas herramientas, el Alert Analyst podrá detectar anomalías o incidencias y pasársela a un técnico de nivel 2, el Incident Responder, si procede.

3. El Incident Responder, aparte de configurar las herramientas de análisis de logs, es responsable de determinar si la anomalía reportada por los técnicos de nivel 1 constituye una incidencia real de seguridad o no. Si se confirma que el evento representa un problema de seguridad, el Incident Responder abrirá un caso para clasificar el problema y asignar

un nivel de riesgo correspondiente. Una vez que se ha clasificado el problema, el Incident Responder asigna el caso a un técnico de nivel 3, como un Security Analyst o un Security Investigator.

Ilustración 3: Flujo de trabajo de un equipo SOC.

4. Cuando un técnico de nivel 3 del equipo SOC recibe un caso, su primera tarea es evaluar los activos de información afectados. Una vez que se comprende el alcance del problema, se procede a dar una respuesta inicial para mitigar los daños y comenzar a recuperar los servicios y activos afectados. En esta fase, se llevan a cabo las primeras acciones para minimizar el impacto del incidente, como la eliminación de *malware* o la desactivación de cuentas de usuario comprometidas. También puede ser necesario aislar o desconectar los sistemas afectados para evitar una mayor propagación del incidente.

5. Una vez que se han tomado medidas iniciales, el técnico de nivel 3 procederá a recopilar y analizar más información para determinar la causa raíz del incidente, su expansión dentro de la organización y cómo ha sucedido para tomar medidas y prevenir futuros incidentes similares.

6. Una vez que se han tomado las medidas necesarias para eliminar la brecha de seguridad y se han recuperado los servicios o activos afectados, se procede a documentar el caso y cerrarlo. Es esencial documentar correctamente el caso, ya que esto facilitará la resolución más rápida y eficaz de problemas similares que puedan surgir en el futuro. Esta información es de gran utilidad para Threat Intelligence Manager.

2.3 Elementos de un SOC

Para cumplir con los objetivos de proteger, detectar y responder, el SOC se nutre de múltiples herramientas.

SIEM (Security Information and Event Management):

SIEM son las siglas en inglés de información de seguridad y gestión de eventos. Este es una herramienta de seguridad crítica para cualquier organización, ya que ayuda a detectar y responder a posibles amenazas cibernéticas. Los SIEM pueden proporcionar una amplia variedad de funciones, como la recolección de registros, la correlación de eventos, la detección de amenazas, la gestión de incidentes y la generación de informes (IBM, s. f.-b).

- **Recolección de registros**: permite recopilar información de eventos de seguridad desde diversos sistemas y dispositivos en la red. Los registros se recopilan y almacenan en una ubicación centralizada para que puedan ser analizados posteriormente.

- **Correlación de eventos**: permite identificar patrones y anomalías en los eventos de seguridad recopilados. Al correlacionar eventos de diferentes sistemas y dispositivos, un SIEM puede identificar comportamientos sospechosos y detectar posibles ataques cibernéticos.

- **Detección de amenazas**: el análisis de eventos de seguridad en tiempo real permite a los SIEM detectar posibles amenazas y generar alertas y notificaciones en tiempo real para que los equipos de seguridad puedan responder rápidamente a los incidentes.

- **Gestión de incidentes**: la integración de esta función con otras herramientas de seguridad, un SIEM puede ayudar a los equipos de seguridad a responder a incidentes de seguridad de manera efectiva y rápida.

- **Generación de informes**: esto permite generar informes y el análisis de datos para ayudar a los equipos de seguridad a comprender mejor las tendencias de seguridad en la organización, identificar áreas de riesgo y mejorar la seguridad en general.

El conjunto de funciones que engloba el SIEM lo consolida como una herramienta esencial en la seguridad de la información de una organización, ya que ayuda a detectar y responder rápidamente a posibles amenazas cibernéticas mediante la recolección, correlación y análisis de eventos de seguridad en tiempo real.

EDR (Endpoint Detection and Response):

La detección y respuesta de *endpoint* o EDR es una solución de seguridad informática que se enfoca en la protección de los dispositivos finales, tales como computadoras, servidores, dispositivos móviles, etc. La solución EDR monitorea y analiza el comportamiento de los *endpoints* en tiempo real, detectando y respondiendo a posibles amenazas de seguridad (Check Point, 2021a; INCIBE, 2021; vmware, 2023).

La solución EDR puede incluir diversas capacidades, como la detección y prevención de intrusiones, la protección contra *malware*, la monitorización del comportamiento del usuario y la detección de actividad maliciosa en tiempo real. Estas capacidades pueden ser implementadas a través de técnicas de inteligencia artificial, *machine learning* y análisis de comportamiento. Una solución de EDR debe contar con los siguientes componentes:

- **Flujo de evaluación de** prioridades de incidentes: debe clasificar automáticamente los eventos potencialmente sospechosos o maliciosos, lo que permite al analista de seguridad priorizar las investigaciones.

- **Búsqueda de amenazas**: deben brindar asistencia para las actividades de búsqueda de amenazas a fin de permitir que los analistas de seguridad busquen posibles intrusiones de manera proactiva.

- **Adición y enriquecimiento de datos**: deben usar la mayor cantidad de datos que estén disponibles a fin de tomar decisiones fundamentadas sobre posibles amenazas.

Además de detectar y responder a amenazas de seguridad en tiempo real, las soluciones EDR también permiten a los equipos de seguridad investigar y analizar incidentes pasados, proporcionando visibilidad completa del *endpoint* y de las actividades realizadas en él. Esto permite una respuesta más rápida y eficiente a incidentes de seguridad y una mejor comprensión de los patrones de ataque.

SOAR (Security Orchestration Automated Response):

SOAR son las siglas en inglés de organización, automatización y respuesta de la seguridad. Este es una plataforma de seguridad que combina la orquestación, automatización y respuesta para ayudar a los equipos de seguridad a mejorar la eficiencia y la eficacia en la gestión de incidentes de seguridad, permitiendo acceder a la información sobre las amenazas, ejecutar consultas y compartir los datos de forma centralizada (Red Hat, 2022a).

- La orquestación permite a los equipos de seguridad automatizar tareas repetitivas y coordinar la interacción entre diferentes sistemas de seguridad y herramientas de respuesta.

- La automatización permite a los equipos de seguridad reducir el tiempo necesario para manejar incidentes de seguridad.

- La respuesta ayuda a los equipos a tomar medidas proactivas para prevenir futuros incidentes de seguridad.

Los equipos de seguridad a menudo tienen que lidiar con un gran número de herramientas y productos diferentes que no están integrados entre sí, como los sistemas de detección y respuesta de *endpoints* (EDR), firewalls y soluciones de gestión de información y eventos de seguridad (SIEM). La gestión manual de estos sistemas puede ralentizar el proceso de detección y resolución de problemas, generar errores en la configuración y falta de uniformidad en la aplicación de políticas, lo que puede exponer a los sistemas a riesgos de cumplimiento y ataques graves. La automatización ayudar a acelerar las operaciones diarias e integrar la seguridad en los procesos, aplicaciones e infraestructura desde el principio.

IDS (Intrusion Detection System):

IDS son las siglas en inglés de Sistemas de Detección de Intrusos. Estos son sistemas de seguridad informática diseñados para detectar actividades maliciosas o no autorizadas en una red o sistema informático. Los IDS monitorean el tráfico de red en busca de patrones y comportamientos anormales, como intentos de acceso no autorizados, escaneo de puertos, *malware*, tráfico sospechoso y otros indicadores de compromiso. Una vez que se detecta una actividad maliciosa, el IDS es un sistema pasivo que genera una alerta para que los responsables de respuestas a incidentes puedan investigar el incidente y tomar las medidas necesarias para proteger la red o sistema. A diferencia de un firewall, no está diseñado para actuar como un sistema de protección. Los IDS pueden ser implementados como soluciones de hardware o software, y pueden ser configurados para operar en diferentes niveles de la red, desde la capa de enrutamiento hasta los servidores y los *endpoints* individuales (Check Point, 2021b).

En función de su localización se pueden clasificar en:

- **HIDS (Host-Based IDS)**: de sus siglas en inglés IDS basado en host. Un HIDS es una solución de seguridad que se instala en un *endpoint* específico para protegerlo contra amenazas internas y externas. Este tipo de IDS puede monitorear el tráfico de red que entra y sale del dispositivo, supervisar los procesos en ejecución y examinar los registros del sistema. Aunque la visibilidad de un HIDS se limita al equipo anfitrión, proporciona una visibilidad profunda de los componentes internos de la computadora host, lo que permite una detección más precisa y detallada de amenazas potenciales. Sin embargo, esto también puede limitar el contexto disponible para la toma de decisiones.

- **NIDS (Network-Based IDS)**: de sus siglas en inglés IDS basado en red. Una solución de NIDS se encarga de supervisar toda una red protegida. Está diseñada para tener visibilidad de todo el tráfico que fluye a través de la red y tomar decisiones basadas en metadatos y contenido de

paquetes. Al tener una vista panorámica de la red, puede detectar amenazas a gran escala y proporcionar un contexto más amplio para la toma de decisiones. Sin embargo, estos sistemas no tienen visibilidad de los componentes internos de los *endpoints* que protegen, lo que limita su capacidad para detectar amenazas específicas del host.

Debido a los diferentes niveles de visibilidad, la implementación de un HIDS o de un NIDS se suelen gestionar de forma unificada, proporcionando una seguridad integral.

En función de la forma de identificar las posibles intrusiones:

- **Detección de firmas**: estos utilizan patrones de amenazas conocidas para identificarlas. Cuando el IDS identifica *malware* u otro contenido malicioso, se genera una firma y se agrega a la lista empleada por el sistema para analizar el contenido entrante. Esto permite que el IDS detecte amenazas conocidas con una alta tasa de precisión, ya que todas las alertas se generan en función de la detección de contenido malicioso previamente identificado. Sin embargo, un IDS basado en firmas no puede detectar amenazas desconocidas o vulnerabilidades de día cero (*zero-day exploit*).

- **Detección de anomalías**: estos buscan detectar comportamientos anómalos que puedan ser indicativos de un ataque. Para ello, crean un modelo del comportamiento normal del sistema protegido y comparan cualquier comportamiento futuro con este modelo. Cualquier anomalía que se detecte se etiqueta como una posible amenaza y se generan alertas para los analistas de seguridad. Aunque este enfoque puede detectar amenazas nuevas o de día cero, la dificultad de generar un modelo preciso de comportamiento normal significa que estos sistemas deben equilibrar los falsos positivos con los falsos negativos.

- **Detección híbrida**: este utiliza tanto la detección basada en firmas como la detección basada en anomalías. Esto le permite detectar una mayor cantidad de ataques potenciales con una menor tasa de error que si se empleara cualquiera de los sistemas de forma aislada.

Dado que un firewall es un dispositivo de protección activo, es más parecido a un sistema de prevención de intrusiones (*Intrusion Prevention System*, IPS) que a un sistema de detección de intrusos. Un IPS es como un IDS, con la diferencia que bloquea activamente las amenazas identificadas en lugar de simplemente generar una alerta. Este método complementa la funcionalidad de un firewall, y muchos firewalls de última generación *(Next Generation Firewalls*, NGFW) tienen las funcionalidades de IDS/IPS integradas. Esto les permite aplicar las reglas de filtrado predefinidas (firewalls) y detectar y responder a amenazas cibernéticas más sofisticadas (IDS/IPS).

Threat Intelligence:

El *Threat Intelligence* o Inteligencia de Ciberamenazas, es un proceso continuo de recopilación, análisis y diseminación de información sobre las amenazas y riesgos existentes para una organización. Esta información se utiliza para mejorar la postura de seguridad de la organización y ayudar en la toma de decisiones estratégicas. Los datos recopilados pueden incluir indicadores de compromiso (IoC), atributos de amenazas, información de inteligencia de adversarios, tácticas, técnicas y procedimientos (TTP), así como cualquier otra información relevante para la seguridad de la organización. *Threat Intelligence* es un componente clave para la identificación y respuesta temprana a los ataques cibernéticos y ayuda a las organizaciones a mantenerse por delante de los adversarios (Check Point, 2023; González, 2021; kaspersky, 2023).

La inteligencia de amenazas abarca varias actividades clave:

- **Recopilación de datos**: Reunión de información de una amplia gama de fuentes, incluidos proveedores de seguridad, organizaciones de investigación, agencias gubernamentales y comunidades de seguridad.

- **Análisis**: Evaluación de los datos recopilados para identificar patrones, tendencias y posibles riesgos. Esto implica comprender las motivaciones, capacidades y objetivos de los actores de amenazas.

- **Contextualización**: Agregar contexto relevante a los datos recopilados, como el sector industrial, la ubicación geográfica y la infraestructura y activos específicos de la organización.

- **Difusión**: Compartir la inteligencia de amenazas analizada y contextualizada con las partes interesadas pertinentes, como equipos de seguridad, ejecutivos y otros departamentos cruciales dentro de la organización.

- **Inteligencia accionable**: Convertir la información recopilada en conocimientos accionables que puedan guiar las decisiones de seguridad y ayudar a las organizaciones a priorizar sus defensas y esfuerzos de respuesta a incidentes.

Threat Intelligence se puede categorizar en diferentes tipos según la fuente de información o el nivel de detalle. Estos incluyen inteligencia estratégica, inteligencia operativa, inteligencia táctica, inteligencia técnica e indicadores de compromiso.

La inteligencia de amenazas se puede clasificar en diferentes tipos según la fuente de información o el nivel de detalle. Estos incluyen:

- **Inteligencia de amenazas estratégica**: Es un análisis de alto nivel que está destinado a un público no técnico, como la alta dirección de una empresa u organización. Se enfoca en cuestiones de seguridad cibernética que pueden tener un impacto en las decisiones empresariales más amplias y considera las tendencias y motivaciones

generales. La inteligencia de amenazas estratégica a menudo se basa en información de acceso público, como informes de medios, documentos técnicos e investigaciones.

- **Inteligencia de amenazas táctica**: Se enfoca en el futuro inmediato y está destinada a un público más técnico. Identifica indicadores de compromiso simples para que los equipos de TI puedan buscar y eliminar amenazas específicas en una red. Los IoC incluyen elementos como direcciones IP sospechosas, nombres de dominios maliciosos conocidos, tráfico inusual, alertas de inicio de sesión o un aumento en las solicitudes de archivos/descargas. La inteligencia táctica es la forma más sencilla de inteligencia de amenazas y a menudo se automatiza. Tiene una vida útil corta, ya que muchos IoC se vuelven obsoletos rápidamente.

- **Inteligencia de amenazas operativa**: Se enfoca en el "quién", "por qué" y "cómo" detrás de cada ciberataque. La inteligencia de amenazas operativa busca responder estas preguntas mediante el análisis de ciberataques anteriores y sacando conclusiones sobre la intención, el momento y la sofisticación. La inteligencia de amenazas operativa requiere más recursos que la inteligencia táctica y tiene una vida útil más larga. Esto se debe a que los ciber atacantes no pueden cambiar sus tácticas, técnicas y procedimientos (TTP) con la misma facilidad con la que cambian sus herramientas, como un tipo de *malware* específico.

2.4 Tipos de SOC

En la actualidad, la implementación de un SOC es fundamental para la gestión de la seguridad en una organización. Existen diversas soluciones en el mercado, pudiendo distinguir entre tres categorías: la contratación de un SOCaaS, la utilización de un SOCaaP o la creación de un SOC interno. Para decantarse por alguna se debe evaluar cuidadosamente las opciones y seleccionar el modelo de SOC adecuado que se ajuste a las necesidades y recursos del usuario. A continuación, exploraremos con más detalle cada una de las opciones disponibles y sus ventajas y desventajas (Jiménez, 2022; Red Hat, 2022b).

SOC as a Service (SOCaaS):

Sigue el modelo SaaS (Software as a Service), por el cual el proveedor proporciona el software y las aplicaciones a través de internet, lo que permite a las organizaciones externalizar la gestión de su operación de seguridad. En este modelo, una empresa de servicios de seguridad proporciona a la organización una plataforma de seguridad, personal de seguridad altamente calificado y herramientas de seguridad avanzadas. Mediante este esquema, el proveedor de SOCaaS es responsable de alojar, mantener y actualizar el SOC,

mientras que el usuario paga una tarifa mensual o anual por la gestión del SOC.

Este tipo de servicio puede ofrecer beneficios como una rápida implementación y actualización del sistema, una mayor escalabilidad, acceso a tecnología avanzada de seguridad y personal altamente capacitado en seguridad. Además, SOCaaS también puede ser una opción asequible para las empresas que no tienen los recursos internos para gestionar una operación de seguridad de forma autónoma.

Los proveedores de SOCaaS ofrecen una amplia gama de servicios de seguridad, que incluyen:

- Monitorización y detección de amenazas en tiempo real.
- Análisis de incidentes de seguridad.
- Investigación de amenazas avanzadas.
- Gestión de vulnerabilidades y parches.
- Respuesta a incidentes.
- Informes de seguridad y cumplimiento.

El uso de SOCaaS puede ayudar a las organizaciones a mejorar su postura de seguridad y reducir el riesgo de brechas de seguridad. Por ejemplo, virus, *malware*, *ransomware* que puedan solicitar posteriormente un rescate económico, ataques DDoS que bloqueen un servidor y tengamos que llevar a cabo una acción rápida, etc. Sin embargo, el ahorro en tiempo y mantenimiento que ofrece podría suponer un costo en términos de control, seguridad y rendimiento, por eso, es importante que las organizaciones seleccionen un proveedor de SOCaaS con experiencia y con una buena reputación en el mercado. Además, las organizaciones también deben tener en cuenta las regulaciones y leyes aplicables en su jurisdicción antes de externalizar la gestión de su seguridad a un proveedor de SOCaaS.

SOC as a Platform (SOCaaP):

Sigue el modelo PaaS (Platform as a Service), por el cual el proveedor proporciona una plataforma de seguridad en línea, donde los usuarios pueden utilizarla para gestionar su operación de seguridad interna. En este modelo, la organización es responsable de la gestión de su propia operación de seguridad, mientras que el proveedor de la plataforma de seguridad proporciona las herramientas, servicios y tecnología necesarios para implementar y gestionar su propia operación de seguridad.

Mediante este esquema, la función del proveedor es exclusivamente ofrecer una plataforma completa de SOC en la nube que permita a los usuarios crear,

probar, implementar y gestionar su propio SOC, sin la necesidad de preocuparse por la complejidad de la infraestructura subyacente. A cambio, los usuarios pagan una tarifa mensual o anual por el acceso a la plataforma SOC.

La principal ventaja de SOCaaP es que ofrece rápida implementación de aplicación, una mayor flexibilidad y control sobre la gestión de la seguridad interna de una organización. La organización puede personalizar y ajustar su operación de seguridad según sus necesidades específicas y requisitos regulatorios. Además, SOCaaP también puede ser una opción más adecuada para las organizaciones que disponen de personal interno cualificado para gestionar su propia operación de seguridad.

La plataforma SOCaaP ofrece una amplia gama de herramientas y servicios de seguridad, entre las herramientas software de SOC en el mercado podemos encontrar (COOPER, 2023; Hitesh, 2022):

- SolarWinds Security Event Manager.
- CrowdStrike Falcon.
- Heimdal Threat Hunting and Action Center.
- ManageEngine Log360.
- LogRhythm XDR Stack.

El modelo SOCaaP, depende en gran medida de la capacidad de la organización para gestionar y operar su propia operación de seguridad. Por lo tanto, es fundamental que la organización cuente con personal altamente capacitado en seguridad y tenga la capacidad de integrar y utilizar de manera efectiva la plataforma de seguridad proporcionada por el proveedor.

Aunque el usuario es responsable de la implementación y gestión del SOC en su propia organización, el proveedor de SOCaaP puede proporcionar soporte y asistencia técnica según lo acordado en el contrato de servicio. Esta solución es atractiva para aquellas organizaciones que no tienen los recursos internos para implementar y gestionar un SOC completo, pero que desean tener el control total sobre su propia operación de seguridad y no quieren depender completamente de un proveedor externo.

SOC propietario:

Un SOC propietario es un centro de operaciones de seguridad que es propiedad y está operado íntegramente por la organización. En este modelo, la organización es responsable de diseñar, implementar y operar su propio SOC, lo que significa que debe tener personal altamente capacitado en seguridad, una infraestructura adecuada y herramientas de seguridad adecuadas para lograr una operación de seguridad efectiva.

El SOC propietario es una solución atractiva para organizaciones que tienen un alto grado de control y propiedad sobre su operación de seguridad, y que desean mantener la seguridad de sus sistemas y datos dentro de la empresa. Al tener un SOC interno, la organización tiene el control total sobre las políticas de seguridad, los procesos y las herramientas utilizadas para proteger sus activos críticos. Los SOC internos permiten un alto nivel de personalización a las necesidades específicas de la empresa.

Sin embargo, la implementación y operación de un SOC propietario puede ser costosa y compleja, ya que la organización debe invertir en la contratación y capacitación de personal especializado en seguridad, la adquisición de herramientas y tecnologías de seguridad, y la infraestructura adecuada. Además, la organización es responsable de mantener su SOC actualizado con las últimas tendencias y amenazas en seguridad.

La decisión de crear un SOC propio o contratar uno dependerá de los recursos y la experiencia de seguridad que tenga la organización. Crear un SOC interno da un mayor control sobre la seguridad y permite personalizar la solución para satisfacer las necesidades únicas de la organización. Sin embargo, crear un SOC propietario puede requerir una inversión significativa en personal, tecnología y capacitación.

Por otro lado, contratar un SOCaaS o un SOCaaP puede ser una opción más rentable para las organizaciones que no tienen los recursos internos suficientes para gestionar su propia operación de seguridad. Un SOCaaS o un SOCaaP también le proporciona acceso a un equipo de expertos en seguridad de la información que puede monitorear continuamente su red y detectar y responder a las amenazas de manera rápida y eficaz.

En términos de seguridad de red para uso doméstico, es posible que no se necesite un SOC completo, pero es útil tener un enfoque en la seguridad en la red. Para uso doméstico, la contratación de un SOCaaS o un SOCaaP implica una infrautilización de los servicios, suponiendo de esta forma un gasto elevado. En contraposición, la creación de un SOC permite personalizar y adaptarlo a los recursos disponibles, evitando el sobredimensionado del SOC. Sin embargo, la creación del SOC desde cero es una labor compleja, por lo que la mejor solución en este caso es la utilización de diversas herramientas especializadas que, al interactuar entre sí, ofrezcan una seguridad completa y efectiva como si se tratara de un SOC integrado.

3 Diseño y arquitectura propuesta

A lo largo de este capítulo se expondrá el diseño de la arquitectura SOC a implementarse en un entorno doméstico haciendo uso exclusivo de herramientas *open-source*.

Este capítulo se organiza en dos apartados, cada una de las cuales aborda aspectos clave de la propuesta presentada. El primer apartado se centra en proporcionar una explicación detallada de las diversas herramientas utilizadas en el modelo, mientras que el segundo apartado se dedica a describir la arquitectura propuesta y la interacción de las distintas herramientas dentro del modelo propuesto. En conjunto, estos dos apartados proporcionan una comprensión completa del enfoque empleado en este trabajo y cómo se aplicó para lograr los resultados deseados.

3.1 Herramientas a emplear

En este apartado se proporcionará una explicación detallada de las diversas herramientas empleadas en el SOC. Se describen las funciones y características de cada herramienta. En la medida de lo posible nos centraremos en el uso de software *open-source* gratuito.

Estar actualizado en ciberseguridad es fundamental debido al panorama de amenazas en línea que está en constante evolución, lo que significa que las herramientas y técnicas empleadas para prevenir y protegerse contra las amenazas también deben actualizarse constantemente. El empleo de herramientas *open-source* en ciberseguridad pueden ser de gran utilidad. Estas herramientas suelen ser actualizadas con mayor frecuencia que las herramientas propietarias, gracias a la comunidad de desarrolladores y usuarios que contribuyen con mejoras y actualizaciones constantes. Además, la personalización y adaptación de las herramientas *open-source* puede ser útil para enfrentar nuevas amenazas y vulnerabilidades, lo que promueve la innovación y el desarrollo colaborativo en el ámbito de la ciberseguridad (IKUSI, 2023).

Elasticsearch:

Elasticsearch es una base de datos NoSQL de búsqueda y análisis de datos altamente escalable y optimizada para búsquedas en tiempo real y análisis de datos. Ofrece una potente capacidad de búsqueda y filtrado de datos de registro, lo que facilita el análisis y la comprensión de los datos. Además, Elasticsearch se integra con una amplia gama de herramientas y plataformas, lo que lo hace fácil de usar en una variedad de entornos. También cuenta con una gran comunidad de usuarios y desarrolladores, lo que proporciona una amplia documentación y soporte (AWS, 2023; elastic, 2023).

Ilustración 4: Elasticsearch.

Los registros o logs son un tipo de dato detallado de eventos o actividades realizadas por un sistema informático. Estos registros suelen incluir información sobre el usuario, la fecha y hora, el tipo de evento, la ubicación y cualquier información relevante sobre la actividad. Del mismo, los registros pueden contener información sobre errores, eventos, transacciones y otros detalles importantes del sistema.

Los registros se generan con gran frecuencia, es por ello, que requieren de un correcto almacenamiento y gestión. Analizar y comprender estos registros es esencial para mantener un sistema informático seguro, confiable y eficiente. Elasticsearch es una herramienta sólida y confiable. Es el motor de búsqueda más popular utilizado para el análisis de registro en tiempo real y el almacenamiento de estos.

TheHive:

TheHive es una plataforma de respuesta a incidentes de seguridad de código abierto que ayuda a gestionar y analizar los incidentes de manera colaborativa y eficiente. La plataforma es gratuita y está disponible para cualquier persona para descargar, utilizar y modificar de acuerdo con los términos de la licencia de código abierto Apache 2.0, contando con una API en Python 3 (TheHive Project, 2022b).

Ilustración 5: TheHive.

TheHive es compatible con varias organizaciones y proyectos relacionados con la ciberseguridad, como el proyecto MISP, que proporciona una plataforma de *Threat Sharing*, y la plataforma Cortex, que proporciona capacidades de análisis y orquestación de seguridad cibernética. Estas organizaciones y

proyectos colaboran y contribuyen al desarrollo y mantenimiento de la plataforma.

Una de las características clave de TheHive es la visualización y gestión centralizada de incidentes desde una única plataforma común, esto permite ahorrando tiempo y esfuerzo. Además, TheHive facilita la colaboración entre los miembros del equipo, permitiendo la asignación de tareas y el intercambio de información y comentarios de manera eficiente. Otra característica importante es la capacidad de automatización de tareas, lo que ayudar a responder a los incidentes de manera rápida y efectiva.

TheHive realiza la gestión de los incidentes en casos. Los casos se subdividen en tareas (pensar en identificación, contención, erradicación, verificar registros de proxy, etc.) y observables (direcciones IP, hashes, direcciones de correo electrónico, nombres de dominio, URL ...). Cuando los analistas están trabajando en tareas, agregan registros a medida que avanzan. En la terminología de TheHive, los registros son entradas de texto que pueden contener archivos adjuntos para ayudar a los analistas a registrar lo que han estado haciendo.

Los observables pueden etiquetarse, marcarse como IoC y analizarse. Cuando la investigación esté en progreso o finalizada, es posible que deseemos compartir los IoC resultantes o un subconjunto de aquellos con algún otro departamento. Esto también se puede ejecutar de manera intuitiva desde la misma plataforma, aunque para nuestro caso particular no lo tendremos en cuenta, dado que nos centraremos en la gestión de los incidentes por una única persona.

TheHive utiliza el Protocolo de Semáforo (Traffic Light Protocol o TLP) como marcador de confidencialidad de los observables. TLP proporciona un esquema simple e intuitivo para indicar cuándo y cómo se puede compartir información sensible, facilitando una colaboración más frecuente y efectiva. Por defecto, cualquier observable agregado se considera TLP: AMBAR. El TLP es una característica que afecta en la posibilidad de ejecución de ciertos analizadores.

Por otro lado, TheHive también hace uso del módulo PAP (Permissible Actions Protocol) este hace del mismo uso de semáforos que TLP, pero en este caso, corresponde al protocolo a seguir por el analista ante el incidente. Esto hace referencia a si tiene o no que permitir al atacante saber que la incidencia ha sido detectada por el sistema, es decir, si tiene que realizar acciones pasivas de visualización y seguimiento o acciones activas de bloqueo y redirección de tráfico.

Cortex:

Cortex es un concentrador de analizadores de ciberseguridad de código abierto, permitiendo integrar y orquestar diferentes herramientas y servicios de análisis de seguridad para mejorar la capacidad de detección y respuesta a amenazas. Al orquestar estas herramientas, Cortex puede ejecutar análisis

automatizados y personalizados en diferentes tipos de recursos, como archivos, direcciones IP, URLs y nombres de dominio. Cortex también proporciona una interfaz de programación de aplicaciones (API) que permite la integración con otras herramientas y plataformas de seguridad cibernética (TheHive Project, 2022a).

Ilustración 6: Cortex.

Con el uso de la API, Cortex se integra con TheHive, actuando conjuntamente como una plataforma de orquestación y análisis de ciberseguridad, ayudando a los equipos a automatizar y orquestar los procesos de análisis de amenazas. La plataforma es altamente extensible que permite a los usuarios crear y compartir sus propios analizadores de IoC. Cortex es una plataforma altamente personalizable que puede integrarse con una amplia gama de herramientas de ciberseguridad para satisfacer las necesidades específicas de los equipos.

La comunidad de desarrolladores y usuarios ha originado y compartido una gran cantidad de analizadores de seguridad para integrarse con la plataforma, lo que proporciona una amplia variedad de opciones de análisis para los usuarios. Actualmente, cuenta con más de 200 analizadores para servicios populares como VirusTotal, Joe Sandbox, DomainTools, Google Safe Browsing y Shodan.

MISP**:**

MISP (Malware Information Sharing Platform) es una plataforma de intercambio de información de seguridad que permite compartir y colaborar en la información de amenazas entre diferentes organizaciones y equipos de seguridad cibernética. Cortex se integra con MISP para permitir a los usuarios buscar y compartir información de amenazas en MISP directamente desde la plataforma Cortex. Esta integración ayuda a los equipos de seguridad cibernética a obtener información adicional sobre las amenazas y a tomar decisiones más informadas en sus procesos de detección y respuesta de amenazas (KEEPCODING, 2023a; MISP Threat Sharing, s. f.).

Ilustración 7: MISP.

MISP permite a los usuarios compartir información de manera segura y controlada, manteniendo la privacidad de la información confidencial. También proporciona herramientas de análisis para correlacionar y enriquecer los datos compartidos, lo que ayuda a los equipos de seguridad a identificar patrones y tendencias en las amenazas de seguridad. Además, la integración de MISP en Cortex permite a los usuarios automatizar y orquestar procesos de detección y respuesta de amenazas basados en la información de amenazas compartida en MISP.

Suricata:

Suricata, desarrollada y mantenida por OSIF (Open Information Security Foundation), es una plataforma de ciberseguridad de código abierto que ofrece funcionalidades de IDS en tiempo real, IPS en línea y monitorización de seguridad de red. Su tecnología de inspección profunda de paquetes (DPI) permite detectar y prevenir amenazas, incluyendo el tráfico cifrado utilizando TLS/SSL, ampliando las capacidades de la versión gratuita del sistema más popular Snort (KEEPCODING, 2023b).

Ilustración 8: Suricata.

Suricata es capaz de detectar una amplia gama de amenazas, incluyendo ataques de red, *malware*, intentos de explotación de vulnerabilidades, tráfico de comando y control, y más. Para hacerlo, utiliza una combinación de firmas basadas en reglas y análisis de comportamiento. Además, Suricata puede

integrarse con otras herramientas de seguridad cibernética, como el SIEM, y puede enviar alertas a estos sistemas para su análisis y seguimiento.

Suricata se puede personalizar y configurar para adaptarse a las necesidades específicas de la organización, y se pueden agregar reglas personalizadas para detectar amenazas específicas. Además, Suricata cuenta con una comunidad de usuarios y desarrolladores activos que contribuyen con actualizaciones de seguridad y mejoras de funcionalidad para el sistema.

Suricata es un sistema NGFW, en el que las reglas desempeñan un papel crucial. Estas reglas son esenciales para la identificación y alerta de posibles amenazas y anomalías en el tráfico de red. Su función principal es detectar patrones específicos en los paquetes de red y tomar medidas correspondientes, como generar alertas, bloquear el tráfico o registrar información detallada sobre los eventos sospechosos. De esta manera, las reglas de Suricata proporcionan una capa de seguridad adicional al monitorear y proteger activamente las redes contra intrusiones y actividades maliciosas.

El formato de las reglas de Suricata está basado en el formato de reglas de Snort. Estas reglas siguen una estructura común que incluye los siguientes elementos (Suricata, s. f.-b):

1. **Acciones**: Determinan qué acciones tomar cuando se cumple una regla, como generar una alerta, bloquear el tráfico o registrar información detallada sobre el evento.

2. **Encabezado**:

    a. Cabecera de la regla: Indica el tipo de regla y contiene información básica sobre la misma, como el protocolo objetivo (por ejemplo, TCP o UDP) y el estado de la conexión (por ejemplo, nuevo paquete o paquete de respuesta).

    b. Encabezado de opciones: Proporciona detalles adicionales sobre la regla, como las direcciones IP y los puertos de origen y destino, así como los criterios específicos utilizados para detectar una amenaza o comportamiento sospechoso.

3. **Opciones de contenido**: Definen los patrones o cadenas de texto específicos que se deben buscar en los paquetes de red para identificar una amenaza. Esto puede incluir palabras clave, expresiones regulares u otros tipos de patrones.

Además de estos elementos básicos, las reglas de Suricata también admiten una amplia gama de opciones y modificadores que permiten ajustar la detección y el comportamiento del sistema según las necesidades específicas. Estas opciones incluyen condiciones avanzadas, límites de detección, supresiones de alertas, entre otros.

```
drop tcp $HOME_NET any -> $EXTERNAL_NET any (msg:"ET TROJAN Likely Bot
Nick in IRC (USA +..)"; flow:established,to_server;
flowbits:isset,is_proto_irc; content:"NICK "; pcre:"/NICK .*USA.*[0-
9]{3,}/i"; reference:url,doc.emergingthreats.net/2008124;
classtype:trojan-activity; sid:2008124; rev:2;)
```

Ilustración 9: Ejemplo de formato de regla de Suricata.

3.2 Diseño de la arquitectura

En este apartado se describe la arquitectura propuesta y la interacción de las distintas herramientas dentro del modelo. Aquí se explicarán las relaciones entre las herramientas y cómo se combinan, así como su contribución específica para lograr el objetivo general del SOC doméstico.

El SOC doméstico está orientado para el manejo exclusivo de una única persona, eso implica que las tareas de los diferentes analistas que podemos encontramos en un SOC recaen en un único sujeto, esto implica que se requiere de la simplificación del modelo, así como de la automatización del mayor número de tareas encontradas en un flujo de trabajo.

Hay que tener en cuenta que existen ciertas tareas esenciales que todo equipo SOC debe contar. Estas se pueden simplificar en 4 bloques, detección, gestión, análisis y almacenamiento. Estos bloques se correlacionan con los elementos IDS, SIEM y EDR. La interacción entre los módulos sería de la siguiente manera. El módulo de detección monitoriza la red lanzando alertas de IoC, estas alertas son gestionadas en el módulo de gestión que posteriormente serán lanzados al módulo de análisis, al finalizar el procesado de los registros serán enviados al módulo de almacenamiento (Ramiro, 2018; Tecnógrafos, 2022).

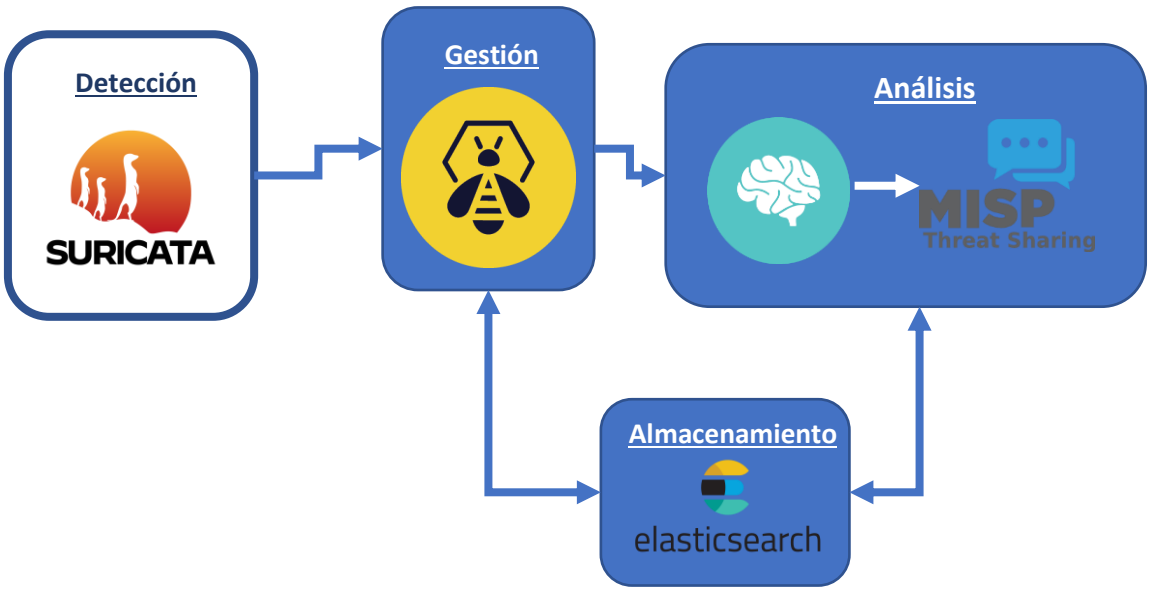

Ilustración 10: Arquitectura del SOC Doméstico.

Módulo de detección:

este contará con la herramienta Suricata, la cual efectuará las acciones de IDS/IPS funcionando como detector y/o bloqueador de amenazas en función de las reglas estipuladas en su instalación.

Módulo de gestión:
los IoC lanzados por el módulo de detección serán visualizados y gestionados con la herramienta TheHive, el cual ejerce las tareas de un SIEM correlacionando y centralizando la administración de los registros.

Módulo de análisis:

este contará con la herramienta Cortex, el cual ejecutará el análisis de los registros seleccionados en el módulo de gestión, devolviendo los resultados contrastados con los distintos analizadores.

Módulo de almacenamiento:

Este módulo contará con la herramienta Elasticsearch, el cual almacenará todos los IoC generados en los diferentes modelos.

4 Configuración de herramientas

Este capítulo alberga los detalles de la instalación, configuración e integración de las distintas herramientas necesarias la construcción del SOC doméstico.

Para la implantación correcta del SOC doméstico se requiere de las siguientes características del equipo: 8 GB RAM y 30 GB de disco duro.

Se usará el sistema operativo Ubuntu, el cual ofrece numerosas ventajas. Es software libre y gratuito, lo que permite acceder a un sistema completo sin costo alguno. Además, Ubuntu se destaca por su enfoque en la seguridad, proporcionando actualizaciones regulares para proteger el sistema contra amenazas. Su comunidad activa de usuarios y desarrolladores también contribuye a la seguridad y solución de problemas.

Accediendo a la última versión estable disponible hasta la fecha Ubuntu 22.04.2 LTS, la versión más reciente de Ubuntu con soporte a largo plazo. Teniendo la ventaja adicional de recibir actualizaciones de seguridad y mantenimiento garantizado hasta abril de 2027.

4.1 TheHive

Comenzando con la instalación de TheHive que actuara como núcleo de la arquitectura. Este centralizará las diferentes acciones de la gestión de los registros. Requiere de disponer modo root para la correcta implantación de la herramienta. La versión instalada será TheHive 4, la cual cuenta con mayor documentación y casos de usos probados. Conjuntamente, se realizará la instalación de la herramienta Cassandra, que almacenará los registros generados de la herramienta TheHive (TheHive Project, 2021).

Se requerirá tener acceso a las siguientes herramientas:

- sudo apt install python3.8
- sudo apt-get install curl
- sudo apt install net-tools

Descarga del repositorio Cassandra
Cassandra es un sistema de base de datos distribuida y escalable que se basa en el modelo NoSQL. Ofrece alta disponibilidad, escalabilidad horizontal y tolerancia a fallos, lo que lo hace adecuado para manejar grandes volúmenes de datos en entornos distribuidos.

```
curl -fsSL https://www.apache.org/dist/cassandra/KEYS  | sudo apt-key add –
echo "deb http://www.apache.org/dist/cassandra/debian  311x main" | sudo tee -a /etc/apt/sources.list.d/cassandra.sources.list
```

Instalación de Java
TheHive se puede cargar con Java 11, pero no con la versión estable de Cassandra, que requiere Java 8.

```
apt-get install -y openjdk-8-jre-headless
echo JAVA_HOME="/usr/lib/jvm/java-8-openjdk-amd64" | sudo tee -a /etc/environment
export JAVA_HOME="/usr/lib/jvm/java-8-openjdk-amd64"
```

Instalar Cassandra

```
sudo apt update
sudo apt install cassandra -y
cqlsh localhost 9042
```

En la consola de Cassandra ejecutamos los siguientes comandos:

```
UPDATE system.local SET cluster_name = 'thp' where key='local';
exit;
```

Refrescamos las tablas de las bases de datos en Cassandra

```
nodetool flush
```

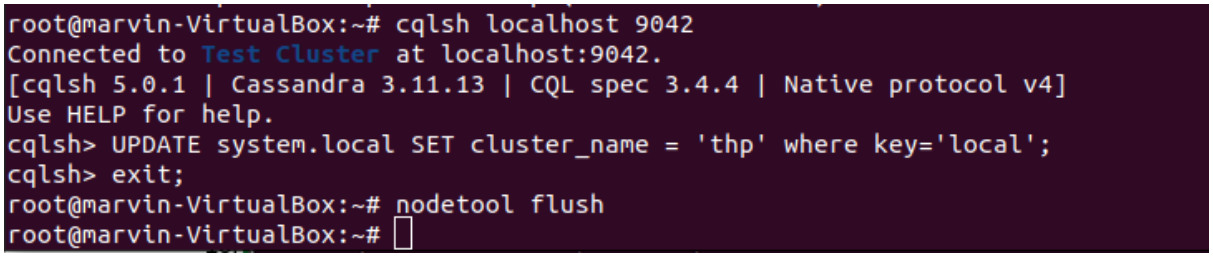

Ilustración 11: Instalación de Cassandra.

Configuración de Cassandra

```
sudo nano /etc/cassandra/cassandra.yaml
```

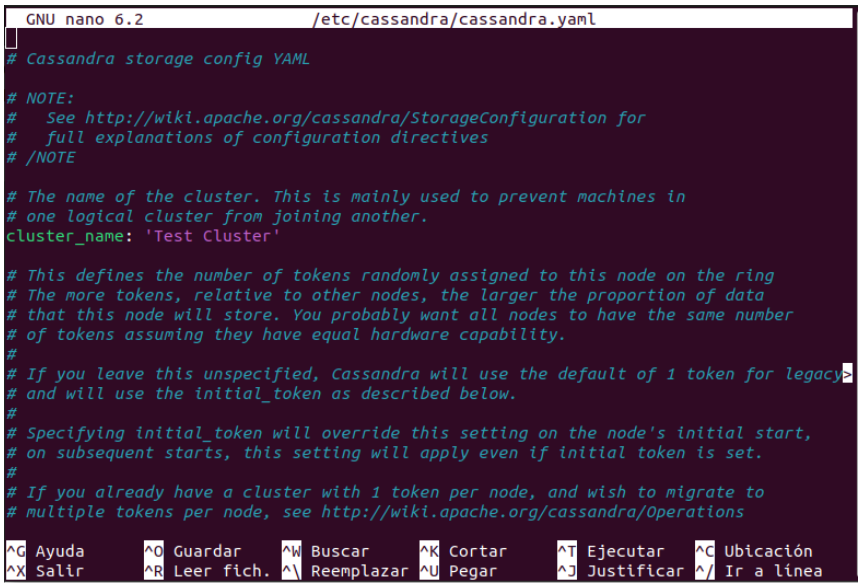

Ilustración 12: Fichero de configuración de Casandra.

Agregar el nombre del cluster previamente configurado y agregar la dirección IP de la red local en nuestro caso '10.0.2.15'.

- cluster_name: 'thp'
- listen_address: '10.0.2.15'
- rpc_address: '10.0.2.15'
- seed_provider
- hints_directory: '/var/lib/cassandra/hints'

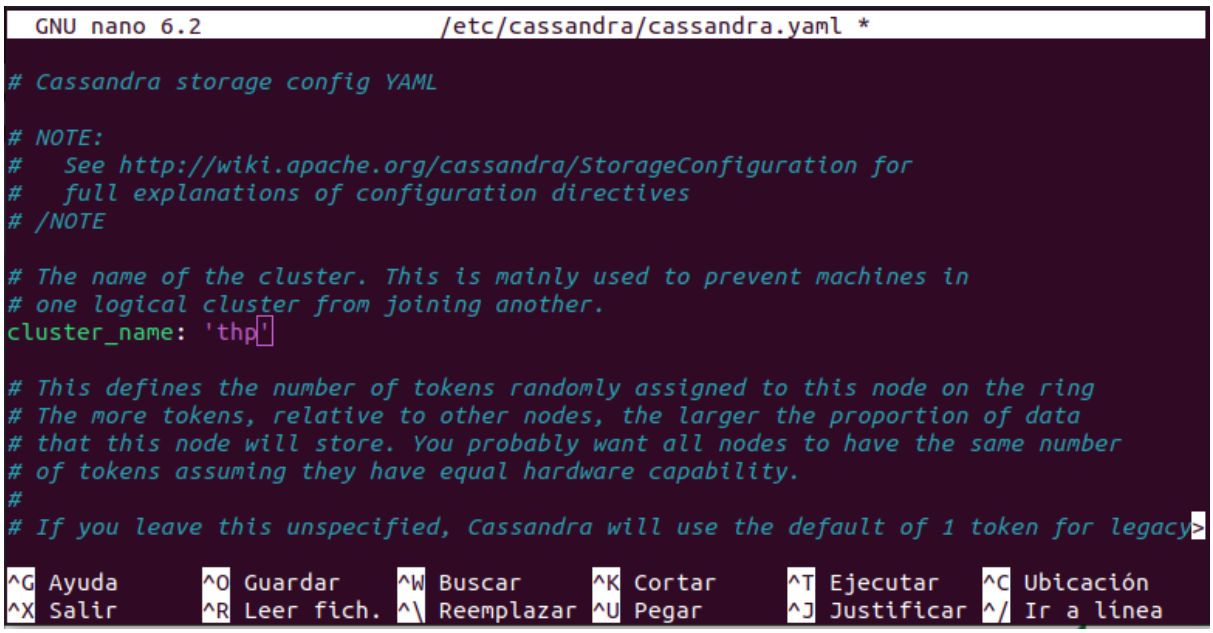

Ilustración 13: cluster_name.

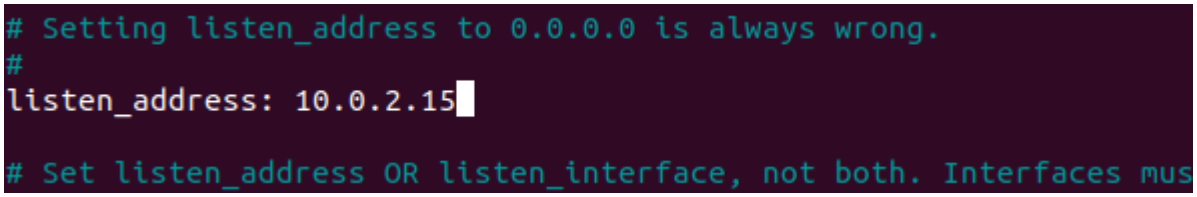

Ilustración 14: listen_address.

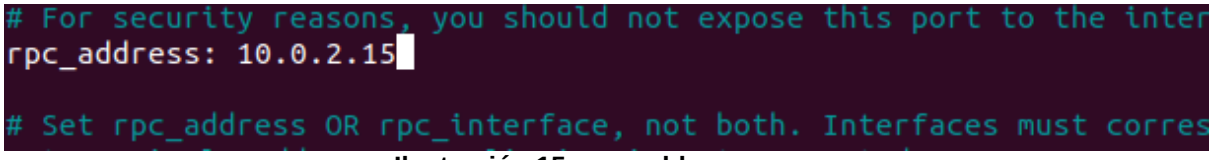

Ilustración 15: rpc_address.

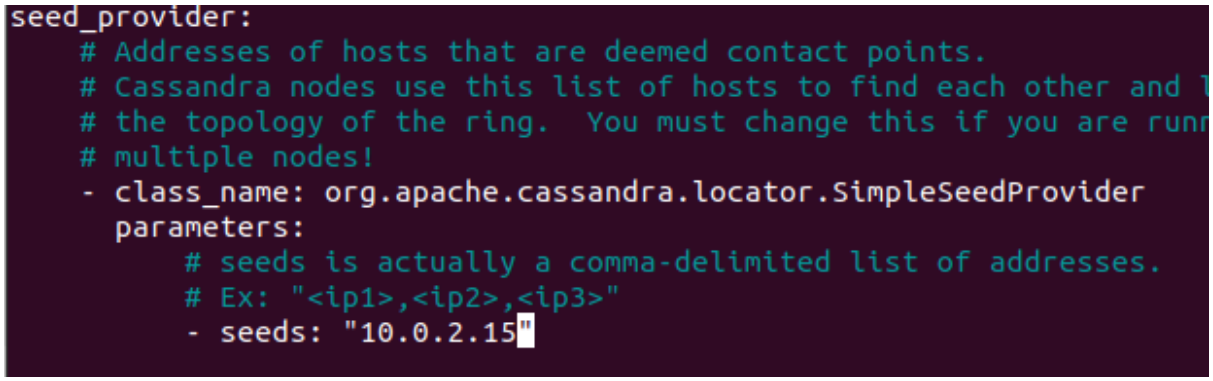

Ilustración 16: seed_provider.

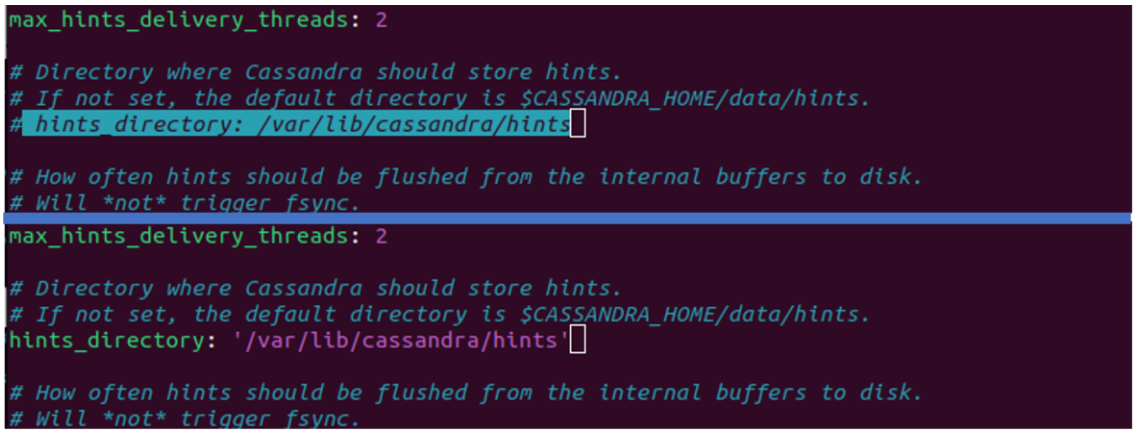

Ilustración 17: hints_directory.

Reiniciar Cassandra

```
systemctl daemon-reload
service cassandra restart
```

Por defecto, Cassandra escucha en los puertos 7000/tcp (inter-node), 9042/tcp (client). Comprobamos que escucha en el puerto indicado utilizando la herramienta net-tools.

```
netstat -an | grep 7000
```

```
root@marvin-VirtualBox:~# netstat -an | grep 7000
tcp        0      0 127.0.0.1:7000          0.0.0.0:*               ESCUCHAR
unix  3      [ ]         FLUJO      CONECTADO     27000
root@marvin-VirtualBox:~#
```

Ilustración 18: Comprobación de Cassandra.

Habilitar que Cassandra se inicie junto al sistema operativo.

```
systemctl enable cassandra
```

Instalar TheHive 4

```
curl https://raw.githubusercontent.com/TheHive-Project/TheHive/master/PGP-PUBLIC-KEY | sudo apt-key add -

echo 'deb https://deb.thehive-project.org release main' | sudo tee -a /etc/apt/sources.list.d/thehive-project.list
```

```
sudo apt update
sudo apt-get install thehive4 –y
sudo mkdir /opt/thp/thehive/index
sudo chown thehive:thehive -R /opt/thp/thehive/index
sudo mkdir -p /opt/thp/thehive/files
sudo chown thehive:thehive -R /opt/thp/thehive/files
```

Configuración de TheHive

```
sudo nano /etc/thehive/application.conf
```

- Hay que Incluir la Secret key
- Especificas la base de datos (Casandra / Elasticsearch)
- Añadir la dirección del sistema de ficheros: /opt/thp/thehive/files

```
## Include Play secret key
# More information on secret key at https://www.playframework.com/documentation/2.8.x/ApplicationSecret
include "/etc/thehive/secret.conf"
```

Ilustración 19: Incluir la Secret key.

```
db.janusgraph {
  storage {
    ## Cassandra configuration
    # More information at https://docs.janusgraph.org/basics/confi>
    backend: cql
    hostname: ["10.0.2.15"]
    # Cassandra authentication (if configured)
    cql {
      cluster-name: thp
      keyspace: thehive
    }
  }
```

Ilustración 20: configuración de base de datos.

```
## Attachment storage configuration
storage {
  ## Local filesystem
  provider: localfs
  localfs.location: /opt/thp/thehive/files

  ## Hadoop filesystem (HDFS)
  // provider: hdfs
  // hdfs {
  //   root: "hdfs://localhost:10000" # namenode server hostname
  //   location: "/thehive"           # location inside HDFS
  //   username: thehive              # file owner
  // }
```

Ilustración 21: Añadir la dirección del sistema de ficheros.

Inicializar TheHive

```
service thehive start
```

Comprobar el correcto funcionamiento escuchando en el puerto 9000.

```
tail -f /var/log/thehive/application.log
```

```
2023-04-08 10:47:17,257 [INFO] from play.core.server.AkkaHttpServer in main [|] Listening f
or HTTP on /0:0:0:0:0:0:0:0:9000
2023-04-08 10:47:56,474 [INFO] from org.thp.scalligraph.AccessLogFilter in application-akka
.actor.default-dispatcher-10 [00000001|] 127.0.0.1 GET / took 51ms and returned 308 0 bytes
2023-04-08 10:47:56,833 [INFO] from org.thp.scalligraph.AccessLogFilter in application-akka
.actor.default-dispatcher-10 [00000002|] 127.0.0.1 GET /index.html took 265ms and returned
200 1191 bytes
2023-04-08 10:49:31,360 [INFO] from org.thp.scalligraph.AccessLogFilter in application-akka
.actor.default-dispatcher-16 [00000003|] 10.0.2.15 GET / took 1ms and returned 308 0 bytes
2023-04-08 10:49:31,402 [INFO] from org.thp.scalligraph.AccessLogFilter in application-akka
.actor.default-dispatcher-16 [00000004|] 10.0.2.15 GET /index.html took 16ms and returned 2
00 1191 bytes
```

Ilustración 22: Comprobación de TheHive.

Entrando desde el navegador http://10.0.2.15:9000 saldrá la siguiente página
Credenciales por defecto login: admin@thehive.local / password: secret.
Posteriormente cambiamos la clave del adminuser por mayor seguridad.

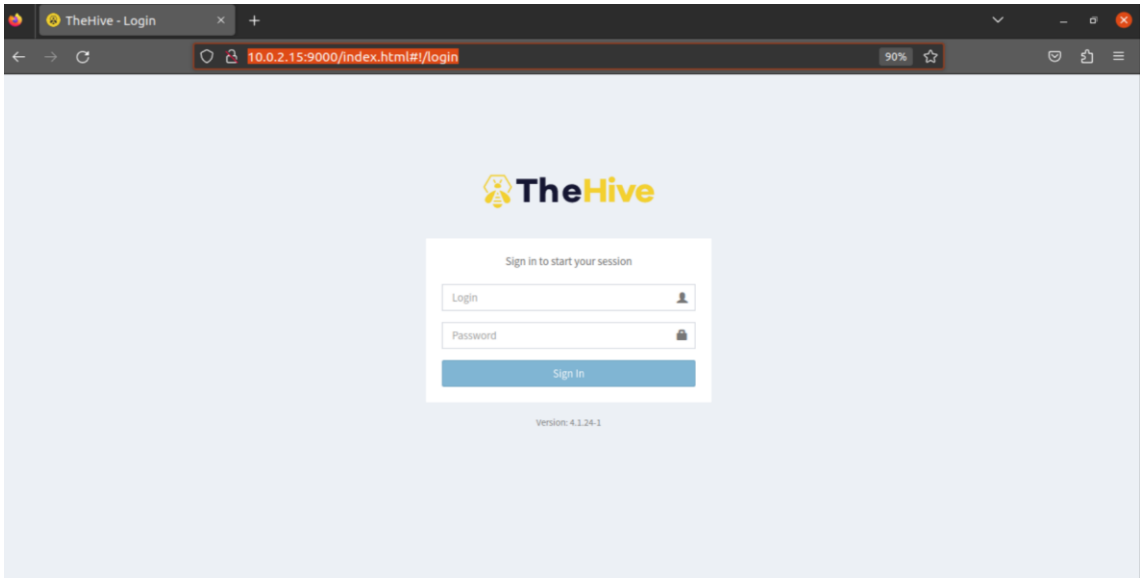

Ilustración 23: Página de inicio de sesión de TheHive.

4.2 Cortex

Actúa como gestor de analizadores del sistema. Requiere de disponer modo root para la correcta implantación de la herramienta. La versión instalada será Cortex 3.1.0, la cual soporta exclusivamente Elasticsearch 7.x (blog.agood.cloud, s. f.-a).

Deberemos tener acceso a las siguientes herramientas:

- python3 y python2 (mirar 10.1Instalación de python3 y python2)
- sudo apt-get install curl
- sudo apt install net-tools
- apt-get install lsb-release curl apt-transport-https zip unzip gnupg

Instalación de clúster de un solo nodo Elasticsearch 7.17.9

Cluster de un solo nodo se refiere a la configuración de un entorno de clusterización en el que solo hay un nodo o servidor involucrado. La configuración de un *single-node cluster* puede ser útil , para la ejecución de pruebas o entornos de menor escala. Permitiendo la configuración y gestión de un entorno de clusterización sin la necesidad de múltiples máquinas. Además, facilita la transición a un *cluster* de múltiples nodos en el futuro (Wazuh, 2023).

Agregar el repositorio de Elastic Stack

```
sudo apt install rpm
```

Importar la clave GPG y Agregar el repositorio.

```
curl -s https://artifacts.elastic.co/GPG-KEY-elasticsearch | gpg --no-default-keyring --keyring gnupg-ring:/usr/share/keyrings/elasticsearch.gpg --import && chmod 644 /usr/share/keyrings/elasticsearch.gpg

echo "deb [signed-by=/usr/share/keyrings/elasticsearch.gpg] https://artifacts.elastic.co/packages/7.x/apt stable main" | tee /etc/apt/sources.list.d/elastic-7.x.list
```

Actualizar la información del paquete

```
apt-get update
```

Instalación y configuración de Elasticsearch

```
apt-get install elasticsearch=7.17.9
nano /etc/elasticsearch/elasticsearch.yml
```

Hay añadir la IP del servidor (en nuestro caso localhost) e incrementar q_size para mejorar el rendimiento:

- cluster.name: hive
- node.name: node-1 (single node)
- network.host: 10.0.2.15
- cluster.initial_master_nodes: [“node-1”]
- add: thread_pool.search.queue_size: 100000

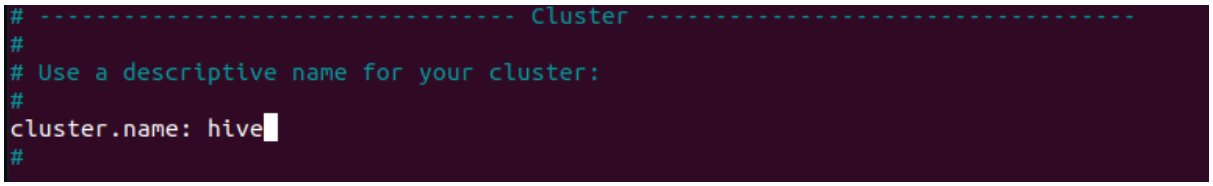

Ilustración 24: cluster.name.

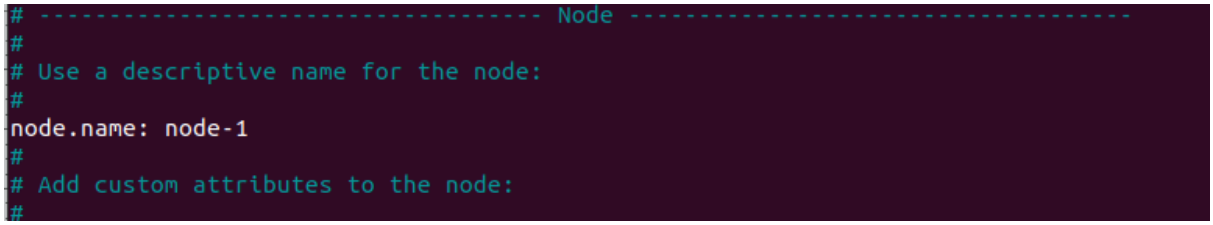

Ilustración 25: node.name.

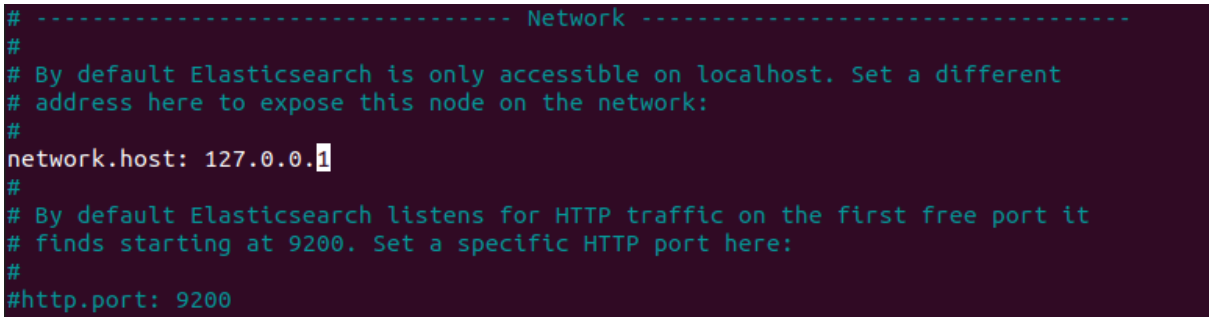

Ilustración 26: network.host.

```
# ---------------------------------- Discovery -----------------------------------
#
# Pass an initial list of hosts to perform discovery when this node is started:
# The default list of hosts is ["127.0.0.1", "[::1]"]
#
#discovery.seed_hosts: ["host1", "host2"]
#
# Bootstrap the cluster using an initial set of master-eligible nodes:
#
cluster.initial_master_nodes: ["node-1"]
#
```

Ilustración 27: cluster.initial_master_nodes.

```
# ---------------------------------- Various -----------------------------------
#
# Require explicit names when deleting indices:
#
#action.destructive_requires_name: true
thread_pool.search.queue_size: 100000
#
```

Ilustración 28: thread_pool.search.queue_size.

Inicializar Elasticsearch

```
sudo systemctl start elasticsearch
sudo systemctl enable elasticsearch
```

```
root@marvin-VirtualBox:~# sudo systemctl enable elasticsearch
Synchronizing state of elasticsearch.service with SysV service script with /lib/systemd/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable elasticsearch
Created symlink /etc/systemd/system/multi-user.target.wants/elasticsearch.service → /lib/systemd/system/elasticsearch.se
rvice.
root@marvin-VirtualBox:~#
```

Ilustración 29: Habilitar Elasticsearch.

Comprobar que funciona correctamente escuchando en el puerto 9200.

```
tail -f /var/log/elasticsearch/hive.log
netstat -an | grep 9200
```

```
tcp6       0      0 ::1:9200                :::*                    ESCUCHAR
tcp6       0      0 127.0.0.1:9200          :::*                    ESCUCHAR
root@marvin-VirtualBox:~#
```

Ilustración 30: Comprobación de Elasticsearch.

Instalación de Cortex 3.1.0+

Importar la clave y agregar el repositorio.

```
curl https://raw.githubusercontent.com/TheHive-Project/TheHive/master/PGP-PUBLIC-KEY | sudo apt-key add -

echo 'deb https://deb.thehive-project.org release main' | sudo tee -a /etc/apt/sources.list.d/thehive-project.list
```

Actualizar la información del paquete.

```
apt-get update
```

Instalamos Cortex.

```
sudo apt install cortex
systemctl status cortex
```

```
marvin@marvin-VirtualBox:~$ systemctl status cortex
● cortex.service - cortex
     Loaded: loaded (/etc/systemd/system/cortex.service; disabled; vendor prese>
     Active: inactive (dead)
       Docs: https://thehive-project.org
```

Ilustración 31: Systemctl status Cortex.

Generar la clave secreta.

```
cd /etc/cortex
```

```
sudo mkdir /etc/cortex
(cat << _EOF_
# Secret key
# ~~~~~
# The secret key is used to secure cryptographics functions.
# If you deploy your application to several instances be sure to use the same key!
play.http.secret.key="$(cat /dev/urandom | tr -dc 'a-zA-Z0-9' | fold -w 64 | head -n 1)"
_EOF_
) | sudo tee -a /etc/cortex/application.conf
```

```
marvin@marvin-VirtualBox:/etc/cortex$ (cat << _EOF_
> # Secret key
> # ~~~~~
> # The secret key is used to secure cryptographics functions.
> # If you deploy your application to several instances be sure to use the same
key!
> play.http.secret.key="$(cat /dev/urandom | tr -dc 'a-zA-Z0-9' | fold -w 64 | h
ead -n 1)"
> _EOF_
> ) | sudo tee -a /etc/cortex/application.conf
# Secret key
# ~~~~~
# The secret key is used to secure cryptographics functions.
# If you deploy your application to several instances be sure to use the same ke
y!
play.http.secret.key="Jb27FlgtDnCMy3jSsKNcxDL7c7XIGdf03sKakFcLogXxawP94dCnE7U0lC
bywcSK"
marvin@marvin-VirtualBox:/etc/cortex$
```

Ilustración 32: Generacion de clave de Cortex.

Con esto se añade la clave secreta al final del archivo /etc/cortex/application.conf. En caso contrario se añadirá manualmente (nano application.conf).

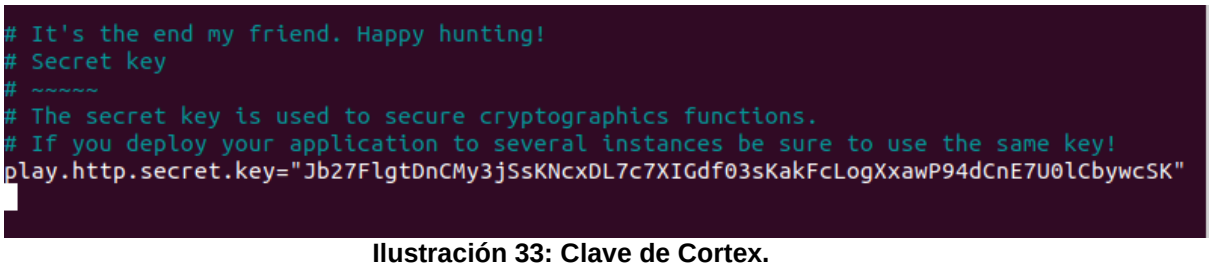

Ilustración 33: Clave de Cortex.

En caso de utilizar un servidor distinto para elasticsearch se indica en el archivo.

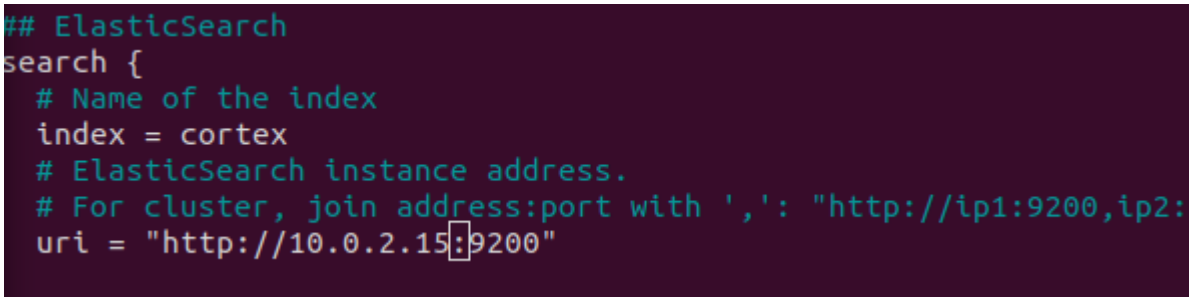

Ilustración 34: url de Elasticsearch.

Inicializar Cortex.

```
sudo systemctl start cortex
sudo systemctl enable cortex
```

Comprobar que funciona correctamente escuchando en el puerto 9001.

```
tail -f /var/log/cortex/hive.log
netstat -an | grep 9001
```

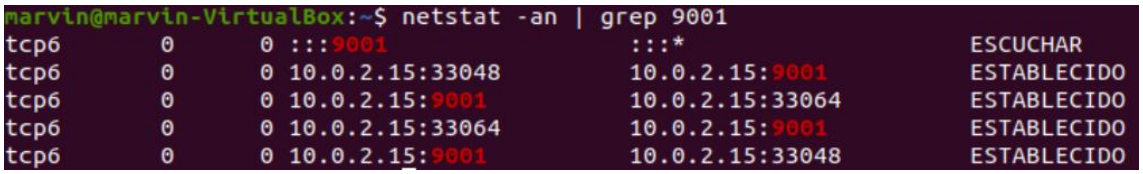

Ilustración 35: Comprobación de Cortex.

Entrando desde el navegador http://10.0.2.15:9001 saldrá la siguiente página Credenciales por defecto login: cortexadmin / name: cortex admin / password: cortex12. Posteriormente, cambiamos la clave del adminuser para mayor seguridad

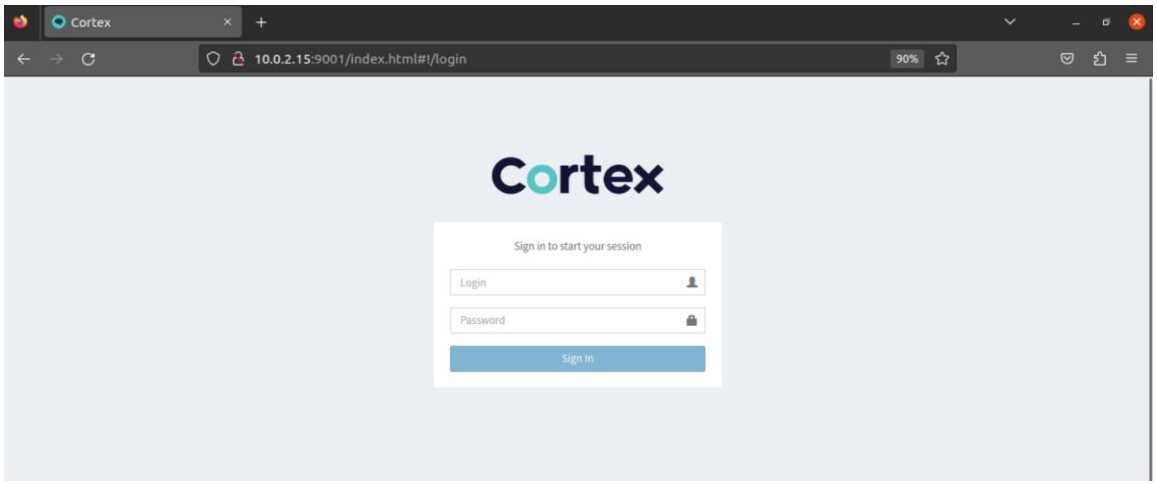

Ilustración 36: Página de inicio de sesión de Cortex.

Agregar analizadores

Clonar los Cortex-Analyzers en el directorio específico (/opt/cortex/).

```
cd /opt/cortex/
sudo apt install git –y
sudo git clone
```

Instalar todos los requisitos.

```
cd Cortex-Analyzers/analyzers/
cd ..//..
for I in $(find Cortex-Analyzers -name 'requirements.txt'); do sudo -H pip2 install -r $I; done && \
for I in $(find Cortex-Analyzers -name 'requirements.txt'); do sudo -H pip3 install -r $I || true; done
```

Configurar Cortex y reiniciar.

```
nano /etc/cortex/application.conf
```

Indicar la ruta a los ficheros de los analizadores: /opt/cortex/Cortex-Analyzers/analyzers.

```
## ANALYZERS
#
analyzer {
  # analyzer location
  # url can be point to:
  # - directory where analyzers are installed
  # - json file containing the list of analyzer descriptions
  urls = [
    #"https://download.thehive-project.org/analyzers.json"
    "/opt/cortex/Cortex-Analyzers/analyzers"
  ]

  # Sane defaults. Do not change unless you know what you are doing.
  fork-join-executor {
    # Min number of threads available for analysis.
    parallelism-min = 2
    # Parallelism (threads) ... ceil(available processors * factor).
    parallelism-factor = 2.0
    # Max number of threads available for analysis.
    parallelism-max = 4
```

Ilustración 37: Configuración de la ruta de los analizadores.

```
systemctl restart cortex
tail -f /var/log/cortex/application.log
```

```
2023-04-09 22:01:16,079 [INFO] from com.sksamuel.elastic4s.http.Java
5:9200
2023-04-09 22:01:17,311 [INFO] from org.thp.cortex.services.WorkerSr
rker list:

        RiskIQ_Certificates 1.0
        IPVoid 1.0
        SEKOIAIntelligenceCenter_Indicators 1.0
        RiskIQ_Projects 1.0
        Shuffle 1.0
        SEKOIAIntelligenceCenter_Context 1.0
        RiskIQ_Cookies 1.0
        HIBP_Query 2.0
        CheckPoint_Unlock 1.0
        DNSSinkhole 1.0
        DomainToolsIris_Investigate 1.0
        ThreatMiner 1.0
        MSDefender-UnisolateMachine 1.0
        Autofocus_SearchJSON 1.0
        MSDefender-PushIOC-Block 1.0
        DomainTools_Reputation 2.0
        PaloAltoCortexXDR_isolate 1.0
        AMPforEndpoints_SCDAdd 1.0
        MaxMind_GeoIP 4.0
        Crowdstrike_Falcon_Custom_IOC_API 1.0
```

Ilustración 38: Comprobación de Analizadores.

Crear una nueva organización y agregar un usuario

Entrando desde a web creamos la organización con una cuenta de usuario asociado. En nuestro caso será:

organización: Domestic_SOC
log: domestic@soc pw: *****

log: domestic_user2@soc pw: *******

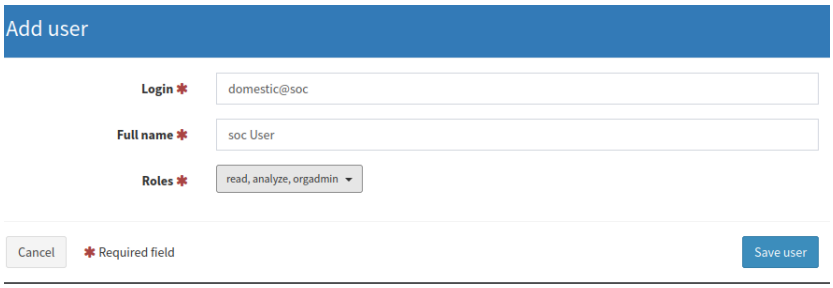

Ilustración 39: Agregar un usuario (Cortex).

4.3 Integración de Cortex-TheHive

Consiste en generar un acceso directo desde TheHive usando la API de Cortex.

1. Entramos en Cortex con el usuario domestic@soc
2. Creamos un usuario que interactúe con la API

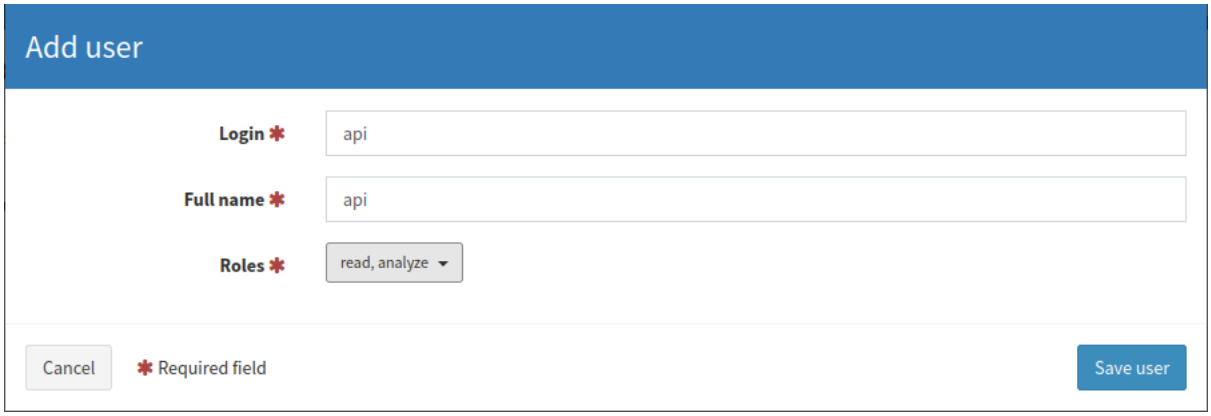

Ilustración 40: Usuario API key.

3. Una vez generado, crearemos la API Key

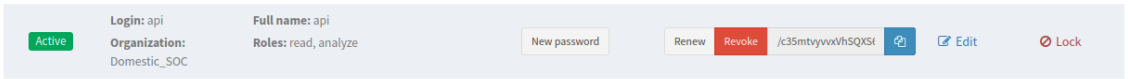

Ilustración 41: API Key (Cortex).

4. Modificamos el archivo etc/thehive/application.conf

```
sudo nano /etc/thehive/application.conf
```

Añadir:

- nombre (“CORTEX1”) y
- URL del servidor donde está localizado Cortex (10.0.2.15:9001)

- API Key que acabamos de crear

```
## CORTEX configuration
# More information at https://github.com/TheHive-Project/TheHiveDocs/TheHive4/A>
# Enable Cortex connector
play.modules.enabled += org.thp.thehive.connector.cortex.CortexModule
cortex {
 servers: [
   {
     name: "CORTEX1"                # Cortex name
     url: "http://10.0.2.15:9001" # URL of Cortex instance
     auth {
       type: "bearer"
       key: "ka+bClctHfPVcw7P5aLU773bjb/jKydM"                # Cortex API key
     }
     wsConfig {}                  # HTTP client configuration (SSL and proxy)
   }
 ]
}
```

Ilustración 42: Configuración de la API en el fichero.

Reiniciar TheHive.

```
systemctl restart thehive
tail -f /var/log/thehive/application.log
```

Entrando en TheHive desde el navegador se puede corroborar la conexión con Cortex desde User>About (señalizando un OK en la conexión).

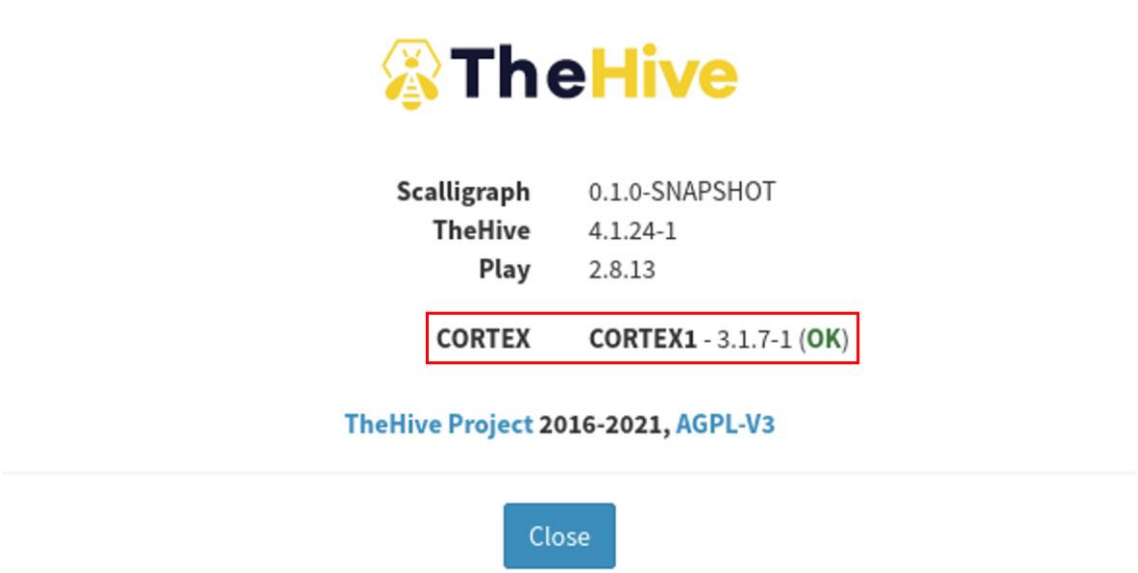

Ilustración 43: Validación de la integración TheHive-Cortex.

4.4 MISP

Permitirá habilitar la herramienta de análisis para correlacionar y enriquecer los datos compartidos.

Para la configuración de MISP **NO** se realiza en modo superusuario, dado que se creara un misp user (blog.agood.cloud, s. f.-b).

Instalación de MISP

```
wget --no-cache -O /tmp/INSTALL.sh
bash /tmp/INSTALL.sh
```

Seleccionar la instalación completa de MISP

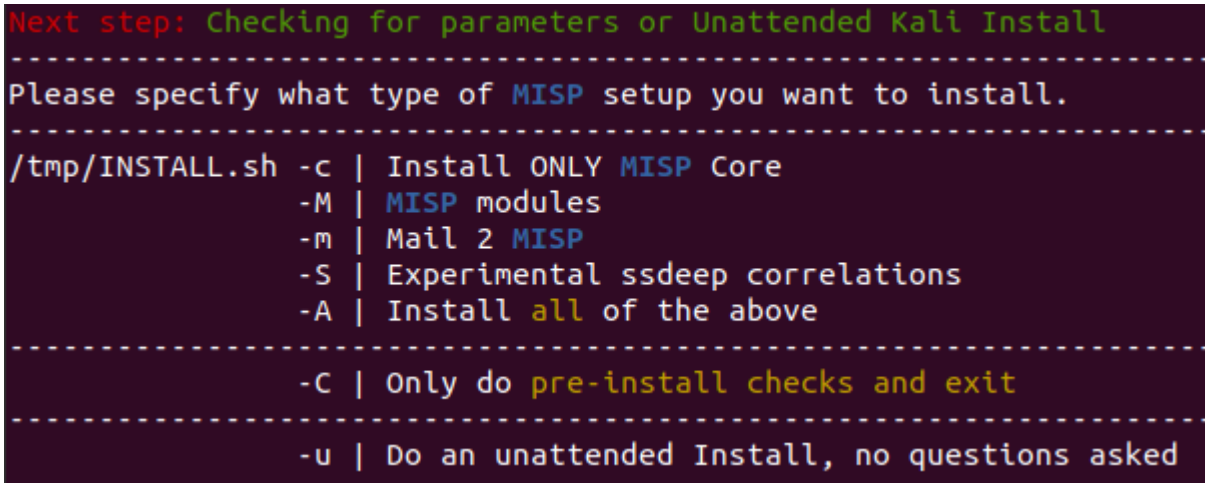

Ilustración 44: Opciones de configuración de MISP.

```
bash /tmp/INSTALL.sh -A
```

Al final de la instalación se muestra la contraseña y el usuario por defecto.

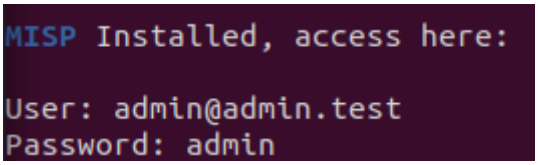

Ilustración 45: Usuarion y Contraseña predefinida de MISP.

Acceso a MISP a través del navegador http://10.0.2.15
User: admin@admin.test. Cambiamos la clave del adminuser para mayor seguridad.

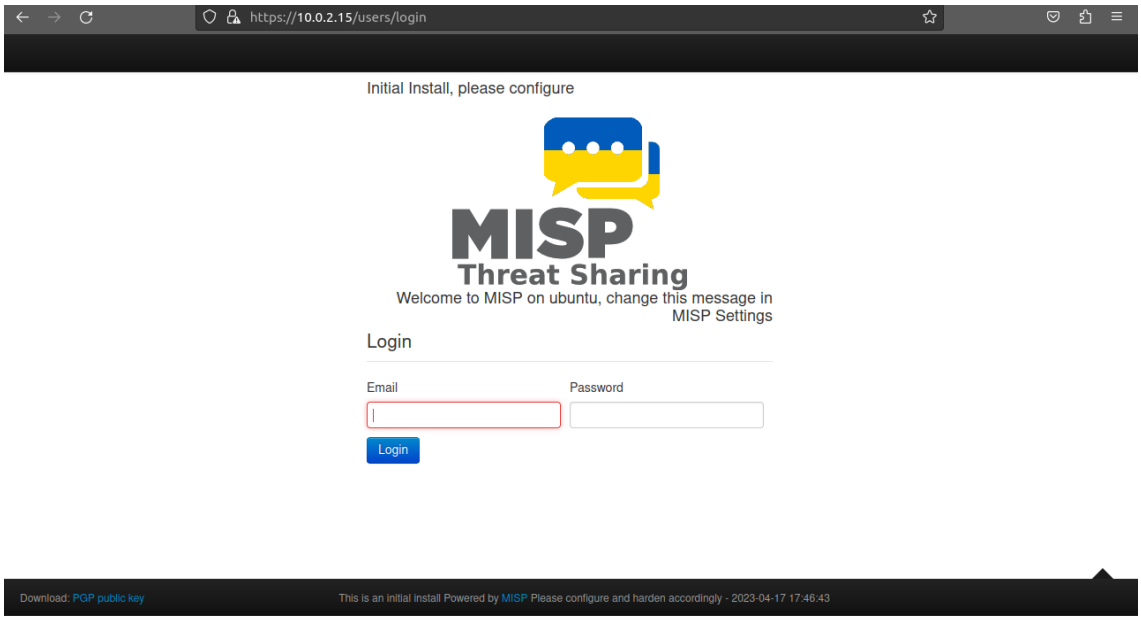

Ilustración 46: Página de inicio de sesión de MISP.

Integrar MISP con Cortex

Primero debemos de obtener la API Key, la cual adquirimos creando una nueva Administration > List > Auth Keys. En este paso se puede crear una capa extra de seguridad especificando las IPs que solo puede acceder uso de esta API Key.

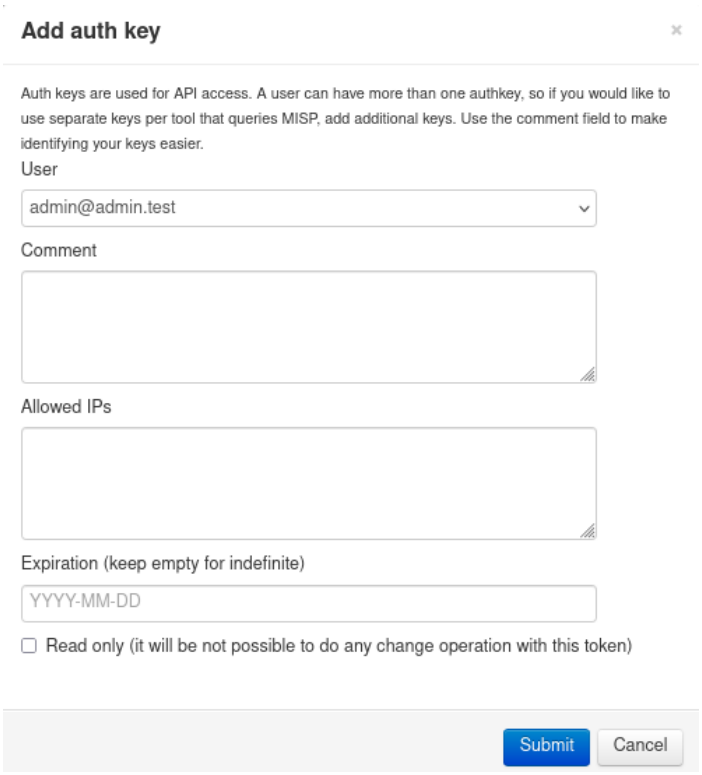

Ilustración 47: Generación de la API Key de MISP.

Al ejecutar ‘Submit’ aparecerá una ventana con la Key. Es solo en este momento donde podremos ver la API Key al completo, por lo que será indispensable copiarla para su posterior uso.

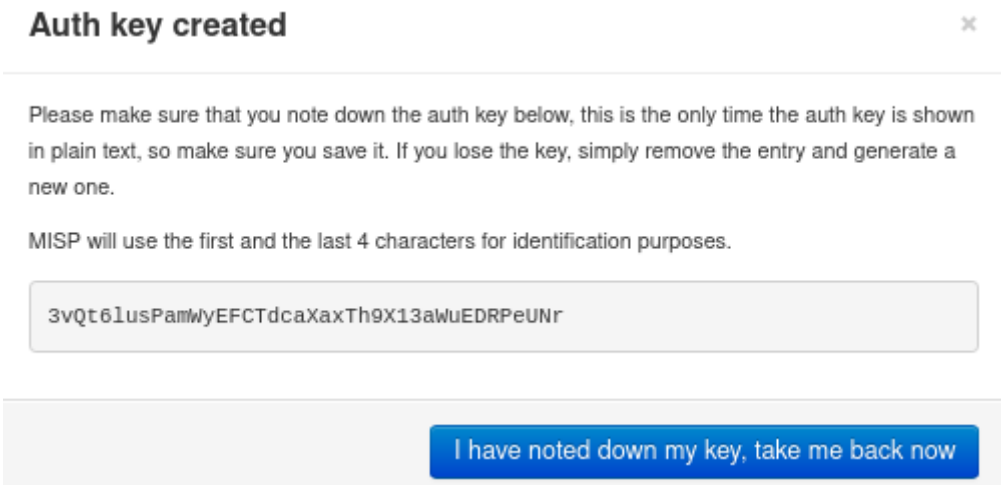

Ilustración 48: API KEY de MISP.

Una vez copiada la clave, procedemos a añadir MISP a Cortex como cualquier otro analizador. Donde tendremos que pegar esta misma clave e indicar que no verifique la credencial dado que se realizara desde nuestro local host, la cual no cuenta con los criterios de seguridad (https).

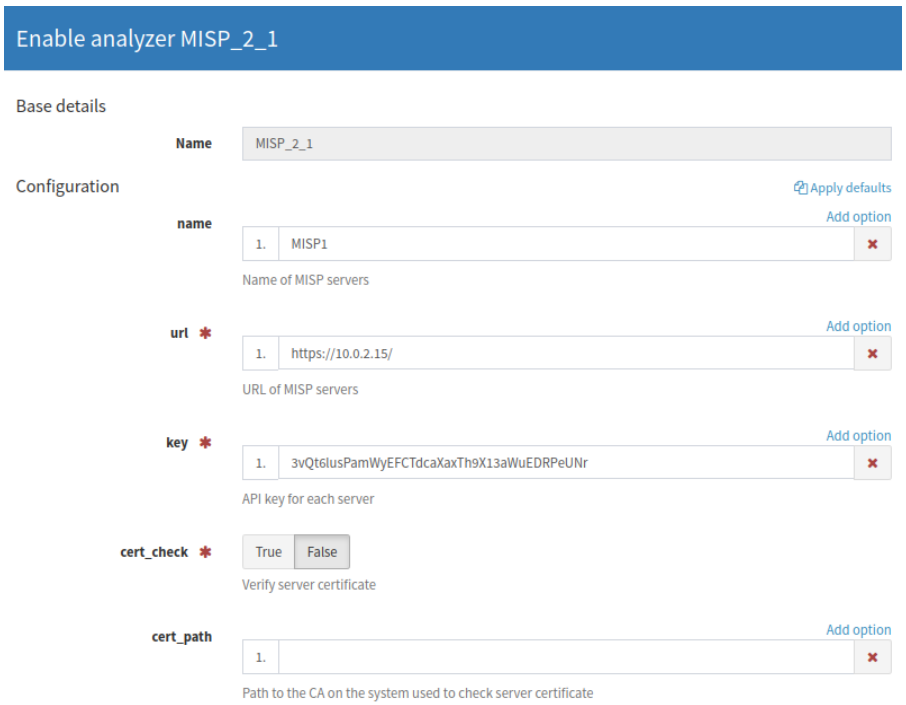

Ilustración 49: Adición del analizador MISP en Cortex.

4.5 Suricata

Instalaremos la herramienta que efectuará las acciones de IDS/IPS (Suricata, s. f.-a, s. f.-b).

Requeriremos e los permisos de administrador root.

Instalación de Suaricata

Comenzamos añadiendo el repositorio de Suricata

```
add-apt-repository ppa:oisf/suricata-stable
```

Actualizar los registros

```
apt-get update
```

Instalamos Suricata y jq, el cual, permitirá mostrar los registros de manera más visual en la terminal.

```
apt install suricata jq
```

Configuración de ficheros

```
nano /etc/suricata/suricata.yaml
```

- **HOME_NET**: representa todas las redes que va a considerar como redes locales,
- **EXTERNAL_NET**: el resto que se ajustan como externas
- **default-log-dir**: indica la dirección donde se almacenan todos los registros log “/var/log/suricata/”
- **interface**: indica sobre la interfaz donde va a realizar Suricata el análisis

Para ver la interfaz (mirar 10.2 Interfaz de trabajo), en este caso el puesto de escucha es enp0s3.

```
vars:
  # more specific is better for alert accuracy and performance
  address-groups:
    HOME_NET: "[192.168.0.0/16,10.0.0.0/8,172.16.0.0/12]"
    #HOME_NET: "[192.168.0.0/16]"
    #HOME_NET: "[10.0.0.0/8]"
    #HOME_NET: "[172.16.0.0/12]"
    #HOME_NET: "any"

    EXTERNAL_NET: "!$HOME_NET"
    #EXTERNAL_NET: "any"
```

Ilustración 50: HOME_NET y EXTERNAL_NET.

```
## Step 2: Select outputs to enable
##

# The default logging directory.  Any log or output file will be
# placed here if it's not specified with a full path name. This can be
# overridden with the -l command line parameter.
default-log-dir: /var/log/suricata/
```

Ilustración 51: default-log-dir.

```
## Step 3: Configure common capture settings
##
## See "Advanced Capture Options" below for more options, including Netmap
## and PF_RING.
##

# Linux high speed capture support
af-packet:
  - interface: enp0s3
    # Number of receive threads. "auto" uses the number of cores
```

Ilustración 52: interface.

Creación del fichero de reglas

```
touch /etc/suricata/rules/my2.rules
nano /etc/suricata/suricata.yam
```

Indiciamos la dirección del fichero de reglas que acabamos de crear (rule-path / rules-files).

```
## Configure Suricata to load Suricata-Update managed rules.
##

default-rule-path: /etc/suricata/rules

rule-files:
  - my2.rules
```

Ilustración 53: Configurar la ruta al fichero de reglas.

Crear la regla

```
nano /etc/suricata/rules/my.rules

alert icmp any any -> any any (msg: “ICMP Packet found”; sid:2000001; rev:1;)
```

Una vez añadida la regla reiniciamos el servidor.

```
service suricata restart
```

Para la comprobación de la correcta escritura de la regla se aplica el siguiente comando:

```
suricata -c /etc/suricata/suricata.yaml -s /etc/suricata/rules/my.rules -i enp0s3
```

- -c indica el fichero de configuración
- -s indica el fichero de reglas
- -i indica la interfaz a analizar

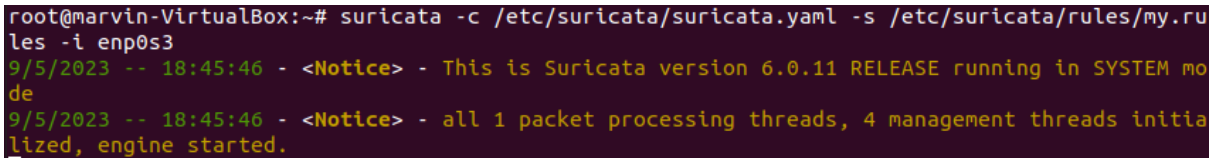

Ilustración 54: Comprobación de la correcta escritura de la regla.

En una terminal nueva comprobaremos si saltan las alertas en el fichero de registro fast.log con el siguiente comando:

```
tail -f /var/log/suricata/fast.log
```

Para simular casos de ataque más sofisticados se puede hacer uso de archivos ".pcap", los cuales cuentan con tráfico artificial con uno o varios ataques registrados. Esto permite comprobar las reglas directamente sobre el tráfico generado en un fichero en concreto.

Para realizar estas pruebas se utiliza el siguiente comando:

```
sudo suricata -r /home/marvin/5d2f3bb8fc9e4610329df081e8ea68c7.pcap -c /etc/suricata//suricata.yaml -s /etc/suricata/rules/my.rules -l /home/marvin
```

- -r: indica el fichero desde el cual realizar el análisis
- -c: indica el archivo de configuración a utilizar
- -s indica el fichero de reglas
- -l: indica el directorio del fichero

Para visualizar las alertas en este caso se hace desde la siguiente dirección:

```
cat /home/marvin/fast.log
```

```
root@marvin-VirtualBox:~# cat /home/marvin/fast.log
02/18/2017-01:41:16.802873  [**] [1:20001:1] Possible PayPal phishing detected [**] [Classification:
(null)] [Priority: 3] {UDP} 192.168.1.25:52508 -> 8.8.8.8:53
02/18/2017-01:41:16.862850  [**] [1:20001:1] Possible PayPal phishing detected [**] [Classification:
(null)] [Priority: 3] {UDP} 192.168.1.25:50187 -> 8.8.8.8:53
02/18/2017-01:41:16.925996  [**] [1:20001:1] Possible PayPal phishing detected [**] [Classification:
(null)] [Priority: 3] {UDP} 192.168.1.25:50208 -> 8.8.8.8:53
02/18/2017-01:41:19.289960  [**] [1:20001:1] Possible PayPal phishing detected [**] [Classification:
(null)] [Priority: 3] {UDP} 192.168.1.25:58156 -> 8.8.8.8:53
02/18/2017-01:41:19.361547  [**] [1:20001:1] Possible PayPal phishing detected [**] [Classification:
(null)] [Priority: 3] {UDP} 192.168.1.25:58371 -> 8.8.8.8:53
02/18/2017-01:41:16.802873  [**] [1:20001:1] Possible PayPal phishing detected [**] [Classification:
(null)] [Priority: 3] {UDP} 192.168.1.25:52508 -> 8.8.8.8:53
02/18/2017-01:41:16.862850  [**] [1:20001:1] Possible PayPal phishing detected [**] [Classification:
(null)] [Priority: 3] {UDP} 192.168.1.25:50187 -> 8.8.8.8:53
02/18/2017-01:41:16.925996  [**] [1:20001:1] Possible PayPal phishing detected [**] [Classification:
(null)] [Priority: 3] {UDP} 192.168.1.25:50208 -> 8.8.8.8:53
02/18/2017-01:41:19.289960  [**] [1:20001:1] Possible PayPal phishing detected [**] [Classification:
(null)] [Priority: 3] {UDP} 192.168.1.25:58156 -> 8.8.8.8:53
02/18/2017-01:41:19.361547  [**] [1:20001:1] Possible PayPal phishing detected [**] [Classification:
(null)] [Priority: 3] {UDP} 192.168.1.25:58371 -> 8.8.8.8:53
root@marvin-VirtualBox:~#
```

Ilustración 55: visualización de las alertas desde un fichero.

Suricata devuelve las capturas en formato JASON cat /var/log/suricata/eve.json. Sin embargo, es difícil de leer, por ello se utiliza la extensión jq la cual lo transforma en un formato más claro.

Ilustración 56: Captura de los registros en JASON.

```
tail /var/log/suricata/eve.json | jq '.'
```

```
"timestamp": "2023-05-09T20:01:00.359069+0200",
"flow_id": 1183081095292062,
"in_iface": "enp0s3",
"event_type": "flow",
"src_ip": "10.0.2.15",
"src_port": 36317,
"dest_ip": "192.168.0.1",
"dest_port": 53,
"proto": "UDP",
"app_proto": "dns",
"flow": {
  "pkts_toserver": 1,
  "pkts_toclient": 1,
  "bytes_toserver": 86,
  "bytes_toclient": 86,
  "start": "2023-05-09T19:52:44.713886+0200",
  "end": "2023-05-09T19:52:44.721569+0200",
  "age": 0,
  "state": "established",
  "reason": "timeout",
  "alerted": false
```


Ilustración 57: Captura de los registros transformados.

Configurar Suricata como IPS

Esto permitirá a suricata bloquear el tráfico basándose en las reglas descritas. Hay que comprobar que NFQueue support este habilitado.

```
suricata --build-info
```

```
Suricata Configuration:
  AF_PACKET support:                       yes
  eBPF support:                            no
  XDP support:                             no
  PF_RING support:                         no
  NFQueue support:                         yes
  NFLOG support:                           no
  IPFW support:                            no
  Netmap support:                          no  using new api: no
  DAG enabled:                             no
  Napatech enabled:                        no
  WinDivert enabled:                       no
```

Ilustración 58: Comprobación que NFQueue support este habilitado.

Una vez revisado, se tendrá que derivar todo el tráfico desde el cortafuegos de Linux que viene por defecto.

- BORRA TODAS LAS REGLAS DE LA CADENA

```
sudo iptables –F
```

- MUESTRA LAS REGLAS ACTUALES

```
sudo iptables –vnL
```

- ANALIZA EL TRÁFICO QUE PASA POR EL ORDENAR

```
sudo iptables -I FORWARD -j NFQUEUE
```

- ANALIZA EL TRÁFICO QUE GENERA EL ORDENADOR

```
sudo iptables -I INPUT -j NFQUEUE
sudo iptables -I OUTPUT -j NFQUEUE
```

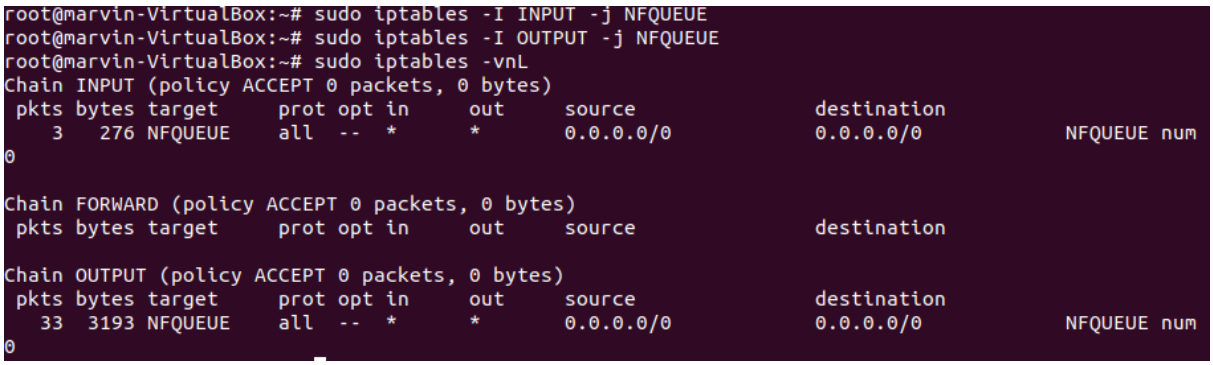

Ilustración 59: Visualización de las reglas configuradas.

Para la comprobación de la correcta escritura de la regla se aplica el siguiente comando:

```
sudo suricata -c /etc/suricata//suricata.yaml -s /etc/suricata/rules/my.rules -q 0
```

El comando es similar al realizado sobre una interfaz, pero modificando esta por -q, que indica el número de cola que queremos que trabaje, en este caso, para simplificar la estructura indicamos 0.

Instalación de las reglas emergin threats

Descargando las reglas de emergin threats nos permitirá evitar tener que crearlas por nuestra cuenta, ya que este cuenta con varias de estas reglas, que a su vez se actualizan varias veces al día.

```
sudo suricata-update
sudo nano /var/lib/suricata/rules/suricata.rules
```

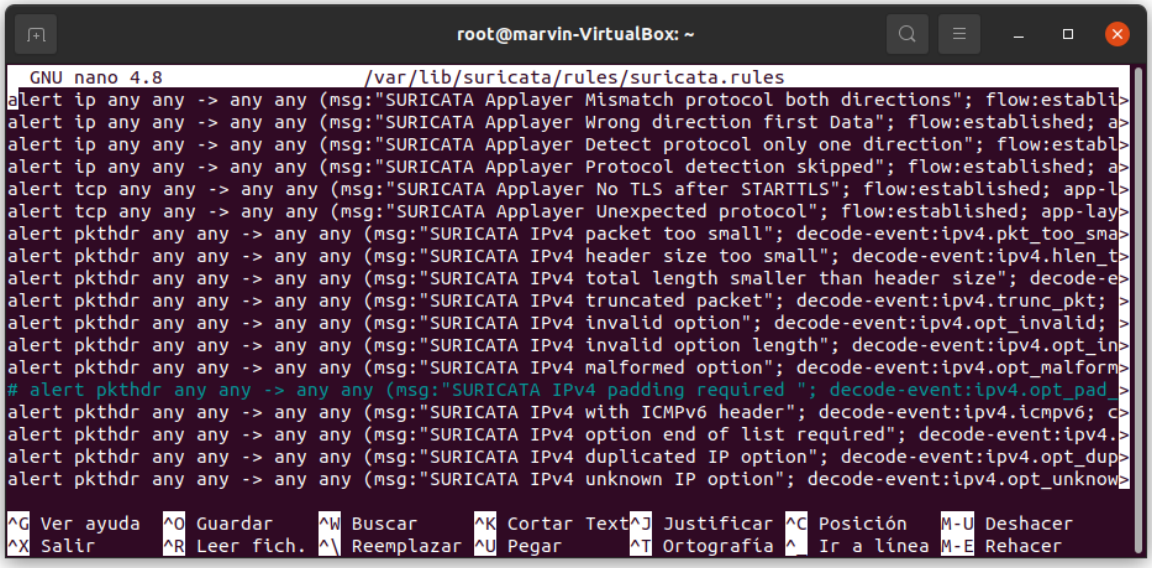

Ilustración 60: Archivo de reglas de emergin threats.

Actualizamos el directorio del archivo de rules en la configuración de suricata.

```
nano /etc/suricata/suricata.yaml
```

```
## Configure Suricata to load Suricata-Update managed rules.
##

default-rule-path: /var/lib/suricata/rules/suricata.rules
```

Ilustración 61: Configuración de la ruta de reglas de emergin threats.

5 Resultados

Este capítulo se divide en dos secciones. En la primera sección se muestra el modo de uso de los módulos independientemente. En la segunda sección se muestra el modo de uso de los módulos en conjunto, conformando de este modo el SOC doméstico.

5.1 Modo de uso de los Módulos

Módulo de detección: Este módulo está compuesto íntegramente por la herramienta IDS/IPS Suricata.

Como pudimos ver en la instalación, Suricata permite añadir reglas propias. Esto permite ajustar la detección de incidentes en función de las necesidades específicas del usuario. Para ejemplificar esto, se realizaron varias reglas atacando diversos casos de uso:

1. **Alerta de conexión con Facebook**

alert tcp any any -> any any (msg:"Facebook access detected and dropped"; content:"facebook"; sid:1000001;)

En el caso de que se quiera bloquear la conexión con Facebook, se sustituye “alert” por “drop”, del mismo modo, es necesario activar los permisos IPS, en caso contrario, simplemente se limitara a lanzar la aleta de conexión “wDrop”.

drop tcp any any -> any any (msg:"Facebook access detected and dropped"; content:"facebook"; sid:1000001;)

El acceso a facebook.com o facebook.es lanzan la alerta del mismo modo, y los bloquea en el modo IPS.

```
marvin@marvin-VirtualBox:~$ tail -f /var/log/suricata/fast.log
05/09/2023-19:02:40.443190  [wDrop] [**] [1:1000001:0] Facebook access detected and dropp
ed [**] [Classification: (null)] [Priority: 3] {TCP} 10.0.2.15:52808 -> 157.240.5.35:80
05/09/2023-19:06:50.710548  [wDrop] [**] [1:1000001:0] Facebook access detected and dropp
ed [**] [Classification: (null)] [Priority: 3] {TCP} 10.0.2.15:33526 -> 157.240.5.12:80
05/09/2023-19:09:15.358572  [wDrop] [**] [1:1000001:0] Facebook access detected and dropp
ed [**] [Classification: (null)] [Priority: 3] {TCP} 10.0.2.15:49678 -> 31.13.83.8:443
05/09/2023-19:09:17.060687  [wDrop] [**] [1:1000001:0] Facebook access detected and dropp
ed [**] [Classification: (null)] [Priority: 3] {TCP} 10.0.2.15:39822 -> 157.240.5.35:443
05/09/2023-19:09:19.166468  [wDrop] [**] [1:1000001:0] Facebook access detected and dropp
ed [**] [Classification: (null)] [Priority: 3] {TCP} 10.0.2.15:39826 -> 157.240.5.35:443
```

Ilustración 62: Registro de acceso a Facebook modo IPS desactivado.

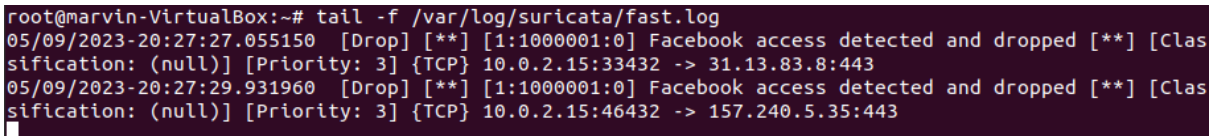

Ilustración 63: Registro de acceso a Facebook modo IPS activado.

2. Alerta de petición GET mediante el protocolo http

alert http $HOME_NET any -> $EXTERNAL_NET 80 (msg:"HTTP GET request"; flow: established, to_server; content:"GET"; http_method; sid:9999; rev: 2;)

Tras realizar una solicitud HTTP a la URL (www.uoc.edu) y mostrar tanto la respuesta del servidor como los encabezados de la respuesta desde el terminal con el siguiente comando curl -i www.uoc.edu. Podemos ver la siguiente alerta:

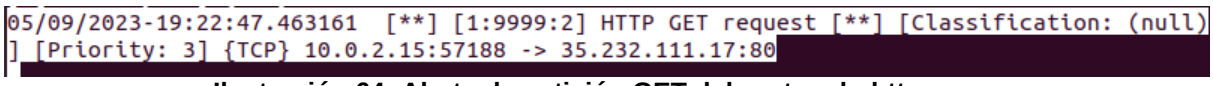

Ilustración 64: Alerta de petición GET del protocolo http.

3. Alerta de escaneo de puertos

alert tcp any any -> $HOME_NET any (msg:"Nmap port scan detected"; flow:to_server, established; dsize:0; flags:S; detection_filter:track by_src, count 1, seconds 10; sid:1000001;)

4. Alerta acceso protocolo ssh

alert ssh any any -> any any (msg:"SSH access detected"; sid:1000001;)

5. Alerta de phishing sobre PayPal

alert dns $HOME_NET any -> $EXTERNAL_NET 53 (msg:"Possible PayPal phishing detected"; dns_query; content:"paypal.com"; nocase; content:"com"; nocase; isdataat:1, relative; sid:20001; rev:1;)

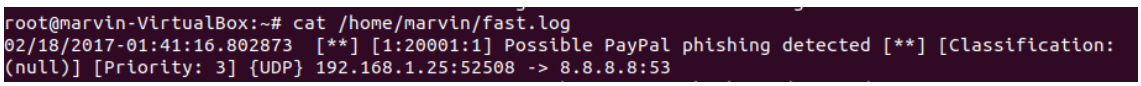

Ilustración 65: Alerta de phishing sobre PayPal.

Suricata, en todos los casos los registros de las alertas se almacenan en el archivo fast.log. En el caso de que se realice el análisis sobre la interfaz, se encontrara en /var/log/suricata/fast.log, mientras que, en el caso de realizarlo sobre un archivo, se encontrara con el mismo nombre en la dirección especificada con la sentencia -l.

Módulo de gestión: Este módulo está compuesto íntegramente por la herramienta TheHive.

TheHive realiza la gestión de los incidentes por medio de la gestión de casos. Los casos, se subdividen en tareas y observables. Para ejemplificar este proceso se efectuó la creación de una organización con varios usuarios gestionando un caso en particular.

Comenzamos creando una organización y agregar un usuario.

Esto se hace desde la sesión del administrador de usuarios que se genera por defecto (admin@thehive.local). En este proyecto la organización se nombró Domestic_SOC.

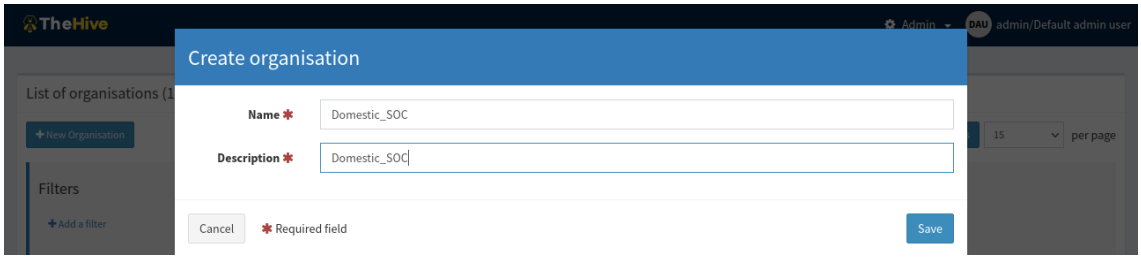

Ilustración 66: Creación de organización (TheHive).

Al crear un usuario de la organización hay que indicar los permisos que tendrá ligado, distinguiendo entre tres, administrador de la organización, analista y solo lectura.

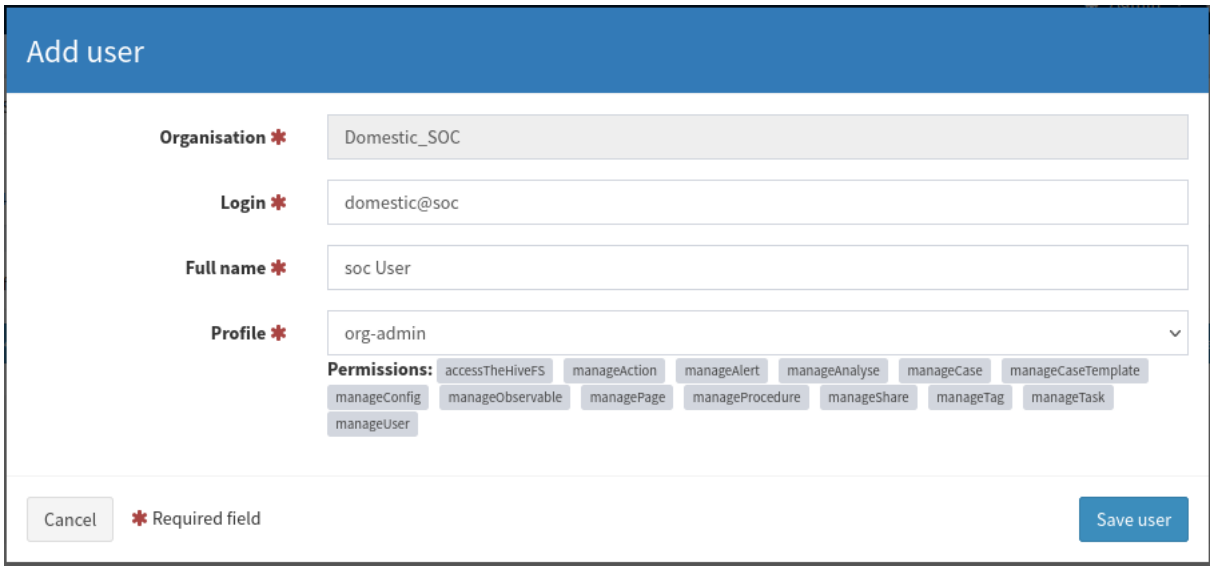

Ilustración 67: Creación de usuario (TheHive).

Posteriormente, se asigna la contraseña del usuario.

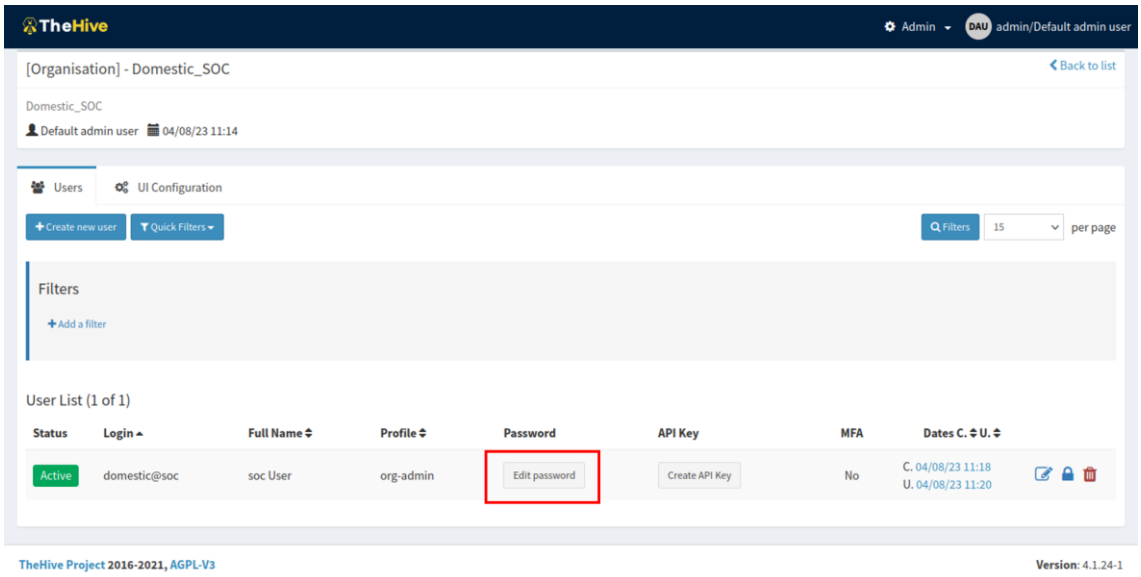

Ilustración 68: Configurar contraseña de usuario (TheHive).

Una vez creado las cuentas de los miembros de la organización se procede con la creación del caso nuevo. Este proceso se realiza desde la sesión del administrador de la organización, en este caso, domestic@soc.

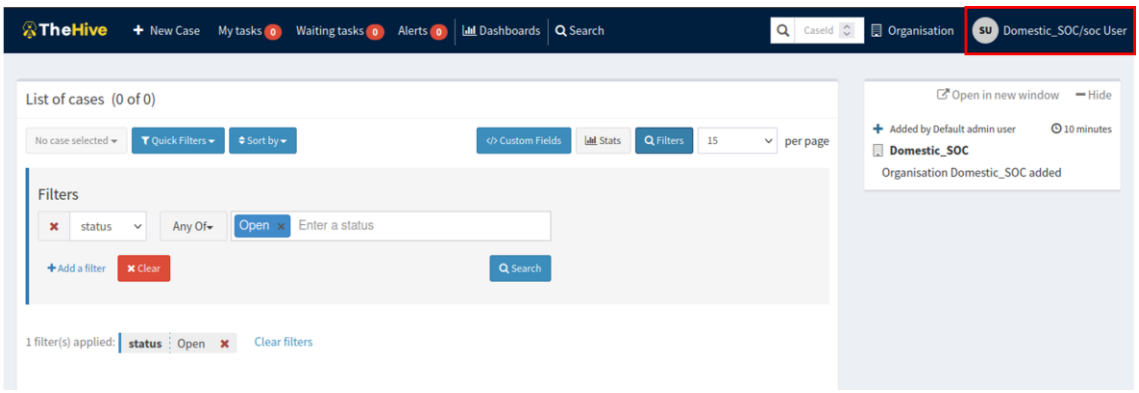

Ilustración 69: Sesión de usuario administrador de la organización.

Para la creación de un caso nuevo desde cero hay que indicar el nivel de gravedad, el TLP y el PAP, finalizando con descripción del caso a resolver. Del mismo modo, es posible asociarle una etiqueta que ayude a agrupar los casos con mismas características.

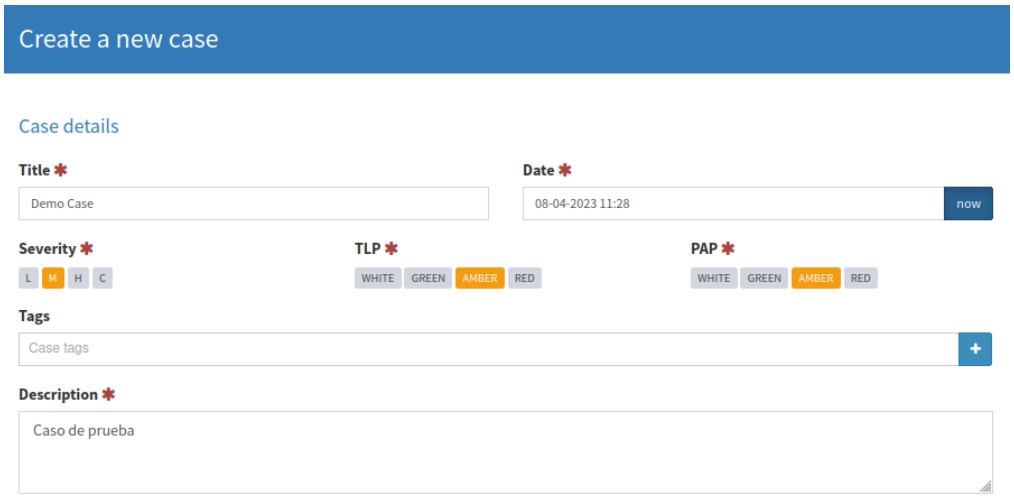

Ilustración 70: Creación de nuevo caso (TheHive).

Posteriormente, se deben añadir las tareas y los observables que el analista tiene que resolver para poder cerrar el caso.

Las tareas tienen un título y un grupo asociado. Al iniciar una tarea, se pueden agregar registros relacionados con la tarea en específico.

Por otro lado, el resto de miembros de la organización puede ver qué tareas y por quien está siendo gestionada.

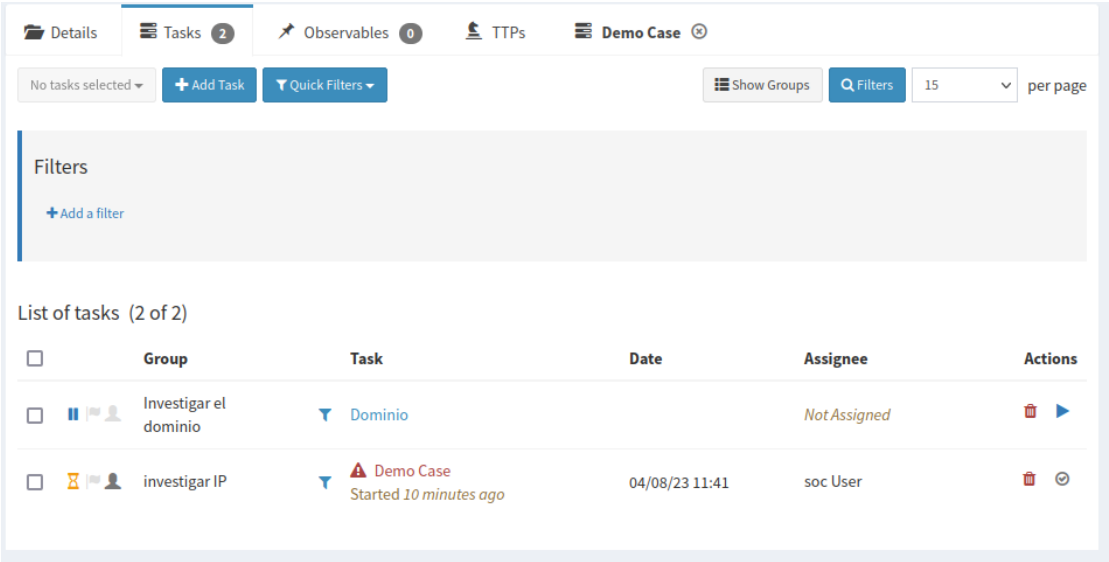

Ilustración 71: Lista de tares del caso.

Los observables cuenta con un campo tipo que permite discernir entre [ip, url, file, mail, …], valor, TLP, visto con anterioridad / ignorar observables similares, etiqueta asociada y descripción. Adicionalmente, los observables pueden marcarse como IoC para su posterior análisis.

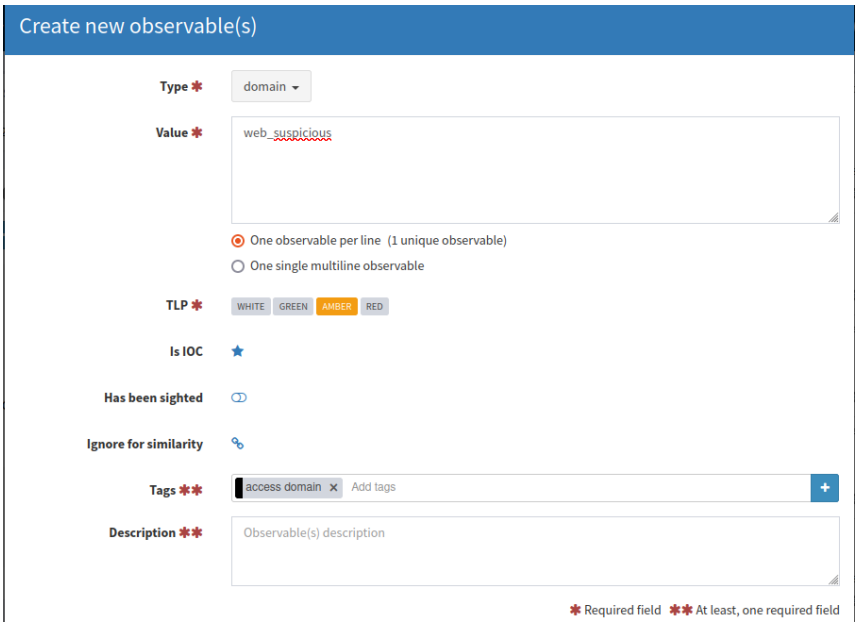

Ilustración 72: Creación de observable.

Al terminar las acciones que requiere cada tarea y el análisis de los observables se procede con el Cierre del caso, indicando la gravedad del caso y un resumen del resultado.

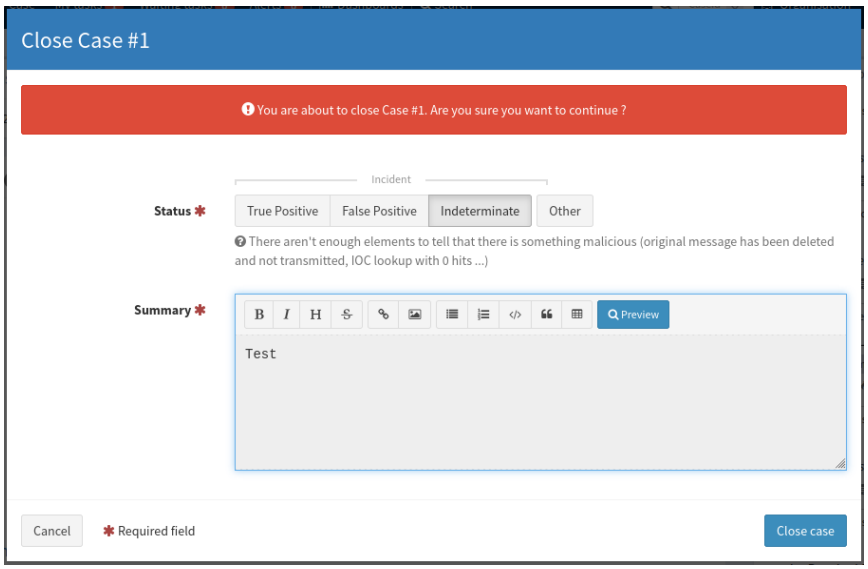

Ilustración 73; Cierre del caso.

Adicionalmente, TheHive cuenta en la banda lateral derecha el registro de las acciones ejecutadas por los diferentes miembros de la organización, actuando como historial de la organización.

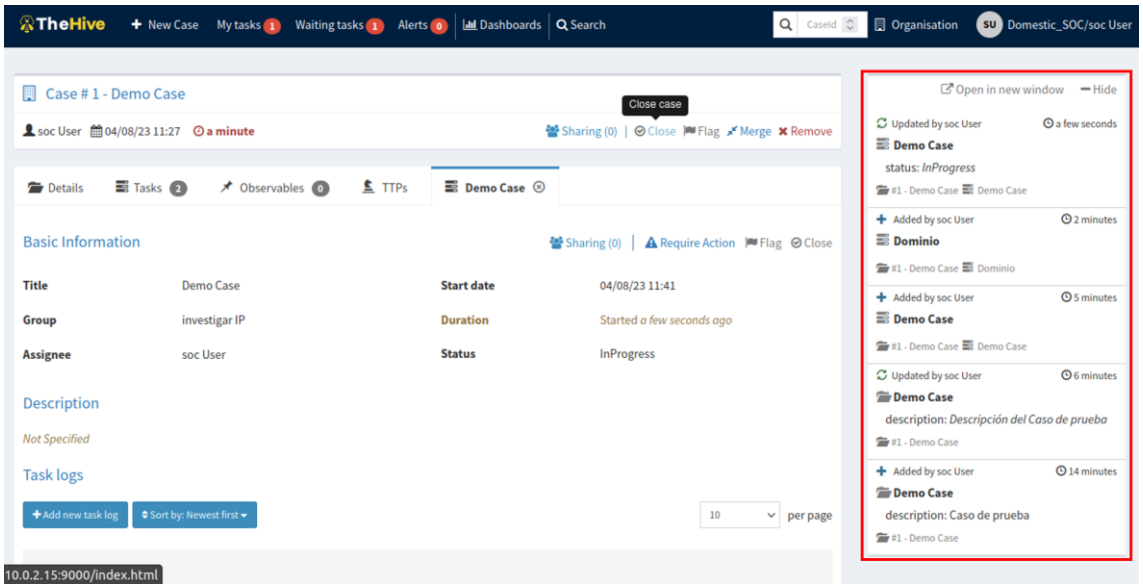

Ilustración 74: Historial de acciones.

Módulo de análisis: Este módulo está compuesto conjuntamente por la herramienta Cortex como concentrador de analizadores y los analizadores, destacando entre estos el caso especial de MISP.

Del mismo modo que TheHive, Cortex requiere la creación de una organización y usuarios. Con el fin de mantener la paridad mantendremos los mismos nombres que En el caso de TheHive.

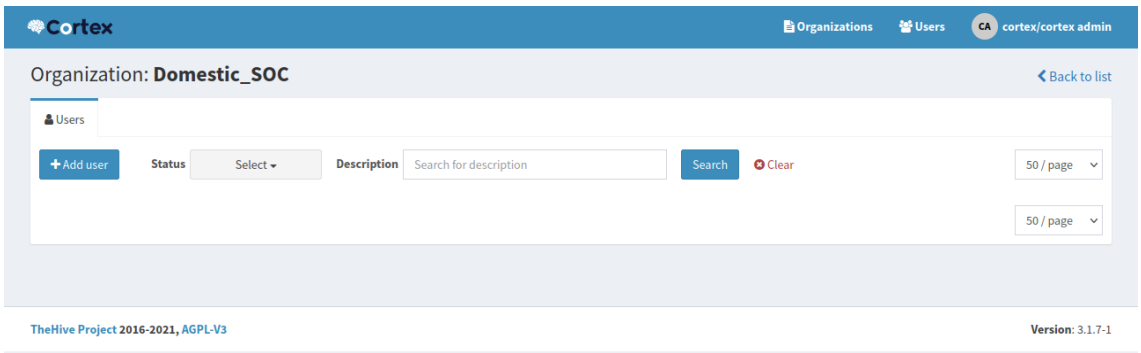

Ilustración 75: Creación de organización (Cortex).

Una vez creado la organización y el usuario se procede con la adición de los analizadores. Para ejemplificar este proceso se realizó la adición de VirusTotal, uno de los analizadores más populares, ya que concentra 61 motores de detección distintos.

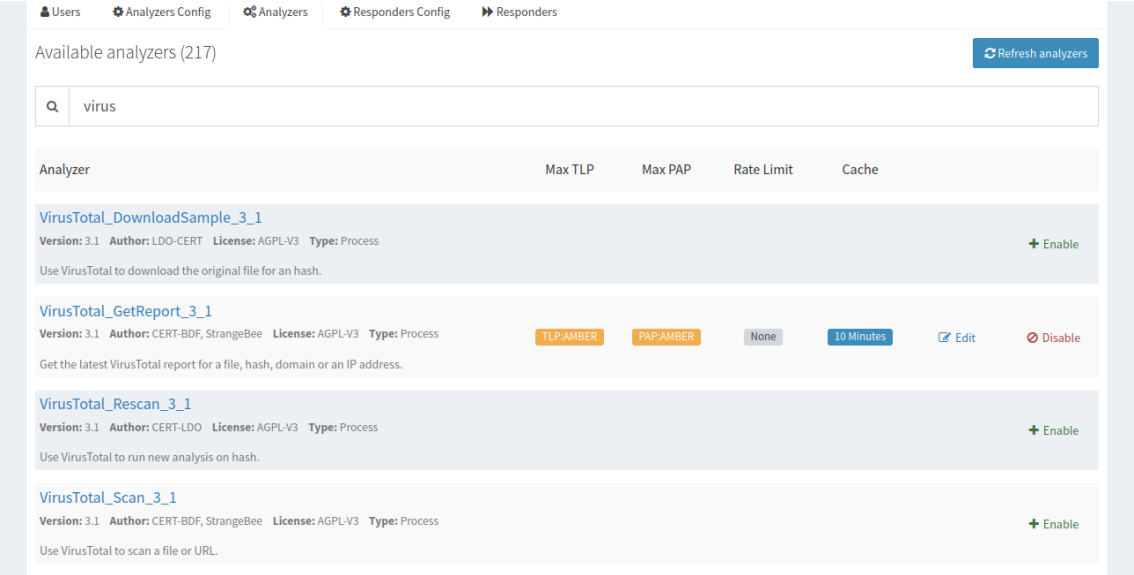

Ilustración 76: Lista de analizadores disponibles.

En este punto, una vez habilitados los analizadores deseados a la organización, podemos ejecutar un análisis de un IoC directamente desde la interfaz de Cortex, indicando los parámetros de TPL, PAP Data Type, Data y el analizador a ejecutar.

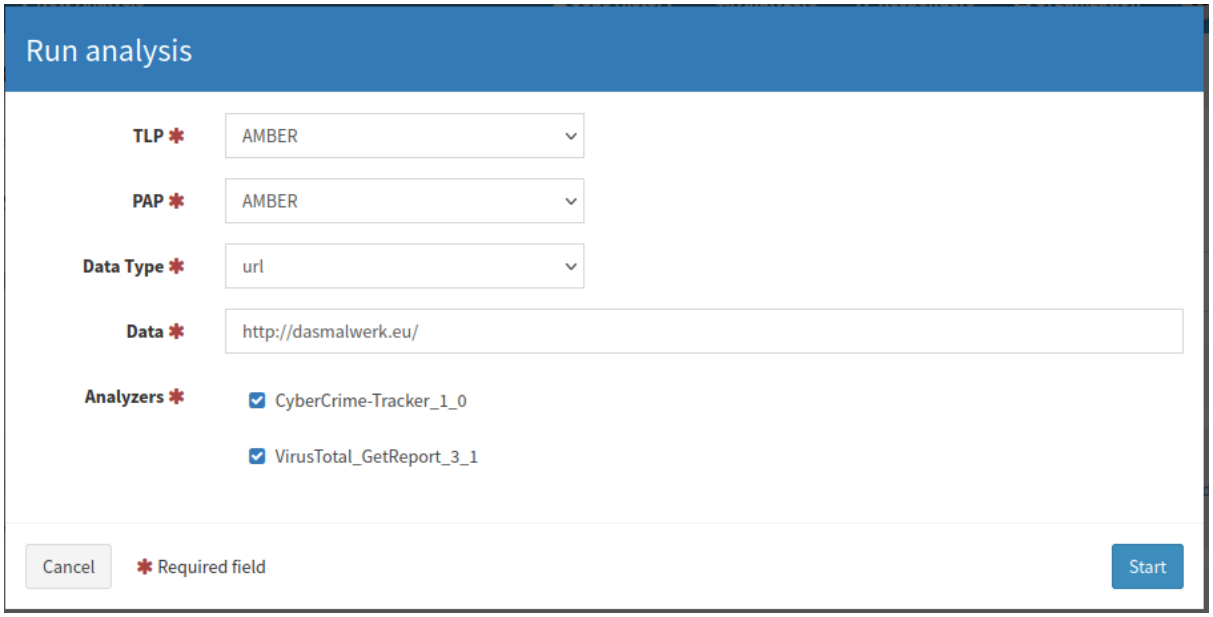

Ilustración 77: Ejecución del análisis en Cortex.

El resultado del análisis se resume en el número de comparaciones que han sido coincidentes con alguna amenaza y mostrando más en detalle un reporte con todo lo relacionado con el análisis.

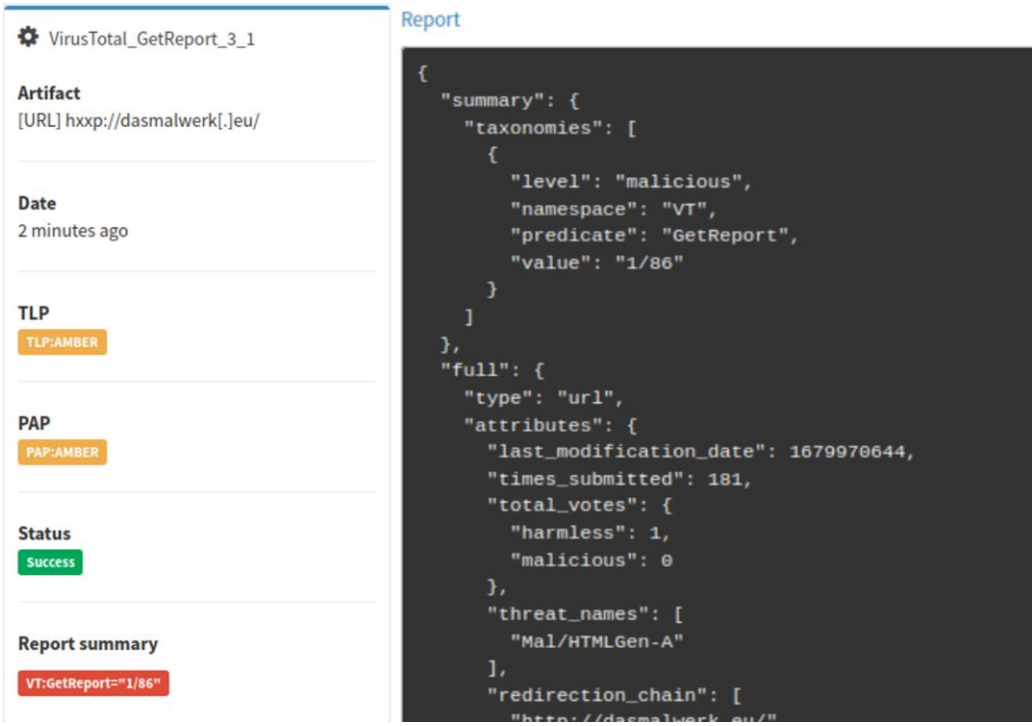

Ilustración 78: Resultados del análisis en Cortex.

Este reporte coincide con el que podemos encontrar realizando el análisis desde la plataforma web de VirusTotal.

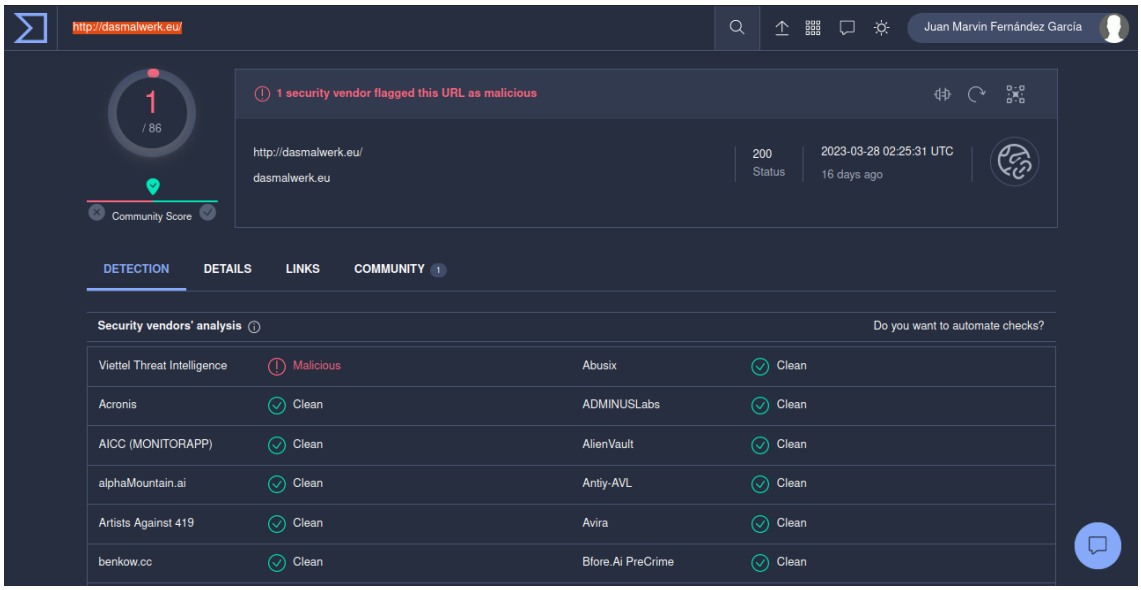

Ilustración 79: Resultados del análisis en la web VirusTotal.

Los resultados se pueden mejorar añadiendo la plantilla que ofrece TheHive (mirar 10.3 Plantilla de visualización de analizadores en TheHive), donde, en el caso de VirusTotal, ofrece el resultado de cada detalle de motores de detección, además de un enlace a la página de resultados en su web.

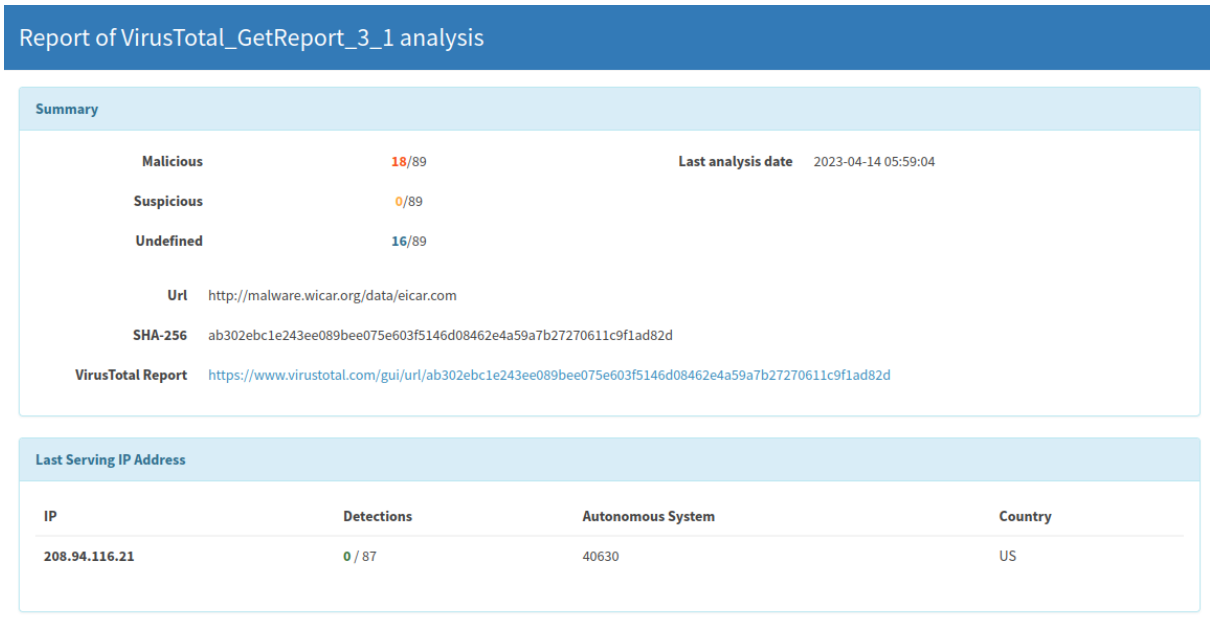

Ilustración 80: Resultados del análisis en Cortex con plantilla.

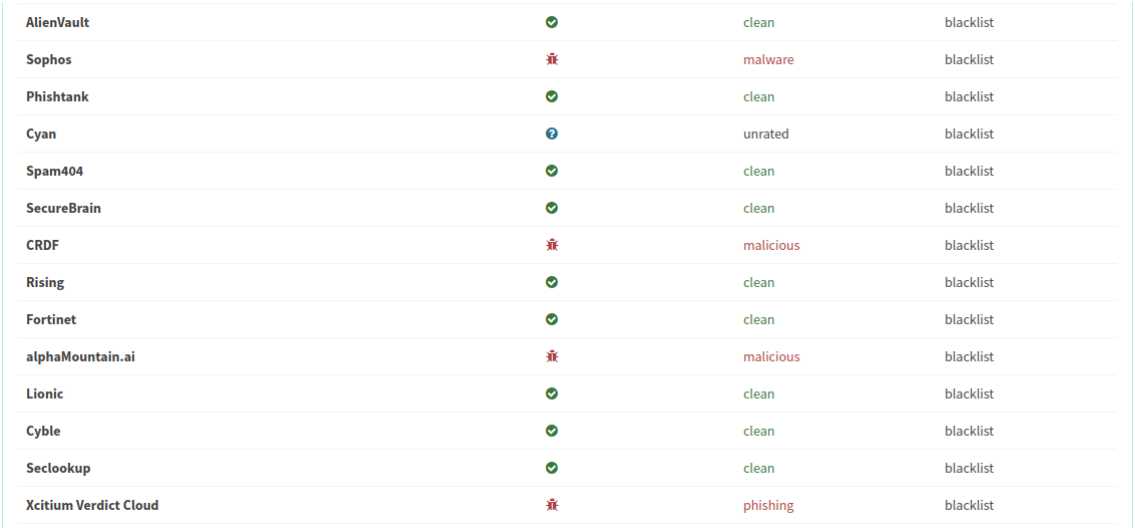

Ilustración 81: Resultados de los motores de detección de VirusTotal.

Módulo de almacenamiento:

Este módulo contará con la herramienta Elasticsearch y Casandra, las cuales almacenan los IoC generados en los diferentes modelos de gestión y análisis.

5.2 Modo de uso del SOC Doméstico

En esta sección se explica el flujo de trabajo entre las distintas herramientas dentro del SOC doméstico.

Las tareas esenciales que desempeña este SOC son: la detección y bloqueo de incidentes, el almacenamiento de los registros, la gestión de los IoC, y el análisis de estos mismos. Ejemplificaremos el flujo de trabajo frente a una amenaza hipotética.

El proceso comenzaría por el módulo de detección, donde la herramienta Suricata está realizando una escucha activa sobre todos los paquetes entrantes y salientes de la red.

En el momento en el que se detecta un evento coincidente con una de las reglas establecidas, Suricata lanzara una alerta del registro del evento.

Estas alertas son recogidas por el usuario, el cual pasará al módulo de gestión donde abrirá un caso en la herramienta TheHive. En este punto, el usuario introducirá los registros como observables e indicará las distintas tareas en específico que debe realizar para poder resolver el caso.

Será necesario indicar el nivel de gravedad, el TLP y el PAP del caso.

En el caso que TheHive encuentre algún caso previo donde coincida el valor de algún observable, lo resaltara. Esto permitirá al usuario revisar el caso anterior para poder tener en cuenta los resultados previos y de este modo agilizar la resolución del caso.

Ilustración 82: Coincidencia de observable previo.

En el caso en el cual TheHive no encuentre ninguna correlación previa, se ejecutará el módulo de análisis, en el cual Cortex como concentrador de analizadores ejecutar el análisis entre los analizadores seleccionados por el usuario. Los analizadores disponibles varían en función del tipo y el nivel de confidencialidad que tenga asignado el observable.

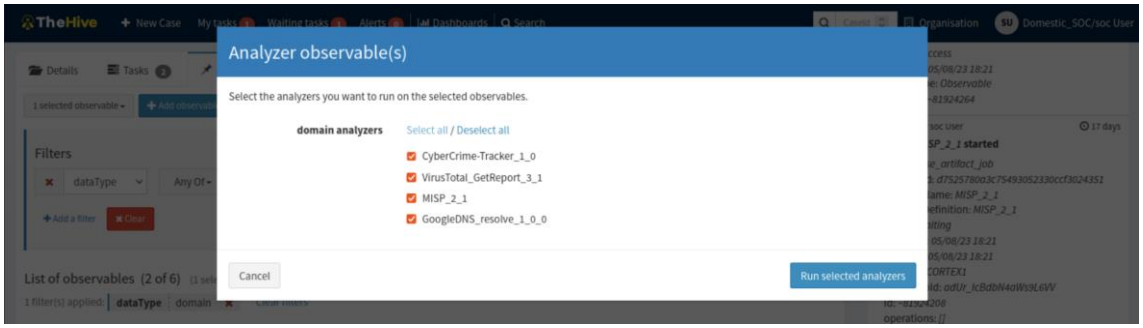

Ilustración 83: Ejecución del análisis en TheHive.

Los resultados arrojados por los analizadores como MISP indicaran el número de comparaciones que han sido coincidentes con alguna amenaza conocida. Este resultado, junto con el reporte detallado del análisis ayudará al usuario a evaluar si el incidente constituye una vulnerabilidad de seguridad real o no.

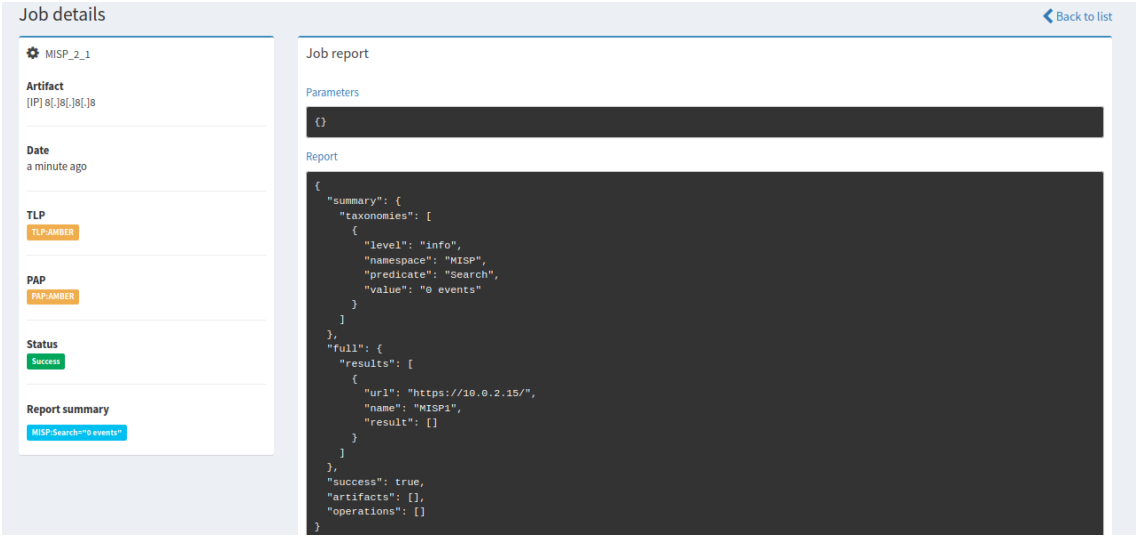

Ilustración 84 Resultados del análisis en TheHive.

Una vez comprendida la gravedad del incidente, se actuaría en consecuencia. Por ejemplo, en el caso en el que se halla establecido un escáner de puertos por parte de un agente externo o se envíen datos a una dirección desconocida, la solución directa sería establecer una regla en Suricata que bloque todas las conexiones provenientes de esa dirección en específico.

6 Discusión

En este trabajo se definieron los cuatro módulos que componen la arquitectura propuesta del SOC doméstico, junto con la interrelación entre ellos, para garantizar la correcta resolución de una amenaza hipotética en una red doméstica.

El flujo de trabajo del SOC doméstico se basa en un modelo semi-mecanizado, donde el monitoreo y análisis continuo de la red se realizan de manera automatizada. Solo se requiere la intervención del usuario cuando se registra un evento sospechoso que no ha sido previamente asignado a una acción específica. Las acciones posteriores de análisis de los IoC son ejecutadas por el usuario, quien toma decisiones en función de los resultados obtenidos de la correlación de incidentes previos, tanto internos como externos.

El SOC propuesto está diseñado para su uso en entornos domésticos, teniendo en cuenta las limitaciones de recursos en este tipo de entornos. Se ha enfocado en que la herramienta pueda ser manipulada por un único individuo. Además, dado el rápido avance de los mecanismos de intrusión por parte de agentes externos, la arquitectura se basa completamente en sistemas de código abierto gratuitos. La disponibilidad de un sistema en constante actualización proporciona una herramienta viable para cualquier usuario, sin la necesidad de contar con profundos conocimientos en ciberseguridad.

7 Conclusiones

En este proyecto, se ha buscado ofrecer una solución adaptada a los recursos y limitaciones de una red doméstica para proteger la integridad, confidencialidad y disponibilidad de los dispositivos y la información generada en los hogares.

Para lograr este propósito, se llevó a cabo un amplio estudio de la estructura y elementos que conforman un SOC, y se analizaron diversas herramientas *open-source* disponibles en el mercado para ensamblar los módulos fundamentales de la arquitectura SOC. Posteriormente, se configuraron las herramientas y se realizó un análisis individualizado de su funcionamiento. En último lugar, se ejemplificó el flujo de trabajo de la arquitectura SOC propuesta.

La arquitectura del SOC resultante está compuesta completamente por sistemas de código abierto gratuitos, lo que permite que el usuario pueda utilizar y controlar el sistema sin requerir profundos conocimientos en ciberseguridad. El proceso de monitoreo continuo de la red actúa de forma autónoma, requiriendo únicamente la atención del usuario para el análisis de eventos puntuales.

A lo largo del proyecto, se ha estudiado una amplia variedad de elementos involucrados en la ciberseguridad, lo que ha brindado una gran oportunidad para el desarrollo de conocimientos sobre el sector y los sistemas que lo componen.

El diseño de la arquitectura SOC propuesta ha demostrado tener un gran potencial para su implementación en entornos domésticos. Esto indica que la solución desarrollada puede ser altamente beneficiosa como medida de seguridad en el ámbito doméstico.

Se encontró que la curva de aprendizaje fue más lenta de lo esperado, lo que resultó en un ligero retraso en las tareas iniciales en comparación con la planificación inicial. Para garantizar el éxito del proyecto, se realizó una modificación en la metodología, simplificando el entorno y reduciendo el alcance de la solución.

Como posibles vías de mejora, se sugiere la implementación de la arquitectura en un entorno *Cloud* y la integración de los *feeds* de MISP para habilitar el *Threat Intelligence* en la arquitectura. Además, se propone ejecutar un estudio del comportamiento del SOC propuesto en un entorno doméstico simulado para evaluar su eficacia ante casos reales de amenazas. Estas mejoras pueden fortalecer y enriquecer aún más la solución propuesta.

8 Glosario

Acrónimo	Descripción
IDS	*Intrusion Detection System*
IoC	Indicadores de compromiso
IPS	*Intrusion Prevention System*
NGFW	*Next-Generation Firewalls*
PAP	*Permissible Actions Protocol*
SIEM	*Security Information and Event Management*
SOC	Centro de Operaciones de Seguridad
TLP	*Traffic Light Protocol*

Tabla 7: Acrónimos

9 Bibliografía

AWS. (2023). *¿Qué es Elasticsearch?* AWS. https://aws.amazon.com/es/what-is/elasticsearch/

blog.agood.cloud. (s. f.-a). *BUILDING CORTEX*. blog.agood.clou. Recuperado 15 de marzo de 2023, de https://blog.agood.cloud/posts/2019/09/22/building-cortex/

blog.agood.cloud. (s. f.-b). *BUILDING MISP*. https://blog.agood.cloud/posts/2019/04/29/building-misp/

Check Point. (2021a). *¿Qué es la detección y la respuesta de endpoint (EDR)?* Check Point. https://www.checkpoint.com/es/cyber-hub/what-is-endpoint-detection-and-response/#:~:text=Endpoint Detection and Response (EDR,respuesta automatizada basada en reglas.

Check Point. (2021b). *¿Qué es un sistema de detección de intrusos (IDS)?* Check Point. https://www.checkpoint.com/es/cyber-hub/what-is-an-intrusion-detection-system-ids/

Check Point. (2023). *Security Operations Center (SOC) Roles and Responsibilities*. Check Point. https://www.checkpoint.com/cyber-hub/threat-prevention/what-is-soc/security-operations-center-soc-roles-and-responsibilities/#:~:text=SOC teams are responsible for identifying%2C deploying%2C configuring%2C and,tickets from an organizations' employees.

COOPER, S. (2023). *The Best SOC Software Tools*. comparitech. https://www.comparitech.com/data-privacy-management/best-soc-software/

Crowdstrike. (2022). *WHAT IS A SECURITY OPERATIONS CENTER (SOC) ?* Crowdstrike. https://www.crowdstrike.com/cybersecurity-101/security-operations-center-soc/

elastic. (2023). *Enterprise Search*. elastic. https://www.elastic.co/enterprise-search

González, S. (2021). *Qué es Cyber Threat Intelligence*. Welivesecurity. https://www.welivesecurity.com/la-es/2021/11/08/que-es-cyber-threat-intelligence/

Hitesh, J. (2022). *Best SOC Software*. Network Admin Tools. https://www.netadmintools.com/best-soc-software/#wbounce-modal

IBM. (s. f.-a). *¿Qué es un Centro de Operaciones de Seguridad (SOC)?* IBM. Recuperado 23 de diciembre de 2022, de https://www.ibm.com/es-es/topics/security-operations-center

IBM. (s. f.-b). *What is SIEM?* IBM. Recuperado 10 de febrero de 2023, de https://www.ibm.com/topics/siem

IKUSI. (2023). *SOC: crear o contratar, ¿qué es mejor para mi empresa?* IKUSI. https://www.ikusi.com/es/blog/soc-crear-o-contratar-que-es-mejor-para-mi-empresa/

IMF. (2020). *¿Qué es un SOC y qué actividades realiza?* IMF. https://blogs.imf-formacion.com/blog/tecnologia/que-es-soc-actividades-realiza-201903/

INCIBE. (2020). *Firewall tradicional, UTM o NGFW. Diferencias, similitudes y cuál elegir según tus necesidades*. (INCIBE). https://www.incibe.es/empresas/blog/firewall-tradicional-utm-o-ngfw-diferencias-similitudes-y-cual-elegir-segun#:~:text=El firewall tradicional,ofrece unas garantías de seguridad.

INCIBE. (2021). *Sistemas EDR: qué son y cómo ayudan a proteger la seguridad de tu empresa*. INCIBE. https://www.incibe.es/empresas/blog/sistemas-edr-son-y-ayudan-proteger-seguridad-tu-empresa

Jiménez, J. (2022). *SOC as a Service: qué es y por qué mejora la seguridad*. redeszone. https://www.redeszone.net/tutoriales/seguridad/que-es-soc-as-a-service/

kaspersky. (2023). *¿Qué es la inteligencia de amenazas? Definición y explicación*. kaspersky. https://latam.kaspersky.com/resource-center/definitions/threat-intelligence

KEEPCODING. (2023a). *¿Qué es MISP en ciberseguridad?* KEEPCODING. https://keepcoding.io/blog/que-es-misp-en-ciberseguridad/

KEEPCODING. (2023b). *¿Qué es Suricata en ciberseguridad?* KEEPCODING. https://keepcoding.io/blog/que-es-suricata-en-ciberseguridad/

Learning, G. (2022). *What is Firewall? – Types, How Does it Work, Advantages*. Great Learning. https://www.mygreatlearning.com/blog/what-is-network-security-firewall/#disadvantages-of-using-firewalls

MISP Threat Sharing. (s. f.). *MISP*. MISP Threat Sharing. Recuperado 4 de marzo de 2023, de https://www.misp-project.org/

ORACLE. (2023). *¿Qué es un SOC?* ORACLE. https://www.oracle.com/es/database/security/que-es-un-soc.html

Park, A. (2022). *What to Know About Open-Source Software: Benefits and Advantages*. Heavybit. https://www.heavybit.com/library/article/open-source-software-benefits-advantages

Ramiro, R. (2018). *Un trio perfecto con TheHive, Cortex y MISP*. ciberseguridad.blog. https://ciberseguridad.blog/un-trio-perfecto-con-thehive-cortex-y-misp/#:~:text=Casos en TheHive&text=Cuando los analistas están trabajando,lo que han estado haciendo.

Red Hat. (2022a). *¿Qué es SOAR?* Red Hat. https://www.redhat.com/es/topics/security/what-is-soar

Red Hat. (2022b). *Diferencias entre IaaS, PaaS y SaaS*. Red Hat. https://www.redhat.com/es/topics/cloud-computing/iaas-vs-paas-vs-saas

Red Hat. (2023). *¿Qué es el open source?* Red Hat Summit. https://www.redhat.com/es/topics/open-source/what-is-open-source

Suricata. (s. f.-a). *Quickstart guide Suricata*. https://docs.suricata.io/en/suricata-6.0.11/quickstart.html

Suricata. (s. f.-b). *Rules Format*. Recuperado 4 de marzo de 2023, de https://docs.suricata.io/en/suricata-6.0.11/rules/intro.html

Tecnógrafos. (2022). *Roles, procedimientos y dinámica de trabajo que todo equipo SOC debería tener*. Tecnógrafos. https://tecnografos.es/2022/03/08/roles-y-procedimientos-que-todo-equipo-soc-deberia-tener/

TheHive Project. (2021). *Step-by-Step Guide TheHive*. https://docs.thehive-project.org/thehive/installation-and-configuration/installation/step-by-step-guide/

TheHive Project. (2022a). *Cortex*. TheHive Project. https://docs.thehive-project.org/cortex/

TheHive Project. (2022b). *TheHive*. TheHive Project. https://thehive-project.org/

vmware. (2023). *¿Qué es la detección y respuesta en los terminales (EDR)?* vmware. https://www.vmware.com/es/topics/glossary/content/endpoint-

detection-and-response-edr.html#:~:text=La detección y respuesta en los terminales (EDR) es una,de respuesta automatizada a amenazas.

Wazuh. (2023). *Elasticsearch single-node cluster*. https://documentation.wazuh.com/current/deployment-options/elastic-stack/distributed-deployment/elasticsearch-cluster/elasticsearch-single-node-cluster.html

Xataka. (2022). *El ecosistema IoT del hogar conectado en riesgo: millones de dispositivos bajo una amenaza que aún no ha sido corregida*. Xataka. https://www.xatakahome.com/seguridad-en-el-hogar/ecosistema-iot-hogar-conectado-riesgo-millones-dispositivos-amenaza-que-no-ha-sido-corregida

Zoho Corporation. (2023). *Herramientas y tecnologías utilizadas en los SOC*. Zoho Corporation.

10 Anexos

10.1 Instalación de python3 y python2

Instalación de python 3

```
sudo apt install python3
sudo apt install python3-pip
pip3 –versión
sudo apt install python3-testresources
```

Instalación de python2

```
curl https://bootstrap.pypa.io/pip/2.7/get-pip.py --output get-pip.py
sudo python2 get-pip.py
pip2 --version
```

Instalación de pip2 y pip3

```
sudo pip2 install -U pip setuptools && sudo pip3 install -U pip setuptools
```

10.2 Interfaz de trabajo

Para indicar la interfaz sobre la que trabajara suricata aplicamos el comando ip ad, que mostrara todas las direcciones IP de la red.

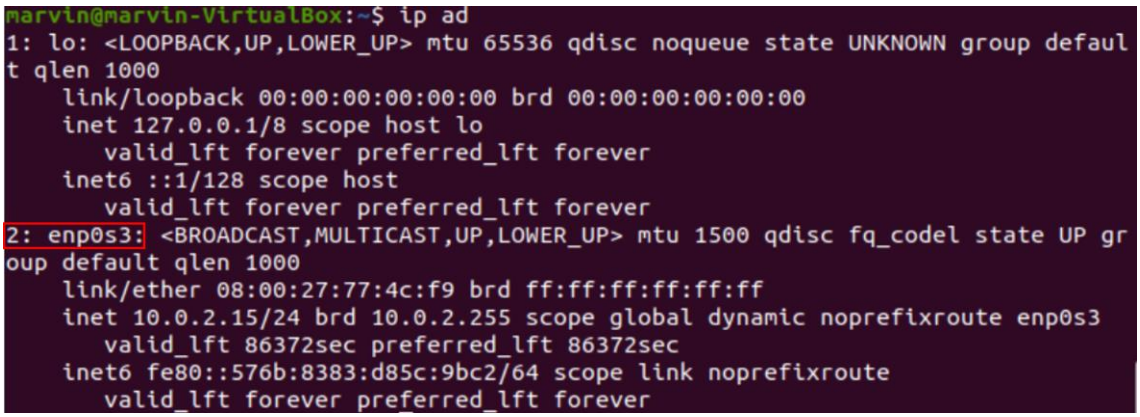

Ilustración 85: Interfaz de trabajo.

10.3 Plantilla de visualización de analizadores en TheHive

En la sesión administrador de usuario. Entramos en admin > Analyzer templates, donde, hay que descargar el archivo oficial de plantillas desde el enlace que proporciona “from here”.

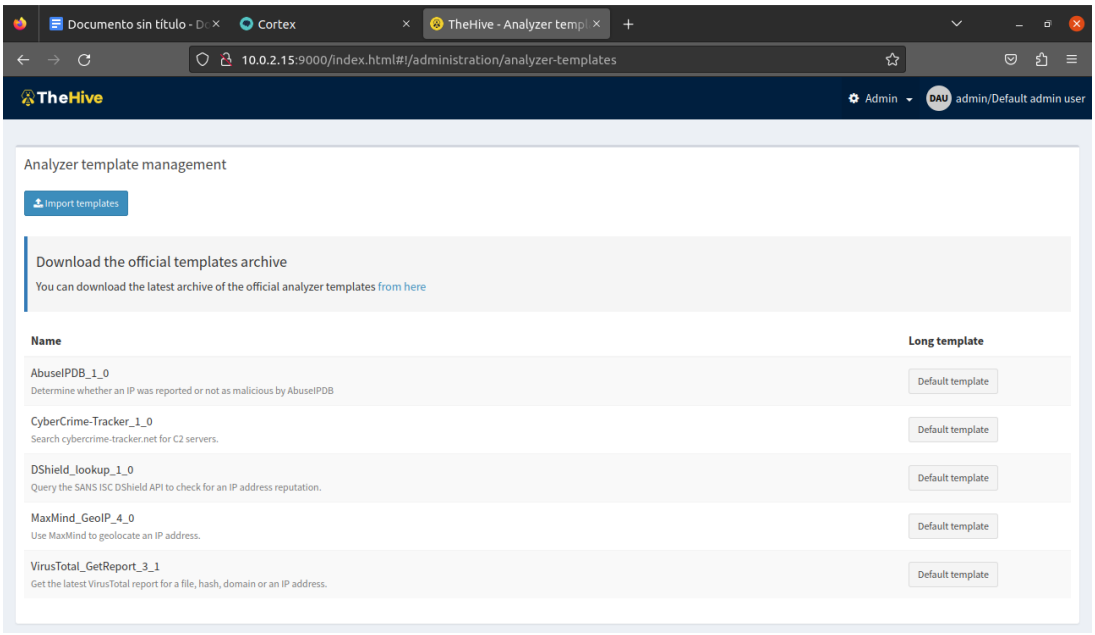

Ilustración 86: Analyzer template management.

Una vez se tenga el archivo report-templates.zip, hay que importarlo en formato zip.

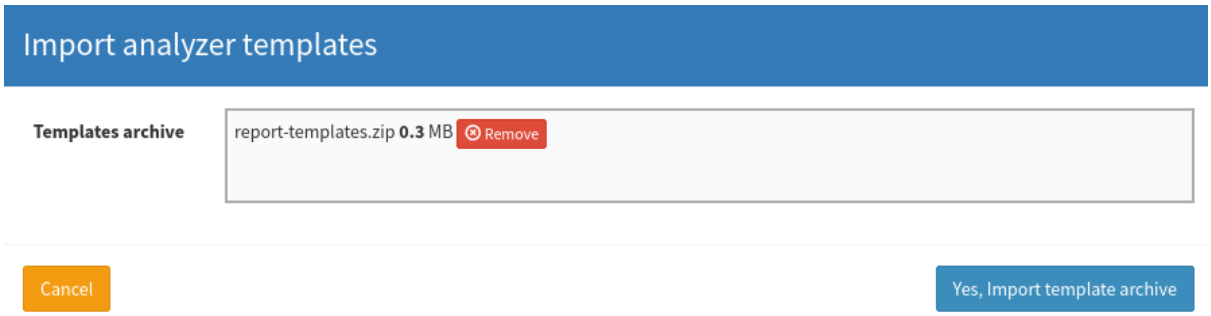

Ilustración 87: Importe de plantilla de analizadores.

Una vez importadas las plantilles se observarán las plantilles asociadas a los analizadores que se tengan habilitados.

Ilustración 88: Plantillas de los analizadores.