Universidad Autónoma del Estado de México

Centro Universitario UAEM Valle de Chalco

Inteligencia de amenazas basada en honeypots para una ciberseguridad oportuna en la nube

T E S I S

QUE PARA OBTENER EL GRADO DE

MAESTRO EN CIENCIAS DE LA COMPUTACIÓN

P R E S E N T A

Lic. Luis Ángel Rivera Nuñez

DIRECTORA

Dra. María de Lourdes López García

CO-DIRECTOR

Dr. Manuel Ávila Aoki

TUTOR

Dr. Mario Farias Elinos

VALLE DE CHALCO SOLIDARIDAD, MÉXICO FEBRERO 2023

CUVCH

Índice

Índice de Figuras

Índice de tablas

RESUMEN

En nuestra actualidad la Internet cuenta con recursos y servicios que brindan la posibilidad de administrar grandes cantidades de información la nube, por ejemplo, facilita la actividad de resguardar y consultar información en una amplia variedad de dispositivos electrónicos como smartphones, computadoras, tablets, televisores, entre otros. Esta práctica permitirá que el usuario se desarrolle de una manera exponencial en el tratamiento y distribución de los datos como el envío, almacenamiento y manipulación.

Los recursos informáticos en la nube se han ido perfeccionando con el paso del tiempo, hoy día, la industria del entretenimiento implementó estos elementos a su favor, al almacenar y distribuir películas, canciones y vídeojuegos en casi cualquier dispositivo, lo que facilita el poder entretenerse en cualquier parte del mundo. A su vez, la manera en cómo se trabaja y estudia evolucionó drásticamente debido a la pandemia del covid-19, al obligar a la conectividad e incorporar nuevas tecnologías en cuanto hardware y software respecta, almacenando y compartiendo grandes cantidades de información por medio de la nube.

El desarrollo de este trabajo de investigación tiene el propósito de resaltar la importancia que tiene la seguridad en informática, dicho lo anterior se busca implementar un sistema trampa llamado Honeypot en lugares estratégicos para recopilar información de posibles amenazas que puedan comprometer la integridad de la red e incrementar la seguridad, para prevenir de una potencial amenaza que pueda vulnerar información sensible de los usuarios, el estudio de los casos que se obtengan nos permitirán crear un plan de acción para contrarrestar, ganar tiempo y minimizar eventos que puedan poner en riesgo un sistema informático.

Capítulo 1.
INTRODUCCIÓN

En la era digital, la conexión a Internet cobra una gran importancia por el uso ilimitado de las aplicaciones como las redes sociales y el entretenimiento. Los usuarios tienen una gran variedad de temas que pueden consultar en línea o descargando las aplicaciones para diferentes actividades (laborales, familiares, personales, etc.) a través de las tiendas de aplicaciones, por lo que la vida digital pone al alcance recursos ilimitados.

1.1 Planteamiento del problema

Debido al desarrollo exponencial en el uso de aplicaciones y servicios, en la Internet ha surgido la creación de ecosistemas digitales donde participan variedad de dispositivos que incluyen software y hardware. En consecuencia, se generan grandes cantidades de información que se transmiten y se almacenan de forma remota.

En muchos de los casos, los usuarios manejan su información de manera que no se protegen y con base en la confianza de las empresas que ofertan sus servicios, proporcionan datos privados o permiten el acceso a sus dispositivos.

La gran amenaza que los usuarios enfrentan es la manipulación de su información que, en muchas ocasiones, es autorizada usando técnicas como no continuar el proceso de instalación hasta que el usuario acepte acceder a sus contactos. Por otro lado, las amenazas a las que se enfrentan las organizaciones o empresas son a los ataques a sus servidores, bases de datos, puntos de acceso, por mencionar algunos.

Por lo cual, surgen las siguientes preguntas de investigación. ¿Cuáles son los mecanismos de seguridad que permiten proteger la información de los usuarios en la nube?, ¿cómo configurar y utilizar la información de los mecanismos de seguridad ante la inteligencia de amenazas?, ¿de qué manera se puede establecer un mecanismo de seguridad que identifique la forma de operar del ataque, se analice la vulnerabilidad, se evalúe el riesgo y se proteja la organización?

Estas preguntas son claramente un tema interesante de investigación que se centra en al área de la seguridad informática.

1.2 Hipótesis

La aplicación de un mecanismo de seguridad que identifique cómo opera un ataque permite conocer la amenaza, analizar la vulnerabilidad, evaluar el riesgo y, por tanto, mitigar un ataque cibernético en la red informática de una organización.

1.3 Justificación

La computación en la nube se define como la utilización de la Internet para acceso a servicios y el almacenamiento de la información en una locación remota. En este sentido, se tienen tres modos de implementarla (Emmett Dulaney, 2011):

- Plataforma como servicio (PAAS, Platform as a Service). En este modelo, los vendedores permiten que las aplicaciones se ejecuten en su infraestructura.
- Software como servicio (SaaS, Software as a Service). En este modelo, se ejecutan las aplicaciones de forma remota en la Web y no se requiere la instalación de software local.

- Infraestructura como servicio (IaaS, Infrastructure as a Service). En este modelo se utiliza la virtualización y los clientes pagan a un tercero por los recursos empleados.

Con estas opciones, el uso de los servicios en Internet se incrementa cada vez más, por lo que es de prioridad alta que la protección de la información se realice y con ello ofrecer mayor seguridad a los millones de usuarios de la Internet.

1.4 Objetivos

Los objetivos, general y específicos de este trabajo son los siguientes:

General

Proponer un mecanismo de seguridad basado en Honeypots para identificar y disminuir los ataques cibernéticos en una organización.

Específicos

1. Investigar las técnicas de prevención de ataques en una organización.
2. Establecer un escenario inseguro utilizando una red local.
3. Realizar hacking ético como instrumento de investigación y demostración al escenario definido.
4. Analizar el uso no apropiado de la información en una organización.
5. Proponer estrategias y cuidados del tratamiento de la información en la organización.

1.5 Metodología de la investigación

En este proyecto se utiliza el método de investigación tecnológica que tiene las siguientes características (Carlos Muñoz Razo, 2015):

- Útil para obtener conocimiento y resolver un problema
- Factible para soluciones en casos particulares

- Las herramientas utilizadas están en constante evolución

Al alcanzarse una mejora, avance técnico o diseño de una innovación, de inmediato surge un nuevo invento o una nueva necesidad.

1.6 Organización de la tesis

El resto de este documento se organiza como sigue. En el capítulo 2 se presentan los conceptos básicos, mientras que en el capítulo 3 se particulariza a los conceptos, definición y clasificación de los honeypots. En el capítulo 4 se muestran los escenarios de ataque a los que se enfrenta el honeypot y por último en el capítulo 5 se muestran las conclusiones.

Capítulo 2.
CONCEPTOS BÁSICOS

En este capítulo se presentan los conceptos básicos de la investigación que contemplan la historia de la comunicación a través de la Internet, los riesgos en la seguridad, la seguridad y por último, las herramientas de seguridad a nivel de red.

2.1 La Información en la nube

El ser humano desde tiempos memorables se ha visto en la necesidad de intercambiar información, es decir, tener un canal de comunicación en el cual le permita sostener una conversación con otra persona, para ello tiene que existir un emisor, un canal y un receptor.

Partiendo de esta premisa, se han buscado distintas formas y herramientas para poder realizar esta tarea, por ejemplo, Thomas Alva Edison, un científico estadounidense, quien con su revolucionario invento de fonógrafo, que más adelante evolucionaría en lo que hoy día conocemos como teléfono, sería el principio de la revolución de las tecnologías de información, el medio por el cual comenzaríamos a interactuar de manera remota de un punto a otro y es preciso mencionar que como primera instancia las señales análogas serían las primeras en ser enviadas por este dispositivo.

Con el paso de los años, la telefonía obtuvo un gran potencial ya que su principal virtud era la inmediatez en la que una persona podía entrar en contacto

y la fluidez en la que una persona en un lugar remoto podía sostener una conversación.

Al expandir este canal de comunicación, se inició la incorporación de conmutadores que eran operados por personas que tenían el propósito de intercambiar conexiones a fin de enlazar la comunicación origen-destino, un trabajo que en nuestros días resulta obsoleto y poco eficiente.

Para fines de este trabajo no se puede dejar de lado los orígenes de la Internet. Es de suma importancia conocer cuáles fueron los cimientos que dieron forma a esta tecnología que resulta ser ya una herramienta de comunicación indispensable para nuestras vidas como el agua, la electricidad y el gas, entre otras.

De primera mano sus orígenes tienen que ver con el Telégrafo, como se observa en la Figura 2.1, un dispositivo pequeño el cual tenía una serie de piezas mecánicas que mediante señales eléctricas enviaba un mensaje a kilómetros, para ello empleaban un protocolo el cual tenía por nombre código morse.

Figura 2.1 Inicios de la comunicación a través del Código Morse (Editorial Etecé, 2021)

La Internet es una herramienta indispensable, permite que la comunicación sea rápida e instantánea, grandes cantidades de información se encuentran a nuestra

disposición un recurso que ha ido creciendo con el paso de los años debido a que las tecnologías de la información están en una constante evolución. El concepto *Internet* hace referencia a una gran red mundial de computadoras conectadas mediante diferentes tipos de enlaces (satélites, por radio o incluso, submarinos). Esta gran red permite compartir información y tiene varias peculiaridades: es barata, pública, fácil de usar, está de moda y da de comer a mucha gente.

Para poder llegar a ser el sistema de comunicación actual, experimentó una serie de cambios en la forma en cómo se armaba una red y las piezas que la integran, pero no siempre fue así, de hecho, desde su origen no se le conocía como la red de todas las redes, se nombraba ARPANET.

Para encontrar los orígenes de internet nos tenemos que remontar hasta 1969 cuando el departamento de la defensa estadounidense decide crear la Arpanet, una red basada en paquetes que interconectaban múltiples centros. Esta red, inicialmente pequeña y utilizada casi en exclusiva por universidades, fue creciendo con la conexión de un número mayor de equipos, especialmente tras la aparición del correo electrónico en 1982 (Jorge Pérez y Carlos González Valderrama, 2015).

En términos prácticos la Internet en los últimos años ha evolucionado de pasar a un entorno de comunicación militar a ser la red de todas redes, un recurso informático y de comunicación que nos ha brindado a incorporarnos a una nueva etapa en nuestras vidas: la era de la información. Su crecimiento se ha incrementado debido a que se han integrado nuevas plataformas de trabajo es decir servicios tales como Correos electrónicos, Sitos Web, plataformas de transacciones bancarias, por mencionar algunas, nos han invitado a depositar grandes cantidades de datos que resultan ser confidenciales y sensibles para la mayoría de las empresas y usuarios finales.

Estas prácticas, para nuestros días en términos de la seguridad en informática, no se puedan dejar en un segundo plano, es evidente que para a aquellas empresas que dependen mucho del capital humano para que puedan sobrevivir son totalmente responsables del manejo de los datos de sus usuarios. La nube nos ha permitido llevar nuestra forma de trabajar y de comunicar a un escalón más alto, es decir, servicios de almacenamiento, juegos, aplicaciones, de entretenimiento, como ejemplo de ello. La forma en cómo tratamos la información y su protección cambió en los últimos tiempos gracias a la incorporación de nuevas tecnologías y su abaratamiento a nivel de producción. Los grandes fabricantes de la industria de la computación, comunicación y entretenimiento han encontrado una forma óptima para integrar, aprovechar y explotar la nube, un ejemplo de ello es cuando consumimos series y películas a través de un servicio de *streaming* y contenido multimedia.

2.2 Modelos de servicios en la nube

Tres modelos arquetípicos y sus combinaciones derivadas describen la prestación de los servicios en la nube. A menudo se hace referencia a los tres modelos individuales como el “Modelo SPI,” donde “SPI” hace referencia a Software, Plataforma e Infraestructura, respectivamente y se definen del siguiente modo (Nelly L. Hernández y Anderson S. Florez Fuente, 2014):

- Cloud Software as a Service (SaaS). En el Software de nube como servicio, la capacidad proporcionada al consumidor consiste en utilizar las aplicaciones del proveedor que se ejecutan en una infraestructura de nube. Puede accederse a las aplicaciones desde varios dispositivos del cliente a través de una interfaz de cliente ligero como un navegador de Internet (p.ej., correo web). El consumidor no gestiona, ni controla la infraestructura de nube subyacente que incluye la red, servidores, sistemas operativos, almacenamiento o incluso capacidades de

aplicaciones individuales, con la posible excepción de unos parámetros de configuración de la aplicación específicos del usuario limitados.

- Cloud Infrastructure as a Service (IaaS). En la infraestructura de nube como servicio, la capacidad suministrada al consumidor es abastecerse de procesamiento, almacenamiento, redes y otros recursos computacionales fundamentales de forma que el consumidor pueda desplegar y ejecutar software arbitrario, que puede incluir sistemas operativos y aplicaciones. El consumidor no gestiona, ni controla la infraestructura de nube subyacente, pero tiene control sobre los sistemas operativos, almacenamiento, aplicaciones desplegadas y la posibilidad de tener un control limitado de componentes de red seleccionados por ejemplo hospedar firewalls.

- Modelos de despliegue en la nube. Con independencia del modelo de servicio utilizado (SaaS, PaaS, IaaS,) hay cuatro formas principales en los que se despliegan los servicios en la nube y se caracterizan con modelos de despliegue adicionales que afrontan requisitos específicos:

    - Nube pública: La infraestructura de nube se pone a disposición del público en general o de un gran grupo industrial y es propiedad de una organización que vende los servicios en la nube.

    - Nube privada: La infraestructura de nube se gestiona únicamente para una organización. Puede gestionar la organización o un tercero y puede existir tanto en las instalaciones como fuera de ellas.

    - Nube comunitaria: La infraestructura de nube la comparten diversas organizaciones y soporta una comunidad específica que tiene preocupaciones similares (por ejemplo, misión, requisitos de seguridad, políticas y consideraciones sobre cumplimiento normativo). Puede ser gestionada por las organizaciones o un tercero y puede existir en las instalaciones y fuera de ellas.

- Nube híbrida: La infraestructura de nube es una composición de dos o más nubes (privada, comunitaria o pública) que se mantienen como entidades separadas pero que están unidas por tecnología estandarizada o propietaria que permite la portabilidad de datos y aplicaciones (por ejemplo, procedimientos de escalado para el equilibrio de cargas entre nubes en el caso de picos puntuales).

2.3 Riesgos y Vulnerabilidades

Debido al crecimiento exponencial de las tecnologías de la información y la integración de nuevas plataformas de trabajo, es decir, sistemas operativos que año con año, sea el caso de las aplicaciones para Android y IOS de Apple los sistemas que la integran se ven en la necesidad de ir mejorando día a día. En consecuencia, existe una curva de progreso en donde las aplicaciones sufren cambios en su estructura funcional como es el de la seguridad.

Todos los dueños de las grandes compañías como por ejemplo Microsoft, Mac OS , Linux, Android, IOS, por mencionar algunos, se ven en la necesidad de proveer en forma de paquetes de actualización, recursos donde el usuario tiene que validar para aumentar o mejorar la seguridad de sus usuarios y la de los equipos que hacen uso de sistemas operativos de estas compañías que a su vez se crean células de comunidades, algunas experimentadas, otras con conocimientos básicos donde se comparten información para alertar que existen ciertas vulnerabilidades que pueden vulnerar la información sensible que depositan en ellas.

Es un error que el usuario final deposite toda confianza en sus dispositivos móviles, computadoras, tabletas, pantallas inteligentes por una sencilla razón son muy pocos los dispositivos en el mundo los cuales puedan soportar cierta fluidez en cuanto a su manejo, aquí entra el criterio y las buenas prácticas que pueda tener el usuario, es decir, que tan educado se encuentra para poner en la balanza

el alcance que puede tener manejar servicios como Facebook, YouTube, Tiktok, Telegram, entre otros recursos que por su naturaleza son bancos de información que a diario experimentan un gran tráfico de datos que va desde imágenes, vídeos, caracteres binarios, etcétera.

Resulta ser que los servicios son parte de la nube, su principal característica es brindar un servicio a través de la Internet en un algún lugar remoto en un sitio o dispositivo que no se conoce su ubicación con exactitud y de alguna manera crea cierta incertidumbre y desconfianza al manejar estos servicios. Si bien una de sus principales virtudes es que los usuarios tienen acceso casi en cualquier punto del planeta, esa característica la convierte muy atractiva y eficiente pero también la posiciona como blanco de ataques para obtener información confidencial o privada.

En los últimos tiempos, con la llegada del Big data, Internet de las cosas y el abaratamiento de las tecnologías de la información, su crecimiento exponencial ha ido al alza, debido a que el COVID-19 se transmite por gotículas, al tocar a una superficie o a alguien infectado por el virus SARS-CoV-2, las autoridades globales ordenaron confinamientos. Desde trabajadores a estudiantes tuvieron que ir a casa y comenzar a hacer sus actividades ayudados por las tecnologías de la información y las comunicaciones, por lo cual, la mayoría de las actividades requerían de una conexión a Internet y una computadora con cámara web.

2.4 Seguridad

La seguridad en informática es un aspecto fundamental en nuestro entorno de trabajo, hogares, escuelas, etc. una actividad al que no se le debe de escatimar en dinero y tiempo que más adelante nos ayudara a mitigar el impacto y los riesgos a los que puedan estar expuestos nuestros equipos y redes.

Es importante resaltar que el usuario tiene cierta responsabilidad en cuanto al manejo de las tecnologías de la información, es decir, el ir desarrollando cierto

criterio acerca de un manejo responsable, a continuación, mencionan los servicios de seguridad informática (Trappe and Washington, 2006):

- Integridad: protección de los datos para que no puedan ser modificados, extraviados o eliminados de forma intencional.
- Confidencialidad: garantiza que la información privada no pueda ser accedida más que por las entidades autorizadas.
- Autenticación: confirma que la entidad es quien dice ser.
- No rechazo: asegura que la entidad no niegue o rechace la transmisión de la información.
- Control de acceso: garantiza que las entidades autorizadas utilicen los servicios en la organización.

Mismos que pueden ser cubiertos de acuerdo con las políticas de seguridad de la entidad protegida como puede ser una institución pública o privada, académica o industrial.

2.4.1 Mecanismos de Seguridad

Debido al crecimiento exponencial de las tecnologías de la información se ha dejado de lado los criterios de seguridad que nosotros como usuarios debemos de implementar en cada uno de los equipos de cómputo que se ocupan a diario, entre una de muchas causas es de la llegada del covid-19 en el mundo. Un gran porcentaje de las actividades tanto económicas, laborales, escolares, por mencionar algunas, en su mayoría migraron a plataformas digitales.

La humanidad realizo esfuerzos titánicos donde desarrolladores crearon apps y plataformas web para cubrir registros y tráfico, lo que permitió el surgimiento de nuevas formas para obtener información, una de ellas son los códigos QR que tienen como objetivo crear un código en forma de símbolo, el

cual, con una aplicación que cuentan los smartphones pueden leerlos y direccionar a una página en concreto ya sea para leer el menú de un restaurante o bien efectuar algún pago.

Desafortunadamente, esto dio pauta a que personas mal intencionadas (hackers black hat mejor conocidos como de sombrero obscuro), aquellos personajes que cuentan con habilidades extraordinarias en el manejo del cómputo pueden robar grandes cantidades de información, es decir, emplearon esta tecnología para implementar aplicaciones de segundo plano para poder sustraer información sensible de los equipos de los usuarios.

El gran crecimiento de la nueva incorporación de usuarios dejo de lado el cuidado de la seguridad en informática y su gran importancia. No existía un gran compromiso con ella hasta que varios usuarios empezaron a ser víctimas de los distintos mecanismos de intrusión, incluyendo las grandes corporaciones que sufrieron las consecuencias de no tener un plan de acción para este tipo de escenarios ya que poco a poco su operación a nivel organizacional fue creciendo al grado que fueron mitigando los criterios de seguridad que algunas por ley o por estrategia mercadológica debían seguir para no comprometer la integridad de sus empresas.

2.5 Cortafuegos y perímetros de seguridad

En la actualidad, a raíz del surgimiento del Covid-19, la humanidad fue orillada a incrementar su infraestructura tecnológica para poder proveer un buen servicio y que un porcentaje importante de la población experimentó una serie de problemas y retrasos a un nivel operacional trayendo en consecuencia una disminución en el cuidado y mantenimiento de las redes institucionales, bancarias y organizacionales, por mencionar algunas.

La implementación de perímetros de seguridad permite tener un control de acceso (entrada y salida) en una red. Los cortafuegos mejor conocidos como

firewalls son de las herramientas más utilizadas para este tipo de mecanismos de seguridad.

Con el paso de los años se ha manifestado que este tipo de técnicas que brindan la seguridad informática a una red no suele ser en mayor parte una solución definitiva ya que se requiere que quien las implementa pueda tener un buen control de ella. A medida que incrementa su tamaño y requerimientos a nivel de hardware y software, el administrador tiene la tarea de supervisar todos los elementos que la integra, así como determinar quién si cuenta con un control autorizado y a su vez, mantener criterios de seguridad oportunos ya que al incrementar su tamaño puede suceder que de manera arbitraria de paso al origen de puertas traseras vulnerando por completo la integridad de la red.

El firewall determina cuál de los servicios de red puede tener acceso dentro de ésta, es decir, quién puede entrar para utilizar los recursos de red pertenecientes a la organización.

Para que un firewall sea efectivo, todo tráfico de información a través del Internet deberá pasar a través de este donde podrá ser inspeccionada la información. El firewall podrá únicamente autorizar el paso del tráfico, y el mismo podrá ser inmune a la penetración. La configuración del firewall puede tener políticas restrictivas o permisivas (Randy Weaver and Dawn Weaver, 2008):

1) Restrictiva: se niega el paso del tráfico por default, es decir, la primera regla prohíbe el uso a cualquier servicio o cualquier puerto a menos que exista una regla que específicamente lo autorice.
2) Permisiva: para todo el tráfico es permitido el paso, a menos que exista una regla que lo prohíba.

2.6 Antivirus

El impacto del uso de las tecnologías de la información ha dado paso a que el tráfico de los datos incrementara. En los últimos tiempos, la incorporación y el fácil acceso a dispositivos como smartphones, tablets, laptops, entre otros, así como servicios en la nube ha dado pie a que personas con conocimientos en informática encuentren vulnerabilidades en software y hardware para usarlo ya sea de forma ética para aumentar su perímetro de seguridad o de forma mal intencionada para lograr sustraer datos así como penetrar las barreras de seguridad, buscan lucrar y delinquir con información altamente sensible, por mencionar algunos ejemplos mensajes, passwords, vídeos y fotografías los cuales pueden jugar un rol importante en contra de la víctima afectándola emocionalmente a un nivel social, personal, empresarial, escolar o comercial.

En la mayor parte de los casos un sector considerable de la población deja o discrimina los criterios de seguridad que debe incorporar a sus vidas. En términos prácticos no toman cuidado del impacto y alcance que tiene el depositar datos en los diversos servicios que la internet y la nube puedan proporcionar.

Los virus informáticos son códigos maliciosos que tienen la capacidad de ser desapercibidos por el usuario final, dependiendo de su configuración su capacidad de reproducción e infiltración radica en cómo está programado y para qué tipo de software va dirigido, en general, puede dividirse en aquel que puede infectar y replicarse y aquellos que están a la espera de que se ejecute el código por parte del usuario ingenuo (Nica Latto, 2022).

Un virus es un pequeño programa capaz de instalarse en la computadora de un usuario sin su conocimiento o permiso. Se dice que es un programa parásito porque ataca a los archivos o sectores de inicio de sesión y se replica para continuar su esparcimiento. Algunos se limitan solamente a replicarse, mientras que otros pueden producir serios daños a los sistemas. Nunca se puede asumir que un virus es inofensivo y dejarlo libremente en el sistema. Los virus tienen

diferentes finalidades, infectar, alterar, eliminar o mostrar mensajes, pero la finalidad es la misma: propagarse.

En el ciber espacio existen una amplia variedad de virus entre los más comunes se encuentra troyanos o caballos de troya, gusano, *spyware* (espionaje), *adware* (publicidad engañosa), *ransomware* (secuestro de equipo), entre otros. Debido a ello, la importancia de conocerlos y desarrollar herramientas para la protección de la información. Los antivirus son una solución preventiva y correctiva, permitiendo al usuario tomar precauciones y aplicar un plan de acción para combatir las amenazas que puedan presentarse en futuras ocasiones.

2.6.1 Características de un sistema antivirus

La respuesta a la pregunta ¿cuál es el mejor antivirus?, puede variar de un usuario a otro. Es evidente que para un usuario inexperto el término define casi con seguridad al software que es más fácil de instalar y utilizar, algo totalmente intrascendente para usuarios expertos, administradores de redes, etc. No se puede afirmar que exista un sólo sistema antivirus que presente todas las características necesarias para la protección total de las computadoras; algunos fallan en unos aspectos, otros tienen determinados problemas o carecen de ciertas facilidades. De acuerdo con los diferentes autores consultados, las características esenciales son las siguientes:

- Gran capacidad de detección y de reacción ante un nuevo virus.
- Actualización sistemática.
- Detección mínima de falsos positivos o falsos virus.
- Respeto por el rendimiento o desempeño normal de los equipos.
- Integración perfecta con el programa de correo electrónico.
- Alerta sobre una posible infección por las distintas vías de entrada (Internet, correo electrónico, red o discos flexibles).
- Gran capacidad de desinfección.

- Presencia de distintos métodos de detección y análisis.
- Chequeo del arranque y posibles cambios en el registro de las aplicaciones.
- Creación de discos de emergencia o de rescate.
- Disposición de un equipo de soporte técnico capaz de responder en un tiempo mínimo (ejemplo 48 horas) para orientar al usuario en caso de infección.

Existen sistemas antivirus que tienen, además, la característica de trabajar directamente en redes LAN y WAN, así como en servidores proxy. Un programa antivirus está compuesto por dos módulos principales: el primero denominado de control y el segundo de respuesta. A su vez, cada uno de ellos se divide en varias partes (Luis Armas Montesinos, 2003).

- Módulo de control: posee la técnica para la verificación de integridad que posibilita el hallazgo de cambios en los archivos ejecutables y las zonas críticas de un disco rígido, así como la identificación de los virus. Comprende diversas técnicas para la detección de virus informáticos y de códigos dañinos. En caso necesario, busca instrucciones peligrosas incluidas en programas para garantizar la integridad de la información del disco rígido. Esto implica descompilar (o desensamblar) en forma automática los archivos almacenados y tratar de ubicar sentencias o grupos de instrucciones peligrosas. Finalmente, el módulo de control también efectúa un monitoreo de las rutinas mediante las que se accede al hardware de la computadora (acceso a disco, etc.). Al restringir el uso de estos recursos, por ejemplo, cuando se impide el acceso a la escritura de zonas críticas del disco o se evita que se ejecuten funciones para su formateo, se limita la acción de un programa.

- Módulo de respuesta: la función alarma se encuentra incluida en todos los programas antivirus y consiste en detener la acción del sistema ante la

sospecha de la presencia de un virus informático. Se informa la situación mediante un aviso en pantalla. Algunos programas antivirus ofrecen, una vez detectado un virus informático, la posibilidad de erradicarlo.

Capítulo 3.
Honeypot

En el capítulo anterior, la información a través de la Internet aumentó considerablemente debido a la sencillez de las conexiones a través de cualquier dispositivo y al encierro que provocó la pandemia que obligó a todo el mundo a trabajar desde casa potenciando el uso del almacenamiento de la información en la nube.

Dado lo anterior, las herramientas de seguridad cobraron mayor sentido para proteger la información privada en su transmisión o en su almacenamiento. Una de esas herramientas son los cortafuegos (firewalls).

El uso de herramientas que brindan seguridad en informática, para este caso, el implementar firewalls por sí solo no es suficiente, existen una serie de amenazas externas ajenas a la red a la que se está operando que pueden comprometer los datos de quien la integra. A inicios de la pandemia y por el confinamiento se obligó a migrar a sistemas operacionales digitales para cumplir actividades burocráticas, empresariales, comerciales y escolares, actividades realizadas con premura y en muchos casos sin protección, acciones que permitieron que personas mal intencionadas y con habilidades especiales en informática aprovecharan la ocasión para espiar o cometer intrusiones en la red para sustraer información sensible de los usuarios para más adelante extorsionar a la víctima.

Las actividades más vulneradas fueron los procesos financieros tales como transferencias bancarias, robo de identidad y obtención de las claves de acceso. En primera instancia, con el engaño al usuario ingenuo a través de llamadas

apócrifas utilizando técnicas de ingeniería social para envolver a las víctimas argumentando anomalías en cuentas bancarias y poniendo de pretexto la imposibilidad resolver de forma presencial, logrando obtener la información bancaria necesaria para vaciar las cuentas. Las preguntas derivadas de estos ataques son ¿cómo es que sucede este tipo de escenarios?, ¿cuáles son las tácticas que implementan?, y ¿quién permite estas prácticas? En términos prácticos en la mayoría de los casos es una culpa compartida por una parte los usuarios al no tener un criterio y conocimientos amplios de lo que consiste la seguridad en términos de informática, las empresas por otra parte dejan de lado los criterios y protocolos de seguridad y enfocan todas sus energías en los procesos operacionales para cubrir la jornada laboral a la cual están sometidas, incurriendo en cierta medida a malas prácticas, como por ejemplo, no invertir en la protección de la red con herramientas especializadas que pueden dar solución y evitar puertas traseras que los intrusos usen y se beneficien de ellas.

Las diferentes herramientas que pueden implementarse, además, de un perímetro de seguridad es el Honeypot que es un recurso que permite integrar un servidor señuelo con datos de interés para el intruso, logrando engañarlo y detectar con antelación el ataque para así, activar un mecanismo de defensa o bien, si el ataque fue efectivo analizar la acción de este.

El uso de los Honeypot y su éxito radica en cómo está configurado, que tan robusto y poderoso sea, partiendo de que tan atractivo puede llegar a ser la información depositada en él, sin dejar de lado todos los mantenimientos oportunos, la estadística que permitirá el análisis del ataque y arrojará información relevante para el sistema o red que se está protegiendo. Con la información recolectada se puede aumentar o mejorar la seguridad de la red debido a la retroalimentación por parte del sistema, aprendiendo en cómo es que se desenvuelve el enemigo en cuestión, es decir, cuáles son las tácticas que usa para poder penetrar la red, si existen puertas traseras en el software que la compone para su oportuna detección y tomar acción a la brevedad posible.

El concepto de Honeypot surge hace más de 10 años y existen varias interpretaciones del término. Lance Spintzer (2002) lo define de la siguiente manera:

> *Un honeypot es un recurso cuyo valor está en ser atacado o comprometido. Se espera de un honeypot que sea probado, atacado y potencialmente explotado. Los honeypot no arreglan nada, proporcionan información adicional y valiosa.*

Reto Baumann and Christian Plattner (2002) en su artículo, propone una definición un poco diferente, centrándola en la distracción del atacante:

> *Un honeypot es un recurso que pretende ser un objetivo real. Se espera de un honeypot que sea atacado o comprometido. Su meta principal es la distracción de los atacantes y obtener información sobre los ataques y atacantes.*

Los honeypot constituyen una tecnología altamente flexible y adaptable a cualquier ambiente. El recurso para ser atacado puede ser de diversa índole como computadoras con diferentes sistemas operativos (Windows, Linux, Unix, etc.), un equipo de red (router, firewall, etc.), un servicio (correo electrónico, página Web, base de datos), etc. (Fernando Cócaro, Mauricio García y Maria José Rouiller, 2008).

3.1 Clasificación de los Honeypots

Se pueden clasificar de 3 formas, por *funcionalidad* (dependientes del objetivo perseguido), según el *nivel de interacción* que se permita o según *la plataforma de instalación*, real o virtual (Fernando Cócaro, Mauricio García y Maria José Rouiller, 2008). Es claro que estas formas de clasificación no son disjuntas entre las categorías ya que por ejemplo se puede tener un honeypot de bajo nivel de interacción y virtualizado (como es el caso del proyecto).

Funcionalidad. Clasificar un Honeypot por funcionalidad se refiere a cuál será el fin de la operativa del mismo. Esta clasificación se subdivide en dos categorías:

1. Honeypot de producción: Son utilizados para mitigar el riesgo en una organización distrayendo o haciendo más lento un ataque. Al detectar una intrusión se toman las medidas de contingencia oportunas (denegación de acceso a un origen determinado, limitación de capacidades de servicios, etc.).

2. Honeypot de investigación: El principal objetivo es el de recoger información sobre los ataques librados sobre el mismo, la forma de obtener comportamientos y técnicas utilizadas por los atacantes. Se utilizan generalmente en organizaciones o universidades, interesadas en aprender sobre las amenazas, como, por ejemplo, para fabricar un antivirus.

Nivel de interacción. Por interacción se entiende el grado de interacción que el atacante tiene con el sistema. Se definen tres niveles bajo, medio y alto. A medida que se escala en el nivel de interacción el riesgo de la red en la que se encuentra implantado también aumenta.

1. Bajo nivel de interacción (Low involvement): Este tipo de honeypot, simplemente simula la existencia de servicios como HTTP, FTP, TELNET, etcétera; escuchando peticiones y almacenándolas en logs, pero sin responderlas. Esto determina un sistema totalmente pasivo, que solo registra peticiones.

2. Nivel medio de interacción (Medium involvement): Este tipo de honeypot, brinda servicios más sofisticados, el modo de captar una mayor atención. Para ello, aumenta el grado de interacción con el atacante, respondiendo a las peticiones según un script que simula el comportamiento del puerto objetivo. Esto permite un análisis más minucioso de la actividad del atacante.

3. Alto nivel de interacción (High involvement): Este tipo de honeypot no simula ningún servicio, sino que presta servicios reales. Resultan muy

atractivos para los atacantes y también son los que más datos proporcionan sobre ellos y pueden poner en evidencia las técnicas utilizadas. A la vez son los más riesgosos, ya que un atacante podría penetrar el sistema apoderándose de él, y podría utilizarlo para atacar otros recursos con valor real. La Tabla 1 establece una comparación de las características de los honeypot según la clasificación por nivel de interacción.

Tabla 1. Comparación de Honeypots según su utilidad (Fernando Cócaro, Mauricio García y Maria José Rouiller, 2008).

Nivel de Interacción	Bajo	Medio
Servicio real	No	No
Nivel de riesgo	Bajo	Medio
Información obtenida	Conexiones	Peticiones
Riesgo-compromiso	No	No
Conocimiento del funcionamiento	Bajo	Bajo
Conocimiento del Desarrollo	Bajo	Alto
Tipo de mantenimiento	Bajo	Bajo

Nivel de implantación. Una tercera clasificación posible es de acuerdo con el nivel de implantación:

1. Física cuando se utiliza un equipo real para soportar la implantación de un honeypots.
2. Virtual: es cuando se utiliza software de virtualización para implantar el honeypot.

3.2 Distribución T-Pot

A continuación, se describen los honeypot que integran la distribución T-Pot Deutsche Telekom y cuáles son sus funcionalidades de cada una de ellas, es importante resaltar que han sido diseñadas para cumplir con distintos objetivos es decir los resultados capturados y los objetivos que desean alcanzar serán distintos para cada objetivo de acuerdo con Fabricio Gabriel Torrico Barahona y Pedro Hecht (2022).

Los Honeypots que incluye la distribución T-Pot son los siguientes:

- **ADBHoney**. El protocolo Android Debug Bridge (ADB) permite la comunicación con un dispositivo que tiene Android como sistema operativo (celulares, tabletas, televisores, etc.) e implementa varios comandos diseñados para ayudar al desarrollador. La comunicación se realiza a través de un cable USB, con amplios mecanismos de autenticación y protección; sin embargo, la conexión TCP/IP no tiene ningún tipo de autenticación y deja al dispositivo propenso a todo tipo de ataques.
- **CiscoASA Honeypot.** El Cisco ASA (Adaptive Security Appliance) es un dispositivo de seguridad que combina capacidades de firewall, antivirus, prevención de intrusiones y red privada virtual (VPN). Proporciona una defensa proactiva contra amenazas que detiene los ataques antes de que se propaguen por la red.
- **Citrix Honeypot** es un honeypot de baja interacción que, con la ayuda de Python, emula un dispositivo Citrix ADC que detecta y registra los intentos de exploración y explotación de la vulnerabilidad con CVE 2019-19781, misma que permite un ataque de Path Traversal. Publica el puerto 443 y los logs se almacenan en la ruta “/opt/citrixhoneypot/logs” del honeypot, volumen mapeado a “/data/citrixhoneypot/logs” del sistema anfitrión.
- **Conpot** es un honeypot de baja interacción que con la ayuda de Python emula Sistemas de Control Industrial SCADA, está diseñado para ser fácil

de implementar, modificar y ampliar. Proporciona una gama de protocolos de control industrial comunes, lo que permite construir un sistema capaz de emular infraestructuras complejas para convencer a un adversario de que acaba de encontrar un enorme complejo industrial. También proporciona la posibilidad de servir como servidor una interfaz hombre-máquina personalizada para aumentar la superficie de ataque de los honeypots.

- **Cowrie** es un honeypot SSH y Telnet de interacción media a alta diseñado para registrar ataques de fuerza bruta y la interacción de shell realizada por el atacante. En el modo de interacción media (shell) emula un sistema UNIX en Python, en el modo de interacción alta (proxy) funciona como un proxy SSH y telnet para observar el comportamiento del atacante a otro sistema. La implementación de T-Pot publica los puertos 22 y 23 y mapea los volúmenes "/data/cowrie/downloads", "/data/cowrie/keys", "/data/cowrie/log" y "/data/cowrie/log/tty" a las siguientes unidades del contenedor "/home/cowrie/cowrie/dl", "/home/cowrie/cowrie/etc", "/home/cowrie/cowrie/log" y "/home/cowrie/cowrie/log/tty".
- **DDoSPot** es una plataforma trampa para rastrear y monitorear ataques de denegación de servicio distribuido (DDoS) basados en UDP. La plataforma actualmente admite los siguientes servicios/servidores de honeypot en forma de complementos relativamente simples llamados pots: servidor DNS, servidor NTP, servidor SSDP, servidor Chargen, servidor UDP aleatorio/simulado.
- **Dicompot** es un honeypot que emula un servidor DICOM completamente funcional, haciendo uso del lenguaje de programación "Go". Publica el puerto 11112 (TCP) y guarda los logs en "/var/log/dicompot", que está mapeado a "/data/dicompot/log" del sistema anfitrión.
- **Dionaea** es un honeypot de baja interacción que tiene el objetivo de atrapar y obtener una copia del malware que explota vulnerabilidades

expuestas por los diferentes servicios emulados. Es el sucesor de otro honeypot denominado Nepenthes, está escrito en C, incorpora Python como lenguaje de scripting, utiliza la biblioteca “libemu” para emular la ejecución de instrucciones y detectar shellcodes; además, la última versión cuenta con soporte para IPv6 y TLS.

- **ElasticPot** es un honeypot que simula un servidor Elasticsearch vulnerable abierto a Internet. Utiliza ideas de otros honeypots, como ADBHoneypot (para compatibilidad con complementos de salida), Citrix Honeypot (para estructura general), Elastichoney, (para un ejemplo general de un honeypot de Elasticsearch). ElasticpotPY (para la idea de usar respuestas con script almacenadas en archivos) y Delilah (para ideas adicionales sobre qué emular). Publica el puerto 9200 y mapea el volumen “/opt/elasticpot/log” a “/data/elasticpot/log” del sistema anfitrión.
- **Endlessh** es un tarpit (servicio de red que intencionalmente inserta demoras en su protocolo, ralentizando a los clientes obligándolos a esperar) que envía muy lentamente un banner SSH aleatorio e interminable, manteniendo a los clientes SSH bloqueados durante horas o incluso días. El propósito es dejar que los scripts kiddies se atasquen en este tarpit en lugar de molestar a un servidor real. En la implementación de T-Pot mapea el puerto 2222 del contenedor al puerto 22 del sistema publicado y mapea el volumen “/data/endlessh/log” al directorio “/var/log/endlessh”.
- **Glutton** es un honeypot que actúa como proxy entre atacante y otro honeypot, proporcionando la capacidad de capturar, registrar y analizar el tráfico enviado. Básicamente, escucha todos los puertos y luego actúa de acuerdo con un archivo de reglas “rules.yaml”. Actualmente tiene soporte de proxy para SSH y TCP, además de ofrecer la posibilidad de inicio de sesión para el protocolo SSH. Para manipular los paquetes y cumplir la funcionalidad de proxy hace uso de varias librerías, siendo la principal

“freki”, la cual manipula los paquetes en modo de usuario haciendo uso de NFQueue (un objetivo de iptables e ip6tables que delega la decisión sobre los paquetes a un software de espacio de usuario).

- **Heralding** es un honeypot de baja interacción que tiene el objetivo de recopilar credenciales, para ello emula con la ayuda de Python diferentes servicios que solicitan inicio de sesión. Los servicios publicados son FTP en el puerto 21, SSH en el 22, Telnet en el 23, SMTP en el 25, HTTP en el 80, HTTPS en el 443, POP3 en el 110, POP3S en el 995, IMAP en el 143, IMAPS en el 993, Socks5 en el 1080, MySQL en el 3306, RDP en el 3389, PostgreSQL en el 5432 y VNC en el 5900. Para registrar los logs, mapea el volumen “/var/log/heralding” a “/data/heralding/log” del sistema anfitrión.
- **HellPot** es un honeypot basado en Heffalump que “envía al infierno” a los bots HTTP rebeldes. En particular, implementa un archivo de configuración config.toml, tiene registro JSON y viene con ganancias de rendimiento significativas. Los clientes afectados sufren consecuencias eternas ya que HellPot enviará un flujo infinito de datos que está lo suficientemente cerca de ser un sitio web real que podrían quedarse indefinidamente. Bajo el entendido del sufrimiento eterno, hay un motor de markov que arroja fragmentos de “El nacimiento de la tragedia (helenismo y pesimismo)” de Friedrich Nietzsche al cliente mediante fasthttp. En la implementación de T-Pot mapea el puerto 8080 del contenedor al puerto 80 del sistema publicado y el volumen “/data/hellpot/log” al directorio “/var/log/hellpot”.
- **Honeypots** en una solución desarrollada en Python que permite la implementación de 23 honeypots en un solo paquete PyPI con los cuales se puede monitorear el tráfico de la red, las actividades de bots y credenciales introducidas. Los resultados obtenidos se pueden registrar en una base de datos postgres, archivos, terminal o syslog y los servicios que emula son: dns, ftp, httpproxy, http, https, imap, mysql, pop3, postgres, redis, smb, smtp, socks5, ssh, telnet, vnc, mssql, elastic, ldap, ntp,

memcache, snmp, y oracle.La implementación en T-Pot publica los puertos 21, 22, 23, 25, 53 (UDP), 80, 110, 143, 389, 443, 445, 1080, 1433, 3306, 5432, 5900, 6369, 8080, 9200 y mapea el volumen “/data/honeypots/log” al directorio “/var/log/honeypots” del contenedor.

- **HoneyPy** es un honeypot de interacción baja o media, determinada por la funcionalidad de los complementos empleados. Está desarrollado en Python2 y emula los servicios “echo” en los puertos 7 TCP y UDP, Telnet Unix en el puerto 2323, Telnet Windows en el 2324 y Elasticsearch en el 9200, pero además puede emular otros servicios como DNS, NTP, SMTP, TFTP, Web etcétera. Si bien T-Pot emplea otros honeypot para publicar los servicios provistos por defecto, el volumen “/opt/honeypy/log” está mapeado a “/data/honeypy/log” para usarlo cuando se necesite.
- **Honeytrap** es un honeypot de baja interacción, escrita en “C” y destinado a detectar ataques contra servicios TPC y UDP. En su configuración predeterminada, se ejecuta como un demonio e inicia los procesos del servidor cuando se realiza un intento de conexión a un puerto. Tiene diferentes modos de operación disponibles que controlan cómo se manejan las conexiones. En modo “normal”, envía datos arbitrarios proporcionados en archivos de plantilla como un medio básico para emular protocolos conocidos. Otro modo de operación popular es el llamado “modo espejo”, en el que las conexiones entrantes se devuelven al iniciador, este truco elimina la necesidad de emular el protocolo en muchos casos. Un tercer modo, es el modo “proxy”, el cual permite el reenvío de sesiones específicas a otros sistemas, por ejemplo, honeypots de alta interacción. Para tomar los intentos de conexión entrantes, se hace uso de la función “NFQueue” de “iptables”, la cual a través de una regla coloca los segmentos TCP-SYN entrantes en una cola donde pueden ser recogidos por Honeytrap.

- **IPPHoney** es el Protocolo de Impresión de Internet o IPP (por sus siglas en inglés “Internet Printing Protocol”) es un protocolo de Internet especializado en la comunicación entre los dispositivos del cliente (computadoras, teléfonos móviles, tabletas, etc.) e impresoras (o servidores de impresión).
- **Medpot** es un honeypot que haciendo uso del lenguaje “Go”, emula la interfaz de programación de aplicaciones (API) FHIR.Su implementación en T-Pot publica el puerto 2575 y registra los logs en “/data/medpot/log/”, directorio que está mapeado al volumen “/var/log/medpot” del honeypot implementado con Docker.
- **RDPY** es un honeypot completamente implementado en Python (excepto el algoritmo de descompresión de mapa de bits que se implementa en C con fines de rendimiento) que emula el protocolo Microsoft RDP del lado del cliente y del servidor. RDPY se basa en el motor de red impulsado por eventos Twisted, admite la capa de seguridad RDP estándar, RDP sobre SSL y autenticación NLA (a través del protocolo de autenticación ntlmv2).
- **RedisHoneyPot** es un sistema honeypot altamente interactivo que admite el protocolo Redis. Está desarrollado en lenguaje Golang y simula la ejecución de los siguientes comandos: ping, info, set, get, del, exists, keys, flushall, flushdb, save, select, dbsize, config y slaveof. La implementación en T-Pot publica el puerto 6379 y mapea el volumen “/data/redishoneypot/log” a “/var/log/redishoneypot”.
- **Snare y Tanner Snare** es un honeypot de aplicaciones web (sucesor de otro honeypot denominado Glastopf) que emula vulnerabilidades (conocidas como “superficie de ataque”) a las que un usuario no autorizado puede acceder y posiblemente explotar.
- **ELK Stak** es una solución liderada por la empresa Elastic y compuesta por tres proyectos de código abierto: Elasticsearch, Logstash y Kibana.

Los tres productos forman una pila de un extremo al otro y conforman una herramienta de análisis de datos en tiempo real, que proporciona información procesable de casi cualquier tipo de fuente de datos.

- Logstash es un pipeline de procesamiento de datos del lado del servidor, que ingesta datos de una multitud de fuentes simultáneamente, los transforma y luego los envía para que algún motor de búsqueda lo utilice.
- Elasticsearch es un motor de búsqueda y análisis distribuido que acepta todos los tipos de datos (textuales, numéricos, geoespaciales, estructurados y no estructurados). Está desarrollado a partir de Apache Lucene y es conocido por sus API REST simples, naturaleza distribuida, velocidad y escalabilidad.

En este proyecto se utiliza un honeypot de investigación. En el capítulo siguiente se muestra la configuración y los escenarios establecidos de ataque y de protección.

Capítulo 4.
Mecanismos de seguridad propuestos

El siguiente ejemplo no busca motivar a realizar malas prácticas que puedan incurrir en un delito, el objetivo es ilustrar intentos de intrusión a la red virtualizada mediante fuerza bruta, poniendo en práctica ataques con fines educativos, se podrá detectar, evaluar, corregir accesos no autorizados y proponer protocolos de seguridad que aumenten la seguridad en informática.

En este apartado se empleó, la tecnología VMware para virtualizar el T-pot herramienta, que para el estudio de caso, se optó por un **honeypot de investigación** por su configuración que no pone en riesgo la integridad de otras redes vecinas ya que es una arquitectura descentralizada de la original, es decir, los datos capturados y cotejados a partir de los ataques nos permitirá tener un contexto más amplio de cómo es que funciona el T-pot todo bajo un esquema de trabajo seguro.

Es importante mencionar que el uso de un honeypot no otorga seguridad en informática por sí solo, el éxito de esta herramienta radica en una buena ubicación para implementar en una red. El cotejo y estudio de los movimientos ejecutados por los ciberdelincuentes contribuye a tener un departamento de inteligencia de amenazas de tal suerte que podremos contener los ataques y ganar tiempo a la hora de cuidar y respaldar los datos.

La configuración utilizada considera las características de la infraestructura en hardware y software:

- Linux Ubuntu: versión 22.04.1 LTS

- Kali Linux: versión 2022.3
- VMware Workstation 16Pro: Sistema que permite virtualizar sistemas operativos, recursos de red, servidores, entre otros.

Requerimientos para despliegue de honeypot / T-pot:

- Sistema Operativo Debian 11
- 4GB en RAM
- 32GB en Disco duro

Máquinas Físicas

- HP OMEN by HP Laptop 15-dc0xxx
    - Memoria: 16 GB
    - Capacidad de disco duro: 1.5TB
    - Procesador: Intel Core: i7 -8750H CPU 2.20GHz x12

- VAIO Sony Fit 14E
    - Memoria: 6GB
    - Capacidad de disco duro: 1TB
    - Procesador: Core i5

A continuación, se presentan los escenarios, indicando la configuración de la red, la configuración del honeypot y la vulnerabilidad encontrada después de ejecutar el ataque.

4.1 Escenario 1

Se dio de alta el honeypot / T-pot en el equipo de cómputo que se encuentra en su configuración inicial. Se recomienda dejar exclusivamente al equipo donde se planea instalar, ya que por sus características la herramienta tiende a consumir

muchos recursos, si existen otros programas desinstale o apague mientras T-Pot se encuentre activo.

En esta fase inicial, como se ve en la Figura 4.1, se propone la siguiente distribución que ayuda comprobar si la instalación funciona correctamente y si podemos hacer conexión en otro equipo anfitrión.

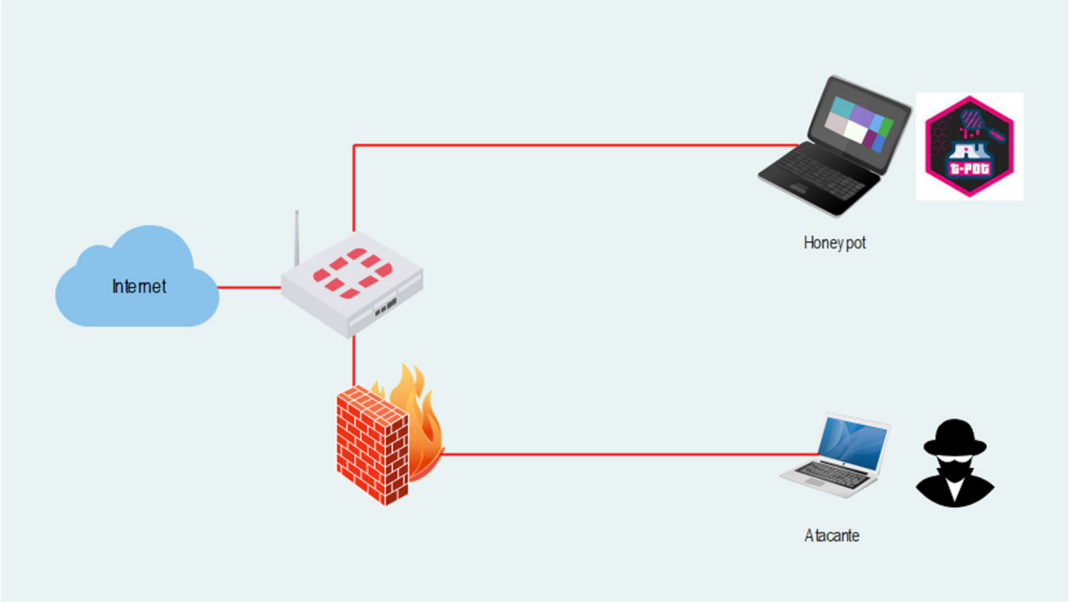

Figura 4.1 Topología propuesta. [Elaboración propia]

En la Figura 4.2 se muestra el menú principal del T-pot otorgado por el sistema una vez finalizada su instalación. Introducimos la dirección IP en otro equipo anfitrión para monitorear los datos que pueda arrojar de futuros intentos de intrusión.

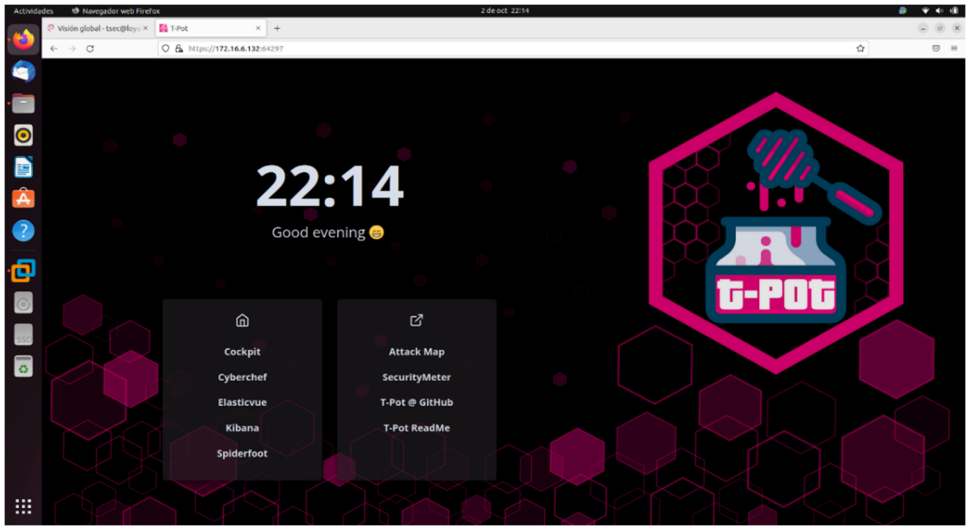

Figura 4.2 Menú principal T-pot, [Elaboración propia].

Una vez confirmado el funcionamiento del sistema trampa nos dirigimos a nuestro equipo atacante, es decir, disponemos del sistema operativo Kali Linux en nuestra máquina virtual, abrimos terminal y mediante el comando Nmap escaneamos el equipo anfitrión donde está activo el honeypot con la finalidad de encontrar una vulnerabilidad, como se puede observar en la Figura 4.3, mediante técnicas de fuerza bruta para poder ocupar ese puerto y conseguir un acceso no autorizado.

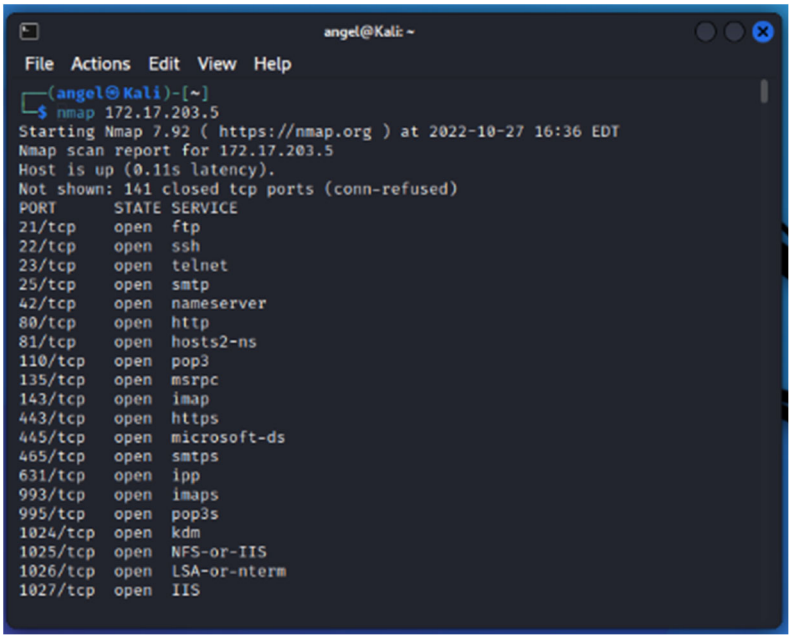

Figura 4.3 Escaneo de puertos usando el comando Nmap, [Elaboración propia].

Una vez localizado el puerto 22 procedemos a realizar el ataque dirigido al servicio SSH ya que es el puerto que se encuentra habilitado, para este caso el vector de ataque puede variar dependiendo de la configuración y naturaleza del proyecto.

En otra terminal activamos *mentasploit* como se muestra en la Figura 4.4, el objetivo es utilizar el puerto para sustraer el contenido de un archivo del equipo anfitrión.

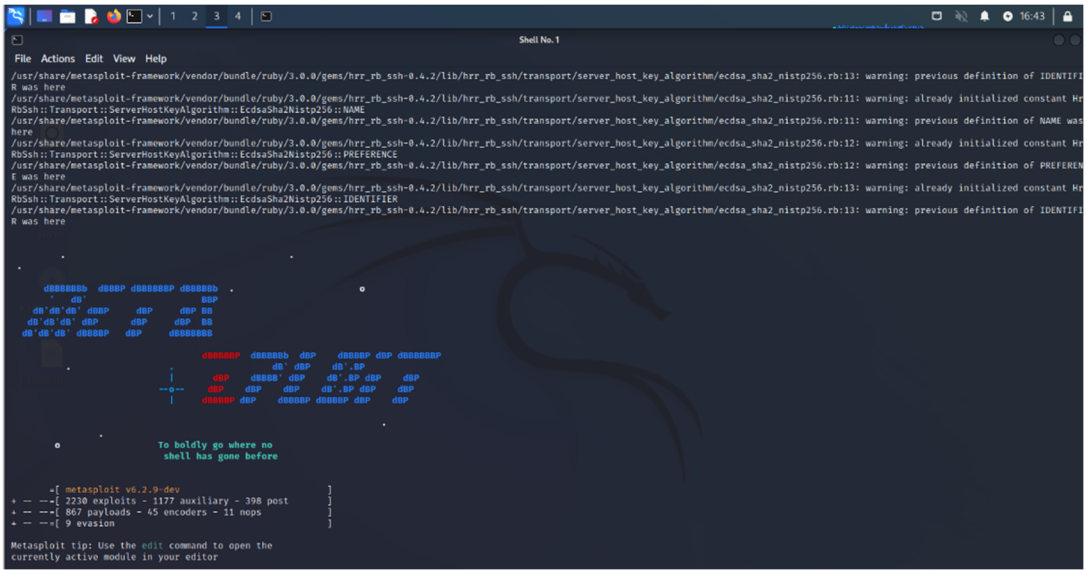

Figura 4.4 Metasploit permite saturar el puerto vulnerable, [Elaboración propia].

Una vez que, mediante la técnica de fuerza bruta, logramos romper los criterios de seguridad propuestos por el sistema trampa, nos dirigimos al menú principal y damos click en el apartado Kibana donde se encuentran todos los honeypot, que nos brinda T-pot, como se puede observar en la Figura 4.5.

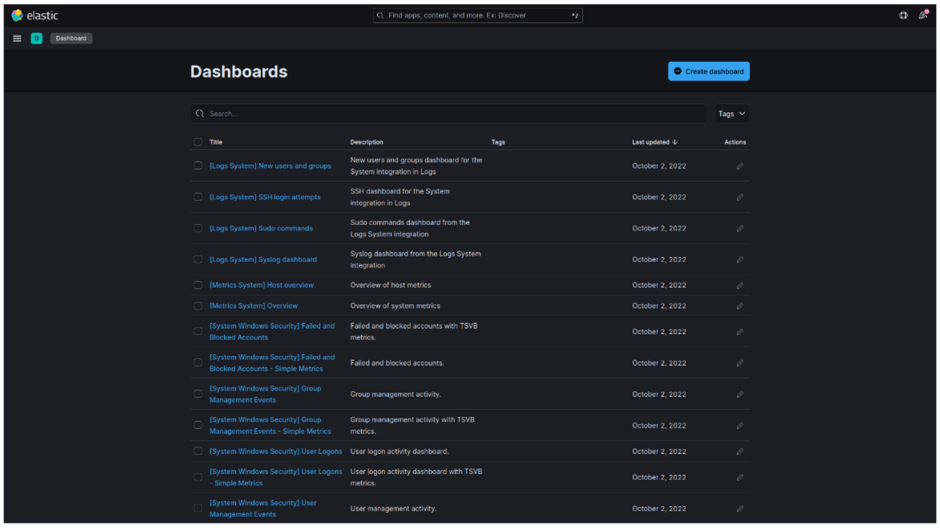

Figura 4.5 Menu kibana lista de honeypot disponible, [Elaboración propia].

Para visualizar los ataques de este primer escenario damos click en el honeypot cowrie, una vez dentro nos entrega un dashboard, la Figura 4.6 muestra la información de accesos no autorizados realizados por el atacante.

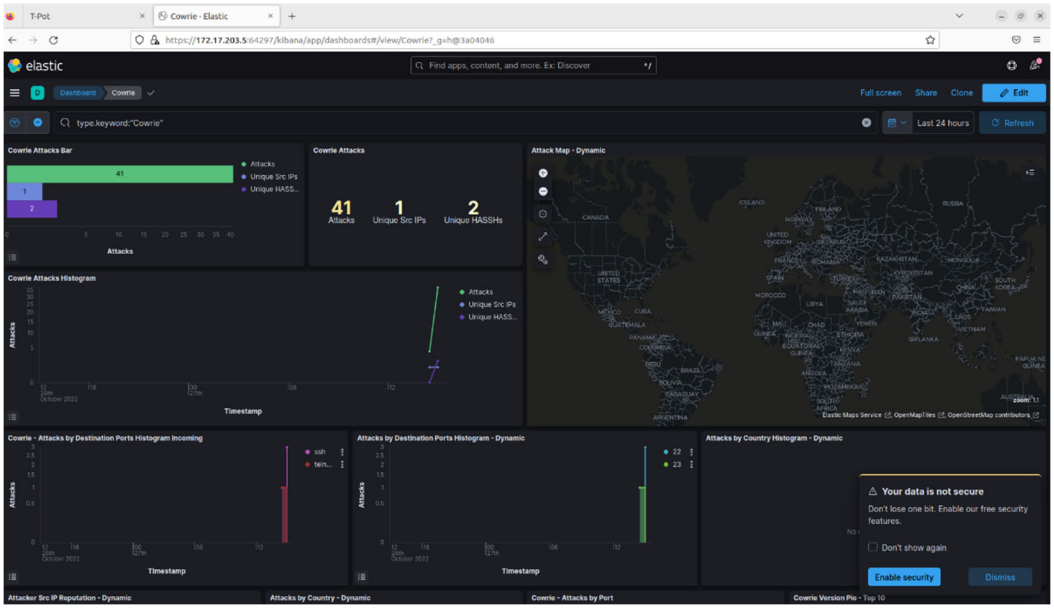

Figura 4.6 Dashboard datos de los ataques que realizo el atacante, [Elaboración propia].

En la Figura 4.7 se muestran los datos almacenados en la maquina anfitrión, podemos visualizar las contraseñas ocupadas por el administrador del equipo que el atacante puede emplear para tener acceso no autorizado.

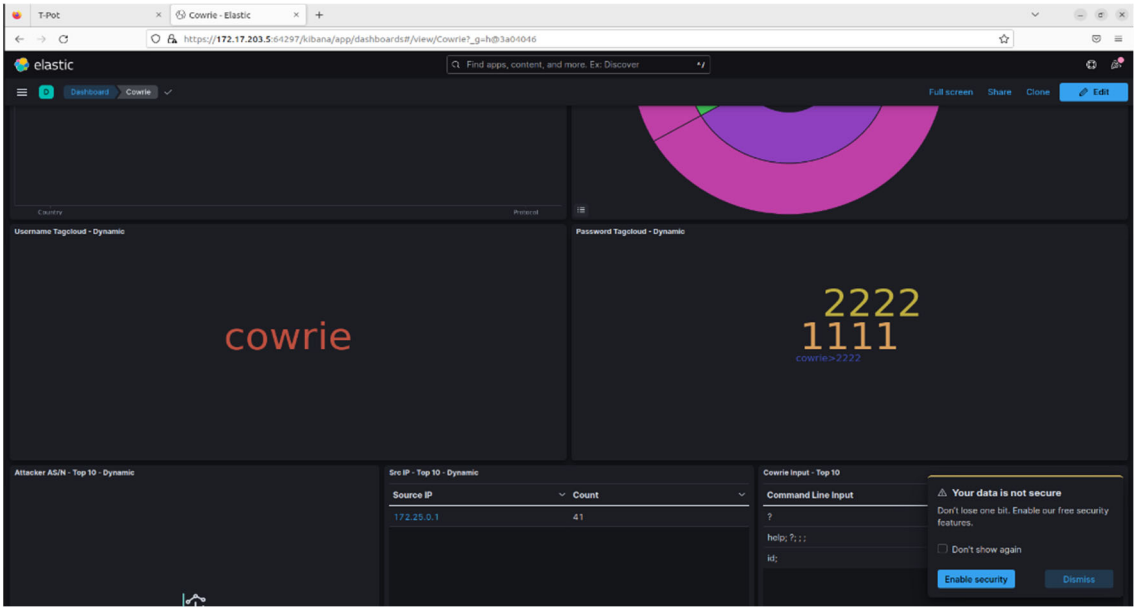

Figura 4.7 Contraseñas y movimientos hechos por el atacante, [Elaboración propia].

Lo que concluye que el Honeypot en su primera fase de despliegue es favorable ya que al realizar las primeras pruebas de ataques de fuerza bruta, el honeypot logró detectar los ataques emitidos por la máquina atacante.

De primera mano, se colocó en uno de los equipos señuelo un archivo sencillo con algunas contraseñas, que previamente se almacenaron para lograr penetrar la puertas traseras que pudieran estar habilitadas y tener acceso no autorizado. Con dicha información, el intruso puede hacer uso de ésta para vulnerar cuentas de usuario y sustraer datos que puedan ser sensibles, es por ello que se aconseja los siguientes parametros para brindar una mayor seguridad a la red:

- Cambiar de manera periodica contraseñas.

- No abrir la sesión en una IP de dominio público o que estén fuera del rango permitido por la empresa o institución, según sea el caso.
- Descentralizar las bases de datos en un lugar lo suficientemente seguro y resguardado.
- Configurar y administrar un firewall con el propósito de restringir el acceso no autorizado de una red local a una red vecina.
- Crear usuarios y contraseñas lo suficientemente robusta, con el propósito de que resulte difícil descifrar.

4.2 Escenario 2

En este caso, se han aumentado el número de maquinas virtuales atacantes, como se puede observar en la Figura 4.8, con la finalidad de ampliar más la inteligencia de amenazas, cada uno de los equipos esta configurado con Kali Linux preparado para hacer uso de las herramientas de pentesting y encontrar fallos en el sistema que más adelante se verán reflejados en el apartado Kibana sistema trampa.

Al contar con un mayor número de atacantes, se aumentan los servicios ofrecidos por el honeypot. Para esta ocasión, atacaremos los siguientes puertos SSH, FTP, HTTP, para ello mediante metasploit y patator.

Mediante las técnicas de fuerza bruta presentadas en el primer escenario se ejecutaron ataques de manera escalonada, el objetivo era llegar al archivo donde se almacenan los nombres de usuario y contraseñas del servidor los cuales aumentaron notoriamente, como se observa en la Figura 4.9.

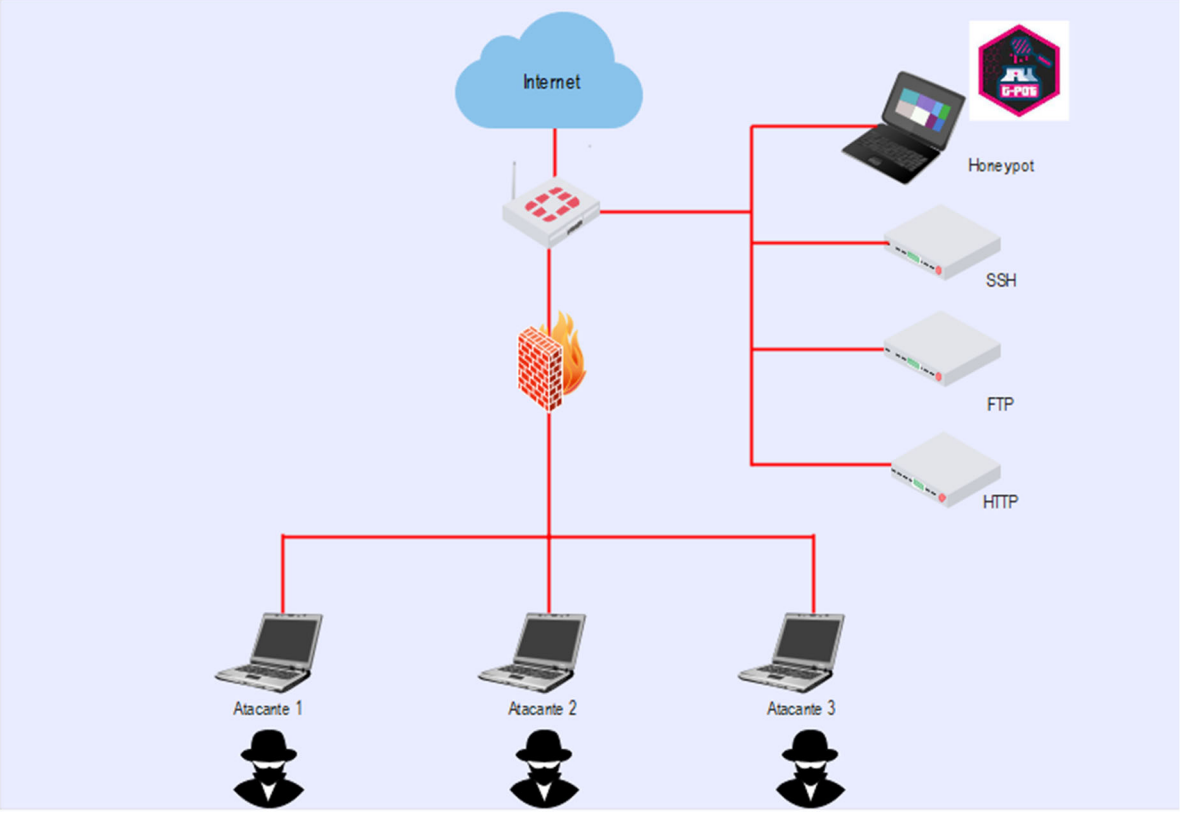

Figura 4.8 Topología propuesta con tres atacantes virtuales, [Elaboración propia].

Importante recalcar que el sistema nos empieza a brindar información contundente de nuestros atacantes como lo es la dirección IP y los comandos practicados una vez que el atacante ha logrado el acceso no autorizado y que exitosamente logra detectar el sistema trampa en tiempo real, información que más adelante nos ayudara a tomar decisiones acerca de cómo proteger la red.

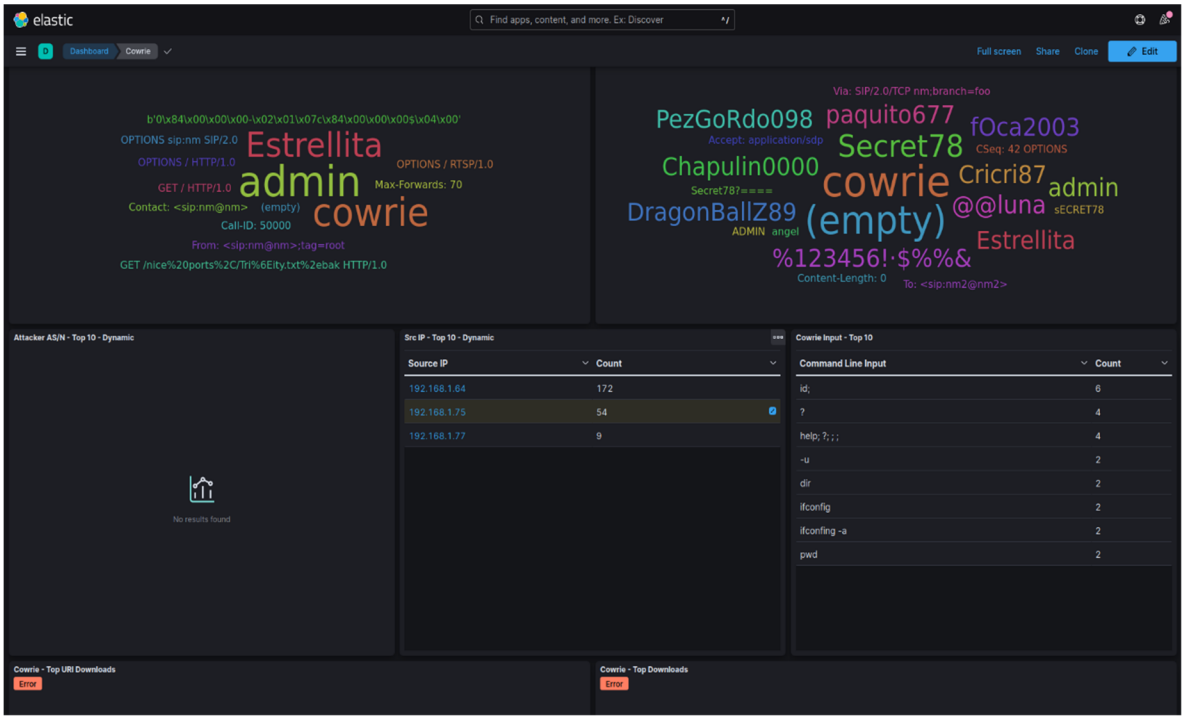

Figura 4.9 Muestra direcciones IP y comandos ejecutados por los intrusos para sustraer información sensible de nuestro equipo anfitrión, [Elaboración propia].

Para las siguientes series de ataques se utilizó el puerto 21, la Figura 4.10, la cual corresponde al honeypot Dionea, nos brinda las siguientes lecturas marcando una serie de 108 ataques y la procedencia de las direcciones IP de los atacantes.

Figura 4.10 Honeypot Dionea correspondiente al puerto TPC 21, [Elaboración propia].

Por último, tenemos las lecturas del honeypot suricata (Figura 4.11), este apartado brinda al administrador una lectura más amplia del número de atacantes en tiempo real, tiempo y fecha, así como, los lugares con más concurrencia y su procedencia; valores importantes para conocer patrones de conducta que tienen los ciberdelincuentes y tomar acciones para hacer más robusta la red.

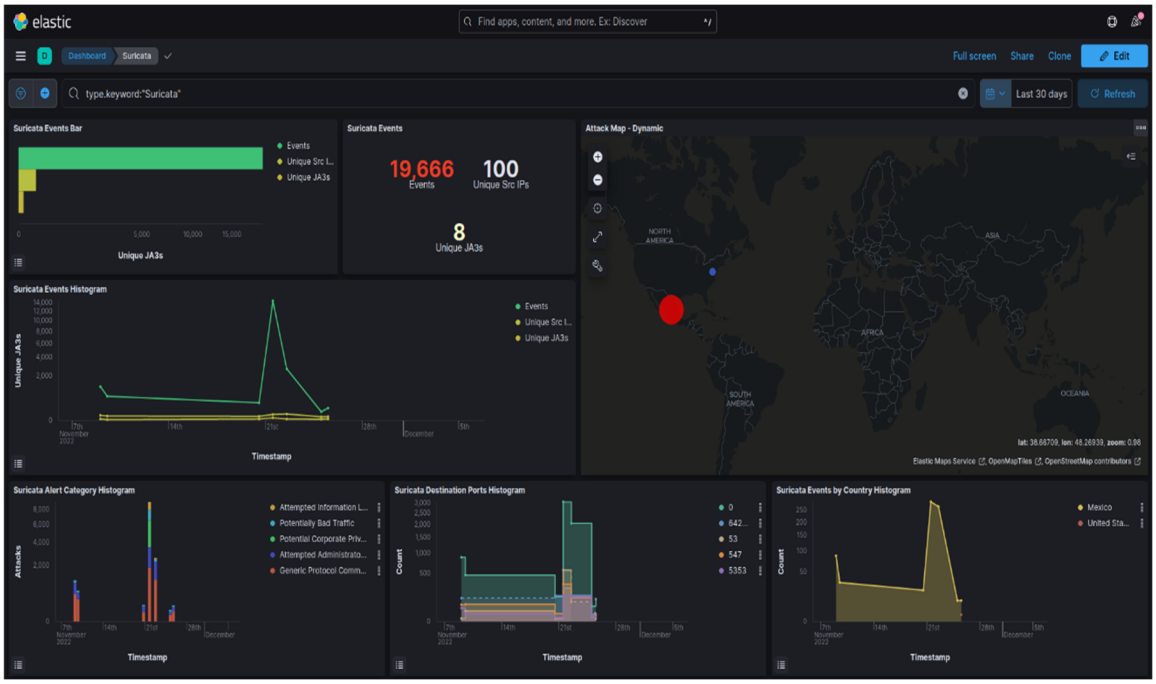

Figura 4.11 Muestra el número de taques de los últimos treinta días, así como la zona geográfica con más incidencia, [Elaboración propia].

En las Figuras 4.12 y 4.13 se muestra, a modo de complemento, el tipo de ataque efectuado por el ciberdelincuente y así como, un gráfico en pastel, el cual entrega la dirección IP (192.168.1.64) que ha realizado más números de ataques en los últimos treinta días. Es importante destacar que es uno de los sistemas trampa que registró más ataques.

Por último, pero no menos importante, la sección T-pot nos entrega una síntesis de los principales honeypots propuestos en este trabajo de investigación: El uso de esta tecnología nos proporciona una interacción en tiempo real de lo que sucede en nuestra red, al ser un sistema descentralizado notamos que si existe cierto riesgo, en el tiempo que se mantuvo activo registró posibles intentos de acceso a nuestra red de otras partes del mundo como se puede visualizar en la Figura 4.13, es decir, que existen amenazas latentes en la Internet que detectan estos riesgos, es por ello que se invita que una vez recopilada, estudiada y segmentada la información se cubran estos sesgos de seguridad a la brevedad posible.

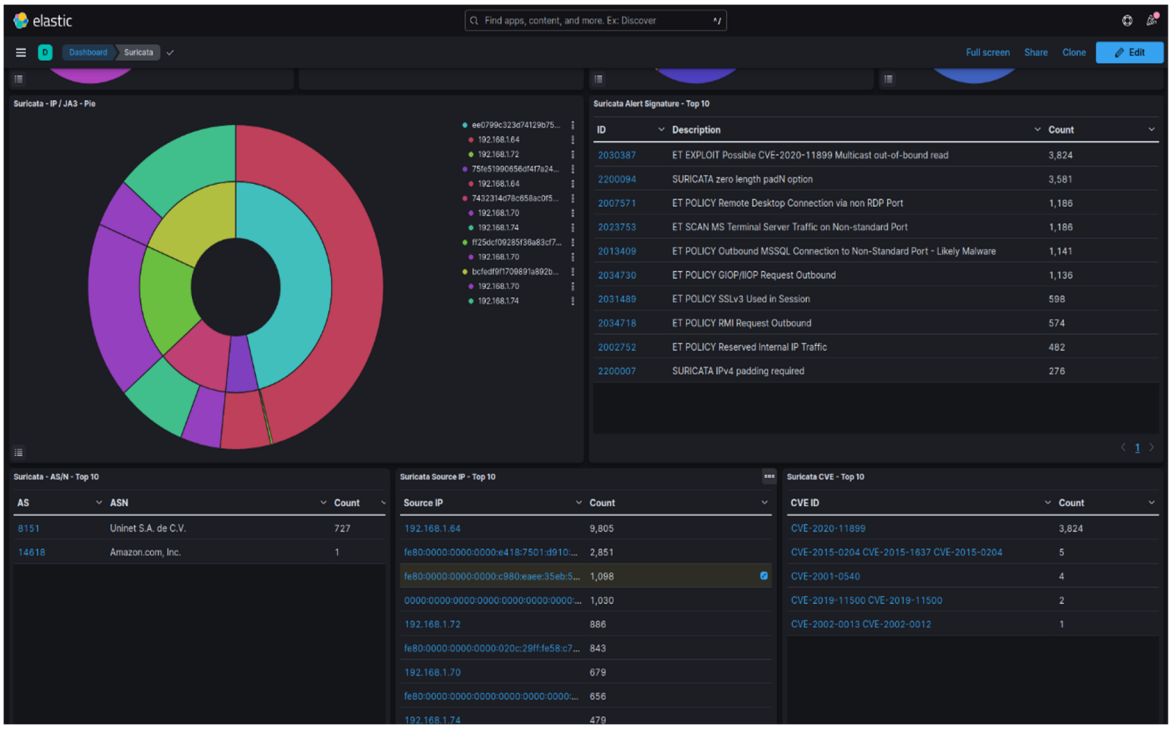

Figura 4.12 Suricata es un honeypot que registra ataques de intrusión en tiempo real, [Elaboración propia].

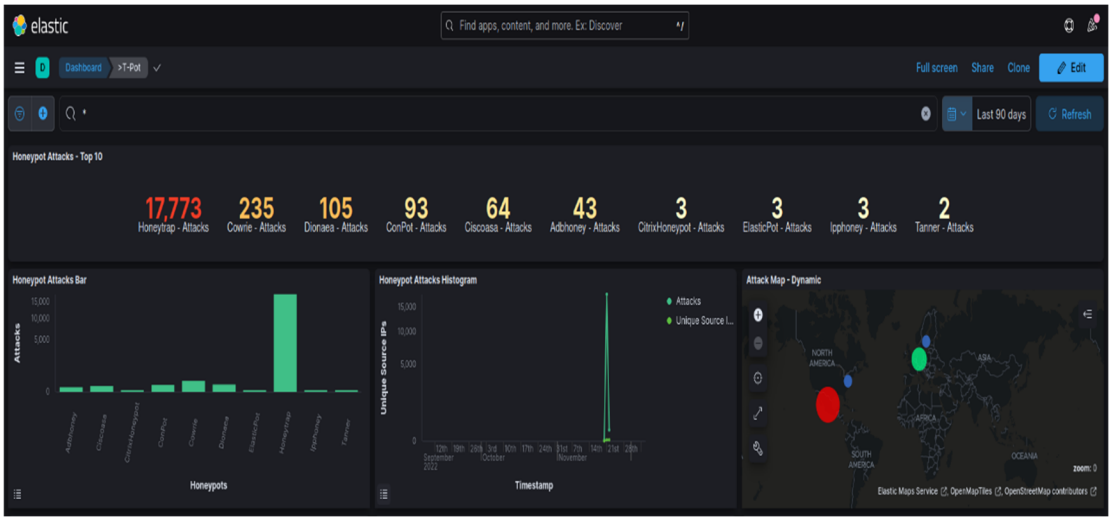

Figura 4.13 Interfaz T-pot muestra un conteo general y georreferenciación de ataques, [Elaboración propia].

También es importante recalcar a los usuarios y a quienes exploren estas tecnologías que cuenten con una buena estrategia de implementación, la razón es que si no existe un entorno contralado y descentralizado del original se puede cometer el fallo de que un atacante logre romper todo protocolo de seguridad, comprometiendo peligrosamente los datos alojados en dicha red, es por ello que se les exhorta a planificar muy bien además de no limar sus recursos en seguridad informática.

El objetivo es llegar a contar con una base de datos que nos proporcione inteligencia de amenazas, para que más adelante en futuras ocasiones que intenten comprometer y vulnerar nuestra red, contar con un plan de acción ante estas amenazas y tener un panorama más amplio de los ciberdelincuentes.

Gracias a los resultados obtenidos en este caso de estudio podemos determinar que un honeypot es un elemento de seguridad favorable para incorporar a nuestro ecosistema de seguridad en informática, para entender cómo es el modus operandi de nuestro enemigo, existe hardware y software que nos brindan seguridad pero no son suficientes, con el pasar de los años surgen nuevas tecnologías que hacen más grande el ciber espacio y nos deja con una mayor cantidad de puertas traseras que un posible ciberdelincuente puede aprovechar para hacernos daño o lucrar con los datos sustraídos.

Capítulo 5.
Conclusiones

Un Honeypot que por su traducción al español “servidor señuelo” o “sistema trampa” permite detectar ataques, vulnerabilidades y puertas traseras, por lo que, al integrar a nuestra línea de defensa esta herramienta facilita la investigación de modus operandi de un ciberdelincuente, es decir, crear inteligencia de amenazas partiendo de los datos capturados por el despliegue y configuración del honeypot, colocados estratégicamente en una arquitectura de red educativa, industrial, organizacional entre otros.

Una correcta planeación y ubicación de un honeypot proporciona información de que tipo de herramientas y técnicas emplean los intrusos, por ejemplo: la suplantación de identidad, pivoteo, fuerza bruta, software especializado en materia de ciberseguridad, favorece su trabajo de obtener acceso no autorizado, su meta es sustraer, secuestrar y lucrar con los datos obtenidos de servicios y servidores dados de alta en la nube o redes privadas.

La pandemia de la covid-19 orilló a reconocer y plantearnos que no podemos escatimar en gastos e infraestructura, el incremento de delitos informáticos en este periodo aumento exponencialmente debido a la incorporación de nuevos usuarios, modalidades de trabajo y estudio, donde el tráfico y almacenamiento de los datos fue mayor a lo que nos tenía acostumbrados, eventos que difícilmente van a ceder ya que surgen nuevos dispositivos y servicios que aumentan significativamente la creación de datos a lo que se le conoce como Big Data.

El despliegue honeypot de investigación, virtualizado para este caso de estudio resulto *positivo,* se optó por implementar T-pot un software open-source basado en Linux que en su contenido tiene varios honeypot, en esta ocasión se ocuparon Cowrie, Dionea y Suricata, los cuales permiten emular servicios y detectar ataques de tipo SSH, Telnet, vulnerabilidades en HTTP, FTP y registros en tiempo real de intentos de intrusión a nuestra red, respaldado por una extensión de Elastic Dashboard -kibana que también está integrada en la distribución T-pot y que nos entrega un informe detallado de las direcciones IP, hora y fecha, y la cantidad de intentos que realizaron los agentes externos para penetrar nuestra red. Por lo cual, el uso del T-pot permitió identificar los ataques y las estrategias del atacante.

5.1 Trabajo Futuro

En concordancia con los resultados obtenidos en esta investigación se plantea como trabajo futuro implementar el honeypot resultado de este proyecto en redes más consolidadas que sean académicas y de investigación, a fin de identificar las vulnerabilidades que se presenten en el tráfico de información en este tipo de redes.

Referencias

Emmett Dulaney (2011). Seguridad Informática, CompTIA Secutity +. Ediciones Anaya Multimedia, Madrid, España.

Carlos Muñoz Razo (2015). Cómo elaborar y asesorar una investigación de tesis. Pearson Educación, México.

Equipo Editorial Etecé (2022). Código Morse. Enciclopedia Concepto. Disponible en: https://concepto.de/codigo-morse/. Última edición: 31 de octubre de 2022.

Jorge Pérez y Carlos González Valderrama (2015). De Arpanet a la revolución digital. Revista Bit 200(1), pp. 200-205.

Nelly L. Hernández y Anderson S. Florez Fuente (2014). Computación en la Nube. Revista Mundo FESC, 8(1). pp. 46-51.

W. Trappe and L. Washington (2006). Introduction to Cryptography with Coding Theory. Second Edition, Pearson Prentice-Hall.

Randy Weaver and Dawn Weaver (2008). Guide to Tactical Perimeter Defense: becoming a security network specialist. Thomson Learning. Massachusetts, EU.

Nica Latto (2022). ¿Qué es un virus informático y cómo funciona?. Avast Academy. Recuperado de https://www.avast.com/es-es/c-computer-virus.

Luis Armas Montesino (2003). Análisis comparativo de los principales sistemas antivirus. Revista Cubana de Información en Ciencias de la Salud 11(5), pp.1-16.

Lance Spitzner (2002). Honeypots: Tracking Hackers. Addison-Wesley Professional.

Reto Baumann and Christian Plattner (2002). White Paper: Honeypots. Recuperado de: http://www.open.ch/en/downloads/whitepaper_honeypot.pdf

Fernando Cócaro, Mauricio García y Maria José Rouiller (2008). Proyecto Honeypots. Tesis publicada en el Instituto de Computación, Facultad de Ingeniería, Universidad de la República. Montevideo, Uruguay.

Fabricio Gabriel Torrico Barahona y Pedro Hecht (2022). Trabajo final de especialización, Facultad de Ciencias Económicas, Ciencias Exactas y Naturales e Ingeniería, Universidad de Buenos Aires. Argentina.