Indicadores de Compromisos Basados en Ataques a Servicios Web

Sebastián Suárez Restrepo

Instituto Tecnológico Metropolitano

Facultad de Ingeniería

Medellín, Colombia

2021

Indicadores de Compromisos Basados en Ataques a Servicios Web

Sebastián Suárez Restrepo

Tesis o trabajo de investigación presentado como requisito parcial para optar al título de:

Magister en Seguridad Informática

Director:

MSc Miguel Ángel Roldán Álvarez

Codirectora:

Ph.D. Paula Andrea Rodríguez Marín

Línea de Investigación:

Automática, electrónica y Ciencias Computacionales

Instituto Tecnológico Metropolitano

Facultad de Ingenierías

Medellín, Colombia

2021

Doy reconocimiento a DIOS que me dio la vida y aprovecho la oportunidad para agradecer a los docentes que me acompañaron en cada etapa y en el desarrollo del presente trabajo, teniendo en cuenta que su acompañamiento fue fundamental en el desarrollo del presente trabajo; a mis compañeros que siempre estuvieron presentes y como dejar a un lado mi familia y mi compañera de vida, que siempre ha confiado en mí proceso y en este me ha apoyado, a todos muchas gracias.

Resumen

En la actualidad el constante desarrollo tecnológico lleva a las organizaciones a enfrentar amenazas que atentan contra la integridad, confidencialidad y disponibilidad de la información, lo cual hace fundamental proteger la información y recursos de la organización. Con el fin de incrementar la seguridad, es necesario implementar esquemas para la protección de los servicios web, esta tecnología utiliza un conjunto de protocolos, estándares que sirven para intercambiar datos en aplicaciones y las compañías están cada vez más expuestas a los diferentes ataques realizados por intrusos, quienes identifican las vulnerabilidades o huecos de seguridad, que son aprovechados por los atacantes, generando robos de información, causando grandes daños económicos y de funcionamiento en la infraestructura de las empresas.

La presente investigación permitió identificar diferentes ataques informáticos basados en la definición de indicadores de compromiso, que fueron fundamentados en el aprendizaje obtenido de la forma de actuar de los atacantes. Mediante la implementación de una herramienta como la *honeypot*, en las cuales se emplean plataformas de virtualización, que permiten realizar la captura, el control, análisis y recolección de información sobre los diferentes ataques informáticos, logrando reconocerlos, identificando los ataques posibles, analizando los riesgos y las vulnerabilidades fáciles de explotar en los servicios web. Esta herramienta permite analizar el comportamiento de los diferentes ataques Web, determinando con ello, algunos controles para ambientes reales.

Así mismo, se presentan informes que describen los ataques registrados en la *honeypot*, para soportar la construcción de controles y la toma de acciones frente a las vulnerabilidades encontradas, mejorando así el sistema de seguridad de los servicios web de manera activa, conforme a los patrones de ataques más frecuentes, lo cual permitió disminuir el impacto generado en un ataque de seguridad.

Palabras Clave: Ataques Informáticos, Implementación *Honeypot,* Indicadores De Compromiso, Riesgos Y Vulnerabilidades, Servicios Web.

Abstract

At present, the constant technological development leads organizations to face threats that threaten the integrity, confidentiality, and availability of information, which makes it essential to protect the information and resources of the organization. In order to increase security, it is necessary to implement schemes for the protection of web services, this technology uses a set of protocols, standards that serve to exchange data in applications and companies are increasingly exposed to the different attacks carried out by intruders, who identify vulnerabilities or security holes, which are exploited by attackers, generating theft of information, causing great economic and operational damage to the infrastructure of companies.

The present investigation found to identify different computer attacks based on the definition of compromise indicators, which were based on the learning obtained from the attackers' way of acting. Through the implementation of a tool such as the *honeypot*, in which virtualization platforms are used, which allow the capture, control, analysis and collection of information on the different computer attacks, being able to recognize them, identifying possible attacks, analyzing the risks and easy-to-exploit vulnerabilities in web services. This tool will analyze the behavior of the different Web attacks, thereby determining some controls for real environments.

Likewise, reports are presented that describe the attacks registered in the *honeypot*, to support the construction of controls and the taking of actions against the vulnerabilities found, thus improving the security of web services in an active way, according to the patterns of more frequent attacks, which will eliminate the impact generated in a security attack.

Keywords: computer attacks, *Honeypot* implementation, Indicators of compromise, risks and vulnerabilities, Web services.

Contenido

Lista de Tablas

Lista de Figuras

Pág.

Lista de Símbolos y Abreviaturas

Abreviaturas

Abreviatura	Término
API	Application Programming Interface
CDB	Constant DataBase
CPU	Central Processing Unit
CVE	Common Vulnerabilities and Exposures
CWE	Common Weakness Enumeration
DMZ	DeMilitarized Zone
DNS	Domain Name System
DoS	Denial Of Service
EDR	End Point Detection and Response
FTP	File Transfer Protocol
HTML	HyperText Markup Language
HTTP	Hypertext Transfer Protocol
IDS	Intrusion Detection System
IEEE	Institute of Electrical and Electronics Engineers
IOC	Indicator of Compromise
IOT	Internet Of Things
IP	Internet Protocol
IPS	Intrusion Prevention System
JS	JavaScript
JSON	JavaScript Object Notation
LAN	Local Area Network
MHN	Modern Honey Network
NIST	National Institute of Standards and Technology
NVD	National Vulnerability Database
OSSEC	Open Source HIDS SECurity
RAM	Random Access Memory
REST	REpresentational State Transfer

Abreviatura	Término
RPC	Remote Procedure Call
SAST	Static Application Security Testing
SIEM	Security Information and Event Management
SIP	Session Initiation Protocol
SMB	Server Message Block
SOAP	Simple Object Access Protocol
SOC	Security Operation Center
SQL	Structured Query Language
SSH	Secure Shell
TELNET	Telecommunication Network
UDDI	Universal Description Discovery and Integration
URL	Uniform Resource Locator
VPN	Virtual Private Network
WAF	Web Application Firewall
WEB	World Wide Web
WLAN	Wireless Local Area Network
WS	Web Service
WSDL	Web Services Description Language
XML	Extensible Markup Language
XSS	Cross Site Scripting
XXE	XML External Entity Injection

Introducción

Las empresas utilizan software para realizar algunas de sus transacciones o incluso para soportar su razón de existir. Han pasado de utilizar los aplicativos locales a implementar servicios web, con el fin de obtener mayor calidad, productividad, interoperabilidad y tener disponibilidad de los sistemas desde diferentes dispositivos, esto incrementa el número de personas que pueden beneficiarse de las ventajas que ofrecen sus aplicativos [1]. Son cada vez más robustos los desarrollos implementados para lograr un correcto y seguro consumo de los servicios web necesarios para llevar a cabo una tarea específica, lo que ha facilitado y potencializado la integración entre tecnologías, que usando aplicativos locales no era viable [2].

No es viable porque los aplicativos locales no son interoperables y su uso se limita en el número de usuarios que pueden usar el aplicativo. Adicionalmente cuando las empresas tienen dos o más sucursales se presentan inconsistencias en la información y eventualmente ataques de seguridad. Tanto en el transporte de la información como en la gestión y uso de ésta, se presentan ataques que afectan los pilares de la seguridad informática. Los ataques informáticos siguen afectando las organizaciones, lo cual tiene serias implicaciones sobre su operación diaria e incluso sobre la continuidad en la existencia de sí mismas [3].

No obstante, es necesario indicar que los ataques informáticos han llegado a crear daños importantes aún en las empresas que más rigurosidad presentan en su esquema de seguridad, los atacantes mantienen una búsqueda incesante por violar toda estructura organizacional a través de la tecnología, con diferentes fines que siempre concluyen en el perjuicio de la víctima, es por esto necesario conocer que tan comprometido se encuentran los diversos sistemas de la organización antes, durante o después de un ataque de seguridad, característica que los indicadores de compromiso pueden llevar a cabo debido que estos, nos permiten preconfigurar las anomalías de los sistemas, obtener evidencias que indiquen las afectaciones que puedan existir en las redes o en los equipos y detectar nuevas vulnerabilidades [4].

Con base en lo anterior se encuentra la importancia de detectar, entender y controlar el comportamiento de los diferentes ataques y atacantes a los servicios web, lo cual es fundamental para comprender la forma cómo los sistemas se pueden proteger, para ello, sistemas como los *Honeypot* permiten la recolección de información clave sobre ataques y diferentes actividades mal intencionadas que se van ejecutando en los sistemas. Una vez obtenida la información es necesario

establecer los parámetros o patrones de ataques hacia los servicios Web, identificando si este tipo de servicios pueden estar o no comprometidos [5].

El análisis de los ataques a los servicios web ayuda a comprender los métodos de ataque más utilizados por los delincuentes informáticos, además de definir controles eficientes para contrarrestar los ataques, los indicadores de compromiso, implementados en un IDS o un firewall pueden tener un impacto considerable al proteger los sistemas de información y servicios web atacados, fortaleciendo los mecanismos de defensa que se encuentran implementados para estos sistemas [5].

Por lo tanto, se hace necesario definir Indicadores de compromiso que identifiquen ataques de seguridad en servicios web basado en la recolección de datos de una *honeypot* académica. Lo cual puede llevar a que las empresas y profesionales dedicados a salvaguardar la información, utilicen este tipo de técnicas y metodologías, con el objetivo de apoyar la seguridad de sus sistemas y de los servicios web que utilicen constantemente, teniendo como tendencia, mitigar riesgos de día cero dados los constantes monitoreos de la red que pueden llevarse a cabo con las *honeypot* y la posterior implementación de indicadores de compromiso y controles relacionados a los patrones de ataques encontrados.

A partir del problema planteado se desarrolló como objetivo general, “Definir indicadores de compromiso que, basado en la recolección de datos de una honeypot académica, identifiquen ataques de seguridad en servicios Web, generando recomendaciones para controlar y reducir los niveles de riesgo”. Para su cumplimiento se desarrollaron los siguientes objetivos específicos:

- Identificar las diferentes amenazas a servicios Web y las tres vulnerabilidades más relevantes mediante la caracterización de los diferentes ataques informáticos.
- Identificar diferentes indicadores de compromisos existentes que permitan, a partir de éstos, el desarrollo o ajustes de un set de indicadores para los servicios Web.
- Implementar un *honeypot* académico en ambiente controlado que permita la captura de los diferentes ataques informáticos y actividades no autorizadas.
- Evaluar los indicadores de compromiso, ejecutando los diferentes ataques informáticos, validando la información recolectada por la *Honeypot* con respecto a los IOC definidos, generando recomendaciones de controles para la reducción de los riesgos.

En este trabajo se realizó una enumeración de las amenazas y vulnerabilidades asociadas a los servicios web, teniendo en cuenta las investigaciones adelantadas por las entidades de seguridad que

estudian las vulnerabilidades de los sistemas de información. Se analizan los ataques que pueden ser utilizados para explotar dichas vulnerabilidades. Por otra parte, se identifican los indicadores de compromiso que son compartidos en el medio de la ciberseguridad, se realizó un rastreo de las *honeypot* más utilizadas y se implementó una *honeypot* de alta interacción capaz de identificar los vectores de ataques de una petición web, por último, se evalúan los indicadores de compromiso encontrados y se generan una serie de recomendaciones de controles técnicos.

1. Marco Teórico y Estado del Arte

En este capítulo, se encuentra la conceptualización del presente estudio, así como investigaciones relevantes, las cuales apoyan el desarrollo de la investigación.

1.1 Marco Teórico

El presente capítulo, corresponde a la sustentación y soporte conceptual en el cual se abordan los fundamentos teóricos transversales al proceso de investigación, así como estudios relacionados al presente proyecto, dividido en tres bloques temáticos: 1) Indicadores de compromisos, 2) ataques a servicios web; 3) *Honeypot*. Este proceso proporciona el lenguaje, los conceptos y los supuestos que orientan el estudio y lo dotan de sentido, en la medida que permiten que el investigador conecte los temas que están investigando con el cuerpo de conocimiento existente en el área.

1.1.1 Indicadores de Compromiso en Servicios Web

Los Indicadores de Compromiso IOC (por su significado en inglés, *Indicators of Compromise)* son datos o fragmentos de información, que bien sea, en un sistema informático, en el tráfico de la red o en un sistema operativo, indican una intrusión. Con los IOC, es posible percibir anomalías, reconocer maquinas afectadas y establecer el comportamiento de un incidente de seguridad presentado, además, describen la forma de como se afecta el sistema. No obstante, se pueden definir los controles técnicos o políticas de seguridad necesarias a partir de los IOC, con el objetivo de prevenir futuros ataques [6].

Para un equipo de gestión de incidentes, los indicadores de compromiso se pueden presentar en diez tipos [6]:

- **Tráfico de Red Inusual:** Conociendo el comportamiento normal de la red y sus variaciones, se puede identificar fácilmente posibles incidentes de seguridad. Con un monitoreo constante de la red, es posible controlar que no se esté permeando o extrayendo información, por medio de ataques de tipo persistente o backdoor.

- **Anomalías en la actividad de las cuentas de tipo super usuario:** Este tipo de actividades, genera afectaciones en la integridad de la información, por lo tanto, es necesario monitorear constantemente el uso de estas cuentas, cuando se generan irregularidades como: Horarios inadecuados, cambios en el tipo y el volumen de la información consultada, intentos de autenticación fallidos e intentos masivos de eliminación de datos. Estos monitores llevarán a la detección temprana de los incidentes potenciales de seguridad.

- **Irregularidades Geográficas:** Los atacantes, cambian constantemente su IP antes de realizar los intentos de autenticación, a los sistemas de información. Este tipo de comportamiento se considera anómalo, se monitorea para mantener un mejor grado de seguridad. Este indicador, evita que una cuenta pueda ser suplantada desde una ubicación no autorizada.

- **Incremento de Consulta en las Bases de Datos:** Este indicador, se identifica cuando, el atacante obtiene acceso y busca extraer la información de interés, lo que representa un incremento en los tamaños de respuesta de las consultas a las bases de datos y a los servicios web; supervisando el tamaño de las consultas, se minimiza el impacto que esto pueda generar.

- **Número Elevado de Solicitudes para un Mismo Archivo:** Encontrar un número significativo de peticiones o intentos fallidos de acceso, sobre un mismo archivo, refleja un posible incidente de seguridad. Al restringir un número de peticiones, se logra contener un ataque de fuerza bruta.

- **Tráfico Irregular en Puertos de Aplicaciones:** Los atacantes utilizan puertos inusuales para la exfiltración de datos, por ende, es necesario conocer qué aplicaciones se utilizan entre sí, y cuáles puertos usan para la comunicación. Un indicador de compromiso puede alertar al momento que una aplicación use un puerto diferente al establecido por el administrador del sistema.

- **Cambio Sospechoso en los Archivos del Sistema y sus Registros:** Teniendo en cuenta que en los ataques de seguridad o en el desarrollo de un incidente, surgen cambios, en los archivos y sus versiones; con la finalidad de crear *Backdoors* para ataques persistentes. Se realizan modificaciones en los registros, para tener el control de los sistemas o inicio de

sesiones con el objetivo de generar elevación de privilegios. Con la ayuda de los IOC, se genera una línea base de los archivos y los registros para detectar cualquier tipo de modificaciones.

- **Consultas Inusuales DNS:** En el momento en el cual se identifiquen solicitudes inusuales al servidor DNS, se debe generar una alerta, por medio de un indicador de compromiso, para evitar ataques de tipo reverso, los cuales transmiten respuesta hacia la IP del atacante.

- **Autenticaciones Fallidas sobre Herramientas de Seguridad Perimetral:** Para evitar que se realicen intentos de conexión, desde una red externa, o mediante una fuerza bruta a las herramientas de seguridad perimetral, se pueden utilizar IOC, permitiendo las conexiones, ya sea, desde la red privada o por medio de VPN y limitando el número de autenticaciones fallidas.

- **Acceso Remoto de un Usuario con VPN:** Se deben restringir, las conexiones simultáneas en el uso de esta plataforma, para controlar la posible suplantación de identidad, y delimitar la geolocalización, con el propósito de que no se hagan efectivas las conexiones desde otras ciudades o países.

Cabe resaltar que existen diferentes estándares para la generación y escritura de IOC, a continuación, se mencionan unos de los más destacados y utilizados en el mercado actual.

- *Incident Object Description Exchange Format* (IODEF): Creado por miembros del *IETF Extended Incident Handling (INCH) Working Group*, que forman parte del *IETF Security Area*. El formato de intercambio de descripción de objetos de incidentes expone información acerca de seguridad de la información, aportando todo un entorno cuyo objetivo es compartir información acerca de los incidentes presentados, como por ejemplo identificación de redes e IP, servicios ejecutados en los sistemas, hosts, comportamientos y tipos de intrusiones informáticas, con los equipos de respuesta a incidentes de seguridad informática (CSIRT). Al compartir esta información se generan nuevos tipos de incidentes conocidos con el fin de desarrollar las estrategias y controles correctos para afrontarlos [7].

- *Open Indicators of Compromise (OpenIOC):* Creado por la empresa MANDIANT, es un estándar que tiene herramientas gratuitas, generalmente utilizado para definir de forma clara y organizada la actividad de los atacantes a través del análisis de un incidente de seguridad o de un análisis de amenazas a una infraestructura, utilizando un fichero con extensión IOC que contiene un esquema de tipo XML para describir las características técnicas que identifican a una amenaza y sus métodos de ataque [7].

- *Oasis Cyber Threat Intelligence (CTI):* Basado en el intercambio de información confiable de manera automática, con el fin de prevenir y defender en tiempo real las infraestructuras empresariales contra las amenazas cibernéticas. Está compuesto por tres estándares STIX, TAXII y CybOX los cuales previenen, detectan y remedian ciberataques. Determinan como actúan los ciberdelincuentes, identificándolos e indicando que acción se necesita para contenerlos [8].

- *Cyber Observable Expressión (CybOX)*: Creado por miembros del MITRE, se utiliza para intercambiar información sobre incidentes de seguridad tales como IOC, evaluación de amenazas, gestión de registros y análisis de programa maligno (malware). [9].

- *Structured Threat Information Expression (STIX):* (En español expresión estructurada de información sobre amenazas), este estándar usa un lenguaje que permite identificar las actividades realizadas por un delincuente cibernético, determinando, la mejor forma que se puede emplear para defenderse de ellos. Es un componente que acepta programas o herramientas el cual permite aprovechar la información descrita en otros formatos, por ejemplo, OpenIOC, reglas de Snort, sistema de detección de intrusos en red y reglas de YARA (identifican malware en base a patrones de texto o binarios y el uso de expresiones booleanas para determinar su lógica) entre otros [8].

- *Trusted Automated Exchange of indicator Information (TAXII)*: (En español, intercambio automatizado confiable de información de indicadores), define mensajes XML sobre HTTP(S) y servicios que se pueden combinar para ser compartidos con los socios que las compañías consideren sus aliados a nivel de seguridad. [8].

- *Malware Attribute Enumeration and Characterization (MAEC):* Creado por MITRE, analiza a alto nivel el malware teniendo como resultado, mitigación de riesgos y respuestas a los incidentes mucho más rápidas con el fin de compartir mediante la comunidad MAEC los casos de uso de tipo IOC resultantes que permite a los investigadores de seguridad capturar un paquete completo de datos sobre muestras de malware [10].

- *Collective Intelligence Framework (CIF):* Creado por CSIRT, es un sistema de gestión para las amenazas de ciberseguridad a nivel mundial, el cual reúne listas blancas y negras de IP, URL, DNS, proxys, spam y genera sobre ellos diferentes IOC, se caracteriza por la capacidad de respuesta y las excelentes fuentes de inteligencia, fácilmente puede ser integrado con SIEM [11].

- *Open Threat Exchange (OTX):* Creado por *AlienVault* (Ahora AT&T Cybersecurity) para la gestión de ciberataques, este estándar indica que tan expuesta se encuentra una organización en la red, analiza el tráfico inusual en la red y sus comportamientos, enfatiza en compartir entre empresas y entidades gubernamentales los diferentes IOC que sean identificados con el fin de tener un medio cibernético cada vez más seguro [12].

- *Vocabulary for Event Recording and Incident Sharing Framework (VERIS Framework):* (En español vocabulario para la grabación de eventos y el intercambio de incidentes), es un estándar que posee diferentes incidentes de seguridad previamente definidos y caracterizados, el cual permite de manera gratuita a las compañías su análisis y adaptación, al igual que les permite compartir datos de incidentes, para apoyar la investigación por medio de una base de datos comunitaria Veris, catalogo abierto de más de 1200 incidentes de violación de datos divulgados públicamente, desde esta base de datos se puede traducir su información en IOC con el fin de darle una mejor gestión a los riesgos y amenazas [13].

1.1.2 Servicios Web

Fueron diseñados en los años noventa, con el fin de mejorar la interoperabilidad de la comunicación entre las aplicaciones. Se trata de métodos o funciones realizadas con un fin específico y que puede ser invocado desde cualquier programa o equipo mediante el protocolo de internet HTTP. [14]. En este sentido, un servicio web bien diseñado debe cumplir estas características mínimas [15]: estar escrito en un leguaje comprensible el cual acepte comunicación de tipo HTTP, debe encontrarse descrito de modo que quien vaya a consumir su recurso sepa fácilmente cual es la funcionalidad concreta para la cual fue diseñado y no debe tener claves para su conexión a menos que no desee ser hallado [15].

Teniendo en cuanta, los que reflejan mayor demanda en el mercado, basado en las comunicaciones de la mayoría de los sistemas y aplicaciones que consumen este tipo de recursos, los cuales son XML-RPC, UDDI, SOAP y REST [16].

El *XML-RPC (Remote Procedure Call)* utiliza instrucciones básicas para intercambiar datos entre una amplia variedad de dispositivos en una red. Utiliza HTTP para transferir rápida y fácilmente datos y comunicar otra información del cliente al servidor normalmente mediante el puerto 80 [17]. Por su parte, el *UDDI (Universal Description Discovery and Integration)*, es un protocolo que cuenta con dos funciones principales, define como se comunican los clientes mediante registros y verifica los conjuntos de servicios duplicados, optimizando las transacciones digitales utilizadas para detallar, publicar y descubrir servicios web [18].

El indicie de *SOAP (Simple Object Access Protocol)* es un protocolo de servicio web que proporciona salida para que las aplicaciones aun siendo diferentes se comuniquen entre sí usando XML, intercambiando datos y documentos mediante los protocolos de transporte HTTP o SMTP [19]. Por su parte, el *REST (representational state transfer),* es una arquitectura para el diseño de servicios Web, fundamentado en solicitud y respuesta que proporciona comunicación y conectividad entre dispositivos e Internet para tareas basadas en API. Normalmente los servicios REST como protocolo de soporte usan HTTP [19].

Es menester resaltar que, en la implementación de los servicios web quedan expuestas debilidades, bien sea de tipo físico, lógico o una combinación de ambas, las cuales representan para los atacantes una oportunidad de vulnerar el sistema, afectando uno o más de los pilares de la seguridad de la información aprovechando dichas debilidades. Respecto a las vulnerabilidades de los servicios web

se encuentran: Inyección de código, Perdida de autenticación, Exposición de datos sensibles Entidades externas de XML (XXE) este tipo de vulnerabilidad permite a un atacante escanear la red, privada de un servidor, Pérdida de control de acceso, Configuración de seguridad incorrecta, Secuencia de comandos en sitios cruzados (XXS), Deserialización insegura, Componentes con vulnerabilidades conocidas, Registro y monitoreo insuficientes, Fallas Criptográficas y, Salteo de autorización a través de entradas de datos controladas por el usuario (IDOR & Path Traversal).

Con relación a la Inyección de código, se indica que, para explotar esta vulnerabilidad el agresor busca al principio obtener información de las bases de datos de un sistema o página definida, como por ejemplo, sus tablas, usuarios, cifras, entre otras; al instante de tener clara esa información su objetivo es alterar las consultas que hace el sistema a la base de datos por medio de los cuestionarios y ocupaciones de consulta al sistema, todo lo mencionado con el objeto de obtener un beneficio o una información fundamental con un definido fin. [20]. La mitigación de esta vulnerabilidad se presenta en el momento de analizar el código de las aplicaciones antes de desplegarlas a producción utilizando herramientas que de manera automática verifican si es posible llevar a cabo una inyección [21].

Así mismo, la pérdida de autenticación se reconoce como el proceso mediante el cual los atacantes buscan identificar la contraseña de usuarios conocidos o no en el sistema de información objetivo, se trata de descifrar los *hashes* de contraseñas y suplantar la identidad posteriormente de alguno de los perfiles registrados, se realiza mediante fuerza bruta o robando la sesión de un usuario ya registrado en el sistema. No es necesario obtener todos los usuarios al momento de realizar el ataque, con un super usuario que logren suplantar, se podría realizar bastante daño a las organizaciones [22]. Esta vulnerabilidad deja de ser explotada mediante la correcta definición de políticas de seguridad, para el manejo de contraseñas, sesiones, incluso desde la misma programación de los sistemas se pueden corregir importantes fallas a nivel de seguridad. Algunas vulnerabilidades de pueden conocer mediante la revisión de las CWE registradas por MITRE [23].

La exposición de datos sensibles, indica que las reglas de negocio son las que definen las estrategias y la forma de trabajar de cada compañía, es lo que les diferencia entre otras empresas, por consiguiente, es importante no tener información sensible expuesta a ataques informáticos, los archivos de configuración deberían tener la información cifrada, los canales para transmisión de datos en las operaciones tener los controles mínimos para evitar el robo. [24]. Utilizando encriptación, protocolos de comunicación en la red, algoritmos y claves criptográficas maduros para

salvaguardar la información, teniendo en cuenta que incluso existen leyes que obligan a la protección de los datos según la actividad económica de las empresas [25].

Al analizar la vulnerabilidad “entidades externas de XML (XXE)”, se establece que esta permite a un atacante escanear la red privada de un servidor, extraer datos del servidor o realizar una fácil denegación de servicios creando un bug de procesamiento en los códigos de las peticiones de tipo XML y SOAP al momento de consumir los servicios WEB utilizados en las aplicaciones [26]. Las mejores prácticas para mitigar este tipo de vulnerabilidad son: la implementación de una herramienta de tipo SAST para percibir intentos o ataques de tipo XXE, implementar listas blancas y negras para la utilización de cabeceras y nodos dentro de las páginas y servicios web consumidos, limitar las cargas de XML dentro del sistema para acotar el público que puede generar la explotación de una amenaza, realizar la implementación de Firewalls de aplicaciones web, realizando un monitoreo y bloqueo constante de ataques de tipo XXE. Dentro de la CWE (CWE 611: Improper Restriction of XXE), se puede ver como se presenta este tipo de vulnerabilidad [27].

Otra de las vulnerabilidades de la Pérdida de control de acceso: existe la posibilidad que se bloquee el control de acceso dentro de las aplicaciones y los servicios expuestos, lo que hace viable a un atacante realizar modificaciones en la configuración de usuarios y del servicio web en su estructura y forma de actuar dentro del sistema de información [28]. La correcta configuración de perfiles y grupos dentro de los directorios activos al momento de la asignación de roles y tareas puede disminuir el riesgo de que se ejecuten actividades no comunes dentro de un determinado usuario En la CWE-284: *Improper Access Control* se aprecia como se aprovechan de esta vulnerabilidad [29].

En otra línea, la vulnerabilidad “Configuración de seguridad incorrecta” se expresa de la siguiente manera: actualmente, se tienen en las empresas grandes sistemas y herramientas de seguridad con el fin de proteger la información y las transacciones desarrolladas en los procesos corporativos, en cambio, se hace importante la correcta configuración de dichos dispositivos, esto por la gran diversidad de mecanismos, equipos y aplicaciones que se utilizan en la gestión de la seguridad, en donde una configuración incorrecta, el desconocimiento por parte de los administradores, la no implementación de las buenas prácticas al momento de configurar estos dispositivos y la poca atención y dedicación prestada en la implementación de los mismos puede generar diferentes vulnerabilidades y amenazas que descubiertos por los atacantes se pueden convertir en diferentes incidentes de seguridad [30]. Para minimizar el impacto de esta vulnerabilidad es recomendable: realizar una correcta estructuración de los procesos, diseñar topologías e infraestructuras sólidas,

mantener tanto las políticas y normas de seguridad como los sistemas de información, detección y remediación, actualizados, *Frameworks* y mejores prácticas documentadas y en constante ejecución, Monitorear internamente los sistemas de información realizando un *Hardening* que nos lleve a configurar los dispositivos, sistemas y herramientas de una forma más optima, entre otras [31].

La vulnerabilidad "Secuencia de comandos en sitios cruzados (XXS), por su parte, Consiste básicamente en la ejecución de datos ajenos en una página web sin realizar un filtro o saneamiento de datos. Normalmente este tipo de ataques utilizan un API basado en el lenguaje *Javascript* para modificar la estructura o el uso de una página web existente. Este tipo de prácticas permiten al atacante secuestrar sesiones, realizar un *Defacement* en donde se modifica un sitio web, redireccionar al usuario final o un sitio malicioso con un fin especifico que en la mayoría de los casos resulta siendo el robo de la información de inicio de sesión o un phishing en general [32]. Para prevenir este tipo de ataque se deben separar el contenido activo del navegador y los datos que no son de confianza, utilizando marcos de programación como Ruby y React JS, aumentando el uso de peticiones HTTPS, logrando así habilitar una política de seguridad de contenido como un plan de defensa [33].

Desde otra perspectiva, la Deserialización insegura, indica que las aplicaciones constantemente reciben información o instrucciones en diferentes leguajes de información la cual se encuentra serializada (codificada en serie de bytes o un lenguaje legible como XML o JSON), el sistema la Des-serializa, lee y/o interpreta con fin de usar dicha información. Los atacantes utilizan este proceso para introducir paquetes maliciosos modificables y borrables los cuales les permiten acciones tales como: elevar privilegios, ataques de repetición, inyección, ejecución de código de manera remota, ataques de persistencia. La definición correcta de estructura de datos y objetos, los mecanismos de defensa para los protocolos de comunicación, los servicios web y los *Brokers*, el uso de certificados digitales en el proceso de deserialización y el monitoreo del mismo proceso, van a permitir evitar que esta vulnerabilidad sea aprovecha o explotada, o el menos no de manera sencilla podrían llevarlo a cabo. Un ejemplo de explotación de esta vulnerabilidad es la CWE-502: *Deserialization of Untrusted Data* [34].

Otro elemento relevante son los "Componentes con vulnerabilidades conocidas", las cuales son aplicaciones, servicios, F*rameworks* y librerías que se implementan en un sistema, plataforma, infraestructura o proyecto sin ser analizados previamente, los cuales poseen vulnerabilidades conocidas por atacantes, lo que facilita la intrusión a estos componentes débiles para una posible afectación en el aplicativo, control de usuarios, acceso a bases de datos o la perdida de información

[35]. Con el fin de no abrir brechas nuevas en los sistemas implementados o los nuevos sistemas que se adaptaran en la compañía se recomienda analizar las vulnerabilidades y las versiones de los componentes a implementar y los que se están usando actualmente, realizando inventarios de componentes y versiones de los mismos, verificar la no utilización de los componentes y darles de baja o separarlos del sistema porque al no utilizarlos se exponen a permanecer desactualizados, incorporar mecanismos de seguridad enfocados en los componentes que permitan deshabilitar funcionalidades que generan vulnerabilidad o que no se utilizan [36].

Registro y monitoreo insuficientes, es una vulnerabilidad que indica que, lo que generalmente se está encontrando es que los ataques de seguridad son ejecutados dejando puertas traseras *(Backdors)* abiertas, con el fin de ejecutar en el tiempo diferentes ataques y cada vez permear más los sistemas, lo anterior se presenta debido que el monitoreo y evaluación de registros y *logs* generados en la aplicaciones y dispositivos de seguridad es insuficiente o no se hace de manera responsable, pasando por alto las advertencias que la misma infraestructura y aplicaciones generan, se debe tener en cuenta que estos registros son indispensables para identificar y resolver los ataques recibidos [37]. Si se implementa un monitoreo periódico tanto a los logs como a la red y dispositivos en general se pueden corregir diferentes vulnerabilidades que ni siquiera sepamos que existen, que están siendo o que fueron explotadas entre los ataques de seguridad recibidos. Realizar pruebas de *Pentesting* hacia las aplicaciones web nos ayudará a corregir y mitigar diferentes riesgos, tomando acciones al percibir diferentes intentos de inicio de sesión errados o intentos de inyección de datos del lado del servidor [38].

Las Fallas Criptográficas, están relacionadas con las debilidades que se presentan en el diseño de las aplicaciones web, afectando así la confidencialidad y la integridad de los datos [39]. Buscando disminuir esta vulnerabilidad adoptamos herramientas con sistemas de cifrado, utilizando formas de codificación digital, analizando previamente las bibliotecas de cifrado y los algoritmos de hash si se pretende llevar a cabo su implementación. Usando cifrado al momento de almacenar y transmitir los datos, protegiendo el código de ataques de fuerza bruta, construcción de algoritmos criptográficos propios, teniendo en cuenta que los atacantes saben permear algoritmos ya diseñados en el mercado [40].

Finalmente, Salteo de autorización a través de entradas de datos controladas por el usuario *(IDOR & Path Traversal)* es una vulnerabilidad tiene como objetivo acceder archivos o directorios que están almacenados fuera de la carpeta raíz web, dejando a los atacantes acceder a estos directorios y

visualizar los archivos que contienen, es conocido como ataque transversal de directorio, puede acceder a códigos fuentes, configuraciones de aplicación y directorios almacenados [41]. Este problema de autorización se presenta cuando un atacante puede leer los mensajes de todos los usuarios, cuando el principio es que cada usuario solo pueda leer sus propios mensajes. Usando referencias indirectas, autorización adecuada, que cada usuario solo pueda verificar su propio Id y no tener acceso a los mensajes de los demás usuarios para evitar que un atacante explote la vulnerabilidad descrita en los sistemas [42].

Ataques Servicios Web

Los ataques a los servicios web en el momento que se habla de ataques de seguridad y partiendo desde el concepto de los pilares de la seguridad informática, caracterizar los ataques según el pilar que se encuentran afectando, disponibilidad, integridad y confidencialidad. Se presenta tanto el ataque como la mitigación que puede realizarse, según la literatura.

Ataques a los servicios Web que afectan la disponibilidad

Análisis coercitivo: Es un ataque de denegación de servicio (DoS) el cual tiene como objetivo agotar la memoria del servidor que contiene el servicio web, inyectando innumerables líneas de código dentro de su configuración XML. Llevando la CPU a un 100 % de uso, mientras procesa el documento XML enviado por el atacante [43].

Para mitigar este ataque es necesario utilizar validaciones estrictas de estructura en los archivos de configuración, ejecución y mensajes de tipo XML. Adicionalmente, se puede limitar este ataque brindando acceso solo a los usuarios que van a consumir el servicio web, mediante un indicador de compromiso de tipo usuario, donde el sistema identifica si es un usuario desconocido y bloquea sus actividades [43].

Expansión de entidad XML*:* Este ataque consiste en definir una entidad bastante larga y repercute al ser llamada en diferentes ocasiones mediante la comunicación, la expansión de entidades con cadena vacía solicita un uso del 100% de CPU generando una denegación de servicio. Actualizar la versión de SOAP es un mecanismo de defensa efectivo teniendo en cuenta que en sus actualizaciones se encuentran exceptuadas las entidades externas, las cuales en su mayoría son las que permiten la explotación de la vulnerabilidad por el atacante [44].

Referencia de identidad XML: El objetivo de este ataque no es agotar los recursos del servidor o llevar a cabo ataques de tipo DoS, en este caso lo que el atacante quiere es exponer información sensible que haya sido digitada en el servidor o en la página HTML, dejando en evidencia, datos del cliente, datos de medios de pago luego de una transacción. En común con la expansión de identidad este ataque puede ser mitigado con la actualización de la versión de SOAP y limitando la referenciación solo a los usuarios que construyen esta aplicación o recurso [45].

Inundación XML: Mediante múltiples solicitudes permitidas el atacante busca denegar el servicio de los servicios web, puede ser de tipo único o distribuido, en donde el único es un cliente o atacante realizando múltiples peticiones permitidas y el distribuido es donde el atacante realiza las peticiones desde diferentes lugares controlados al mismo tiempo. La manera de mitigar o evitar este tipo de ataque es el uso de computación en la nube, también utilizar servidores con gran capacidad sin importar que no se requiera para la utilización del servicio, aunque esto conlleva a innecesarios gastos a la compañía puede evitar la indisponibilidad de los servicios [46].

Recuperación de claves DOS mediante firma XML: Existen mensajes SOAP en los que se necesita una clave publica para verificar la firma XML, en algunos casos se utilizan métodos para recuperar dicha firma toda vez que esta no se encuentra descrita en los archivos XML, haciendo autollamados al método<RetrievalMethod> el atacante puede generar un bucle lo cual repercute en una denegación del servicio. La manera de mitigar este riesgo es prohibiendo el uso de este método <RetrievalMethod> y validar si la recuperación de clave se encuentra en el mensaje SOAP [26].

Criptografía Recursiva: En los servicios web es posible adoptar métodos de seguridad como lo es la criptografía donde los mensajes SOAP pueden ser firmados con diferentes claves. Este ataque agota los recursos del sistema web atacado almacenando todas las claves y al mismo tiempo descifrando estas claves con algoritmos de clave pública, cifrando varias veces el mismo contenido y aumentando excesivamente el consumo de memoria y el alto uso de la CPU. Una manera de evitar este ataque es adoptando como un requisito que cualquier mensaje SOAP que no cumpla las políticas de seguridad WS sea descartado [27].

Redireccionamiento de Referencia: Los usuarios poseen gran desempeño en los datos firmados o cifrados una vez que en un mensaje SOAP es usado el procedimiento de cifrar o firmar, inclusive es viable usar los procedimientos fuera del mensaje original, ahora, es ahí donde el agresor visualiza una vulnerabilidad, pues una vez que son cifrados o firmados externamente el receptor debería bajar

el documento completo donde es dirigida la alusión antecedente de seguir con la solicitud, teniendo presente que la recuperación del mensaje completo puede tener un gran tamaño podría ser denegado el servicio. [28].

Ataque de Matriz SOAP: Dada la alta adaptabilidad de los mensajes SOAP se logran usar matrices, una mala configuración de esta clase de recursos en los mensajes puede llevar a una denegación de servicio, bien sea limitando el número de recursos máximos, o comparando el número de recursos máximos declarados con el número de recursos mayor que llega por medio del mensaje, descartando el mensaje en caso de no concordar logramos mitigar o restringir al agresor las formas de denegar el servicio**.** [29].

Denegación de Servicio SOAP: Teniendo presente que es la manera de que un comprador o un servidor se comunique con el servicio web, se estima uno de los ataques más frecuentes con fin de hacer la denegación de servicio, y más todavía al tener en cuenta que es protocolo de comunicación más usado e implementado por las compañías para la utilización y consumo de los servicios Web. Al influir este mensaje se perjudica la lógica de este y como resultante produce bugs que imposibilitan la comunicación, si bien el mensaje se reinicia una y otra vez al descubrir cortes en la comunicación, la mala configuración de dichos reinicios o la no integración de errores en la comunicación tienen la posibilidad de afectar en gigantes tiempos en la indisponibilidad del servicio. [30].

Spoofing de direccionamiento WS: Existen diferentes tipos de *Spoofing* de redireccionamiento WS, la suplantación de dirección consiste en configurar un mensaje SOAP tipo *Reply To*, en donde básicamente el usuario ingresará a un servicio web desconocido o malintencionado, donde le es recreado un escenario similar para darle confianza al usuario de realizar allí transacciones. La segunda es la suplantación de servicio o sistema destino la cual consiste en redireccionar el resultado de la transacción a una dirección o sistema diferente, el cual normalmente debería tener como destino el cliente o el sistema real. Por último, entre el servicio del cliente y el servicio servidor puede ser redirigida la transacción o la información con el fin de modificar alguna instrucción antes de llegar al siguiente destinatario o receptor. Para disminuir la efectividad de este ataque se puede verificar el objetivo final de la persona que invoca el servicio web, puede ser a través de un certificado digital que garantice que el cliente quien realiza las peticiones al sistema y que además el cliente si se encuentra en la dirección o campo transaccional correcto [31].

Ataques a los servicios Web que Afectan la Integridad

***Morphing* malicioso también conocido como la alteración de datos o manipulación del contenido:** Utilizando un esquema de hombre en el medio, el atacante busca interceptar normalmente los mensajes SOAP entre la comunicación de un cliente o servidor y un servicio web, modificando el cuerpo del mensaje normalmente o añadiendo otras direcciones al mensaje con el fin de capturar datos, para modificarlos en pro de nuevos ataques enviando el mensaje modificado al receptor. Haciendo el uso correcto de la criptografía, firmando los componentes más críticos de los mensajes SOAP, se puede mitigar este tipo de ataques [47].

Desvío de enrutamiento: También usando metodologías de tipo hombre en el medio el atacante busca agregar en el encabezado del mensaje información de enrutamiento con el fin que se tengan nuevos intermediarios en el transcurso del mensaje SOAP mientras es consumido el servicio Web con el fin de mal formar los mensajes para que el sistema incurra en una negación del servicio o para que se produzcan nuevos redireccionamientos. Omitir intermediarios desconocidos y firmar los datos del encabezado que contienen los intermediarios autorizados, al igual que incluir datos de autenticación en los intermediarios por los cuales pasa el mensaje puede contribuir en la disminución de ataques de enrutamiento [48].

***Spoofing de Metadata* o envenenamiento de esquema:** Anterior a la comunicación que se debería producir con un servicio web debería ser recuperados los fronteras o información de cómo debería ser invocado el servicio, formatos, direcciones, límites y requisitos de estabilidad son ciertos de los metadatos que tienen que ser conseguidos anterior a la comunicación, los cuales da el servidor. Hay 2 archivos que tienen dentro dichos datos, el documento WSDL y el documento de políticas de estabilidad. [49].

Spoofing WSDL (*Manipulación de parámetros *WSDL): El atacante busca cambiar los parámetros contenidos en el archivo WSDL, uno de los objetivos sería redirigir la comunicación a diferentes destinos o causar errores en la transacción mientras establece una transacción similar con los datos correctos, pero en beneficio propio. Se sugiere adoptar un enfoque de criptografía en la capa de seguridad del servicio Web, con el fin de proporcionar privacidad WSDL, mitigando así el efecto de este tipo de ataques [50].

***Spoofing* de política de seguridad WS:** En esta ocasión el atacante tiene como objetivo eliminar o modificar la información de seguridad contenida en el archivo de políticas de seguridad del servicio web, con el fin de disminuir el nivel de seguridad y hacer visibles los datos contenidos en la

comunicación del servicio. Lo cual es un gran riesgo puesto que se podrían identificar y exponer datos altamente sensibles. Con el propósito de mitigar este tipo de ataques se puede considerar los requisitos de acuerdo de nivel de servicio, los cuales se centran en generar políticas de seguridad más eficientes y seguras en cuanto a su integridad [51].

Reescritura de *XML:* En general los diseñadores de servicios web y programadores de estos, suelen utilizar diferentes técnicas de cifrado de los mensajes SOAP lo cual genera una alta seguridad en la comunicación en cambio, el objetivo de los atacantes es reescribir esta configuración irrumpiendo tanto en los algoritmos diseñados para las firmas como en los archivos de configuración XML, con el fin de tener un control desde el inicio hasta el fin o entrega del mensaje. Definiendo políticas estrictas de lectura del XML tanto del lado del cliente como del lado del servidor y en el trayecto del mensaje se hace más difícil que sea reescrito el XML, adicionalmente incluir la verificación del XML en el trayecto del mensaje reduce aún más la posibilidad de ataque [47].

Exclusión de firma XML: Normalmente el empaquetamiento del mensaje SOAP mediante el archivo de configuración XML es exitoso por la firma que se le brinda bien sea con un certificado o un algoritmo propio, lo cual es más común teniendo en cuenta que cuando se descubre una vulnerabilidad para una firma, esta puede ser explotada en diferentes sistemas, caso contrario a diseñar la seguridad del aplicativo en el código de programación puesto que esto hace única la configuración realizada. Este ataque busca que sea excluida la firma en el archivo XML, [52].

Ataques a los servicios Web que afectan la confidencialidad

Intrusión de mensajes o *Sniffing Message:* El atacante utiliza diferentes métodos y herramientas para llevar a cabo la lectura de los datos enviados entre un servidor y el consumidor de un servicio web, toda información que logre obtener que no está destinada a él es llamado intrusión de mensajes, para disminuir este tipo de ataques es recomendable cifrar las comunicaciones utilizando criptografía dado que entre los registros de un ataque de escucha del tráfico solo se verá que el mensaje fue enviado mas no tendría acceso a su contenido [53].

Divulgación de *WSDL:* Consiste en descubrir el archivo WSDL de un servicio web, dado que normalmente en este se encuentran configurados métodos, funciones, parámetros de entrada y salida y el resultado esperado u obtenido al consumir el servicio web. Luego de recuperado se puede intentar obtener acceso al servicio web, explotar sus datos o simplemente publicarlo en la red de internet con el fin de que alguien más lo explote según sus intereses. Este archivo con extensión

.wsdl contiene la información o meta data base para consumir un servicio web. La clave para que no sea encontrado o explotado es realizando una configuración responsable desde la programación del servicio web, utilizando código limpio y compacto que no contenga información de las funciones o parámetros usados en el archivo de configuración [50].

Escaneo y enumeración de *WSDL:* Es el estudio a profundidad que le hace un atacante a un archivo WDSL que haya obtenido o descubierto en un escaneo de una página web, este tipo de análisis puede llevar a que se conozca el usuario o página final que utilizara el servicio web, creando la puerta a un ataque de redireccionamiento, puede obtener los pasos por los que transcurre [50].

WSDL Google Hacking: El atacante utiliza los diferentes métodos de búsqueda de Google con el fin de encontrar el archivo WSDL de un servicio o página web en específico, este tipo de búsqueda utiliza el motor de bases de datos para tener la mayor cantidad de resultados posibles con un filtro avanzado de opciones, entre tantos puede encontrar archivos WSDL divulgados por otros atacantes. Usando métodos de encriptación de extensión se puede disminuir la efectividad de dichas búsquedas dado que por lo general uno de los parámetros es que el archivo tenga la extensión .WSDL, utilizando un algoritmo que lea el archivo sin importar la extensión en que se encuentre podría evitar diferentes lecturas de los atacantes [50].

Ataques adaptativos de texto cifrado: En general las comunicaciones de los servicios web independientemente de su leguaje, utilizan cifrado híbrido con el fin de proteger la confidencialidad de los datos y mensajes enviados. Dentro de este tipo de cifrado se utiliza inicialmente un cifrado asimétricos (algoritmo para la codificación de los datos) mediante el cual se cambian los datos para evitar su fácil lectura y posteriormente en el envío del mensaje utiliza los esquemas de cifrado público con el fin de solicitar una llave publica para recibir los mensajes y que estos puedan ser leídos por el receptor.

Teniendo en cuenta lo anterior el atacante centra su atención en descifrar la información, mediante técnicas avanzadas de descifrado en el momento que asimétricamente se cifran los datos, esto puede conllevar a que el atacante pueda modificar la información y/o instrucciones antes de enviar el mensaje, entregando al receptor información e instrucciones diferentes bien sea para que ejecuten otras acciones o que le sirven para generar errores en la comunicación, para distraer el personal de seguridad mientras aprovecha la información obtenida. Utilizando nuevas técnicas para el cifrado de datos y algoritmos con inteligencia artificial o que den alertas en caso de que intenten ser descifrados por entes externos a la comunicación normal de los mensajes. Se recomienda sean diseñados los

algoritmos puesto que los mercados se encuentran expuestos y se pueden presentar explotación de vulnerabilidades compartidas por los atacantes [54].

Ataques a los servicios Web que afectan el control de acceso

Ataque repetitivo: El objetivo del atacante será capturar mensajes SOAP de inicio de sesión, toda vez que puede lograrlo capturando el tráfico, usando el modelo de hombre en el medio o utilizando cifrados adaptativos. Esto con el fin de modificar estos mensajes para tener acceso a las funciones de los servicios Web o simplemente modificar e introducir mensajes los cuales el servidor tomara como confiables. Este tipo de ataques puede ser mitigado utilizando estampas de tiempo de seguridad en las sesiones que se levantan en la comunicación SOAP y en el transporte de los mensajes que se utilicen en el servicio Web [51].

Fuerza bruta: Mediante diferentes herramientas el atacante busca obtener las credenciales de ingreso frente a sistemas de autenticación y/o administración del servicio o de la página web donde se alojan o se consumen los servicios web, aplica para toda comunicación dada mediante mensajería o por los protocolos SSH, FTP, TELNET. [55].

Inyección XML: Consiste en inyectar en diferentes etiquetas del mensaje SOAP instrucciones para obtener un control de acceso administrativo no autorizado, haciendo que el receptor perciba como normal el mensaje original y permita su ejecución. [56].

Ataque cruzado para recuperación de clave *XSA:* Normalmente en la comunicación de mensajes SOAP de utiliza una firma para que el receptor del mensaje identifique la confiabilidad de dicha firma, hay diferentes formas de usar esta firma, quemada en el código fuente, referenciándola de manera interna y de manera externa. [57].

Inyección *Xpath:* Se asemeja a las inyecciones de tipo SQL con la diferencia que los XML no cuentan con el mecanismo de control de acceso que caracteriza las bases de datos de tipo SQL, el objetivo del atacante se basa en leer el documento XML del servicio o el mensaje mediante una consulta Xpath, en caso de tener éxito el atacante puede elevar los privilegios en el sitio web cuando se usa para autenticación. Obtención de usuario de bases de datos para iniciar sesiones con un fin determinado. Limitar la mayoría de los caracteres especiales posibles, verificar los usuarios relacionados con las consultas tipo Xpath recibidas, se debe considerar la generación de una lista de usuarios conocidos para este tipo de consultas [58].

1.1.3 Honeypot

Es un sistema que permite identificar, como los piratas informáticos emplean sus herramientas para tratar de entrar en un sistema y lograr aprovechas las vulnerabilidades existentes, con el propósito de ejecutar un ataque informático. Estos sistemas son clasificados por niveles de interacción definidos como; bajo, medio y alto. Cabe resaltar que, a mayor nivel de interacción se identifican con mayor facilidad y precisión los vectores de ataque [59].

En cuanto, a su funcionamiento, las *honeypot* se visualizan en la red donde se instalan, como un dispositivo vulnerable. Esta contiene una bitácora, que puede registrar los pasos realizados por los atacantes, que se interesan en perpetuar una violación de seguridad, sin que los atacantes puedan darse cuenta. La *honeypot* desvía la atención del atacante, de manera que no se comprometan los recursos principales, pretendiendo que el atacante tenga acceso, limitado o nulo al sistema. También, permite capturar nuevos virus para su posterior estudio, formar perfiles de atacantes y sus métodos de ataques favoritos [60].

Al obtener información de los vectores de ataque es posible implementar IOC para mejorar la seguridad e identificar remitentes de correo, trazabilidad de cuentas y máquinas no usuales en la red en los dispositivos de seguridad perimetral. Para seleccionar el tipo de *honeypot* se debe tener en cuenta, la ubicación en la red, si es virtual o física y tener claridad del sector objetivo que se desea

Para la presente investigación, una honeypot académica se entiende por una honeypot implementada dentro de un ambiente controlado, el cual puede estar expuesto, o no expuesto en la red local. Con el fin, de qué, los ataques externos no alteren los resultados esperados y se puedan analizar y enfatizar en los ataques que se realicen a la misma.

Soluciones Honeypot

En el mercado de *Honeypots*, existen pocas herramientas comerciales, aun cuando, el código libre ofrece muchas utilidades que sirven tanto a empresa como a particulares [62]. Para el desarrollo de la presente investigación, se exponen algunas soluciones de *Honeypots* para la tener en cuenta en la implementación.

Trap-X: Es una tecnología que otorga un alto nivel de detección y prevención de amenazas en tiempo real, instala servicios trampa, con algunos servicios reales de la organización, con el propósito de detectar intentos de intrusión, con fines que se encuentren lejanos al ambiente de producción y no

tener afectaciones. Una vez detectados tiene la función de automatizar la respuesta a un incidente del caso detectado. [61].

Allure: *S*e centra en verificar ataques internos con inteligencia artificial y estudio a los patrones y comportamientos de red y navegación de los empleados. Se enfoca en centralizar los comportamientos de los usuarios y combinarlo con las estadísticas de red, lo cual permite anteponerse a una situación de ataque [62].

StrongARM: Es un sistema, que bajo ambientes de señuelos tiene como objetivo, identificar las infraestructuras utilizadas por atacantes, con el fin de realizar una lista negra de dispositivos, lo cual logra mediante el estudio de fuentes de código abierto y apoyados en empresas aliadas tales como ThreatConnect. Adicionalmente tiene autonomía para descifrar código de ramsomware y destruirlo o sacarlo del sistema para evitar afectaciones [63].

LogRhyhm: Es una herramienta, la cual evidencia que tan comprometidos fueron los sistemas de información y los archivos después de un ataque. Se basa en análisis forense de amenazas, para evidenciar los vectores relevantes de los ataques y lograr posteriormente la aplicación de reglas y políticas. Realiza un ambiente que permite la búsqueda elástica, teniendo como resultado los patrones de ataque y el reconocimiento de los orígenes del ataque para tomar acciones legales según el caso [64].

Dionaea: Es un *honeypot* que se caracteriza por los múltiples protocolos de red soportados entre ellos FTP, HTTP, MYSQL, SIP, SMB. Se especializa en el protocolo SMB, dado que, es el que mejor cobertura presenta, además se utiliza para identificar parámetros básicos de los atacantes tales como IP, URL, conexiones e IP por país, tipos de protocolo utilizados [65].

Kippo: Se describe como un *honeypot* de interacción mediana, su enfoque principal es el servicio SSH. Está escrito en Python y alojado en GitHub con licencia libre, esta herramienta permite evidenciar los ataques de tipo fuerza bruta que se presenten, tiene la capacidad de capturar los comandos realizados por el atacante, en su mayoría los no cifrados [66].

DejaVu: Marco o framework de trampas de código abierto, permite crear y ejecutar señuelos en una infraestructura definida. Este ambiente virtualizado con métodos de contenerización soporta servidores de tipo HTTP, SQL, SMB, FTP, SSH, entre otros, es utilizado para, el estudio e investigación de los ataques, que a diario se llevan a cabo en la red y propone generar un ambiente

atractivo de ataque, para verificar la forma en que los ciberdelicunetes buscan penetrar la infraestructura, teniendo como resultados, palabras claves, listas blancas y negras de IP, tipos de consulta SQL y factores que permitan definir nuevas políticas y configuraciones que fortalezcan los ambientes productivos, apoyado con estampas de tiempo que facilitan la interpretación [67].

HoneyDrive: Se trata de la agrupación de diferentes *honeypots* preconfigurada tales como *Kippo, Kojoney, Anum, Dionaea, Worpot, Glastopf, Conpot,* entre otras. Permite simular sistemas vulnerables, con el fin de atraer atacantes y posteriormente, identificar que patrones de ataques utilizan, integrando diferentes herramientas y permitiendo la instalación de otras herramientas Linux tales como *snort* para dar apoyo a la captura de datos que sean procesados [68].

Modern Honey Network (MHN): Consiste en un ambiente servidor donde se pueden ejecutar diferentes honeypots, tales como Kippo, Dionaea, Cowrie, HoneyMap, entre otras. El cual permite visualizar de manera gráfica y dinámica las actividades o ataques que les surjan a las diferentes soluciones, permitiendo tener sensores tales como Snort para la rápida y efectiva recopilación de datos. Normalmente se maneja en un ambiente de contenedores, aunque puede ser también configurada como máquina virtual desde Ubuntu la distribución conocida de Linux. Esta desarrollada en código abierto y factible, para la implementación en ambientes empresariales, donde la seguridad de la información es un factor de alto riesgo, a pesar de ser de media interacción se considera efectiva y competente ante algunas soluciones que requieren pago. Una administración adecuada de la herramienta puede fortalecer en gran manera un sock de seguridad o un área específica, para entender las maneras en que se reciben ataques, para tomar acciones que logren contener y evitar dichos ataques [69].

Honeypot de Alta Interacción: Es la simulación de un entorno productivo, donde el atacante supone que tiene acceso a la información, servicios o activos importantes para la organización. Con base en lo expuesto, la información relacionada con las herramientas y técnicas que utilizan los atacantes permite identificar controles técnicos adecuados para proteger las aplicaciones y los datos [70].

1.1.4 Controles Técnicos

Son herramientas tecnológicas o metodologías que se utilizan para evitar que una vulnerabilidad sea explotada, que se presente fuga de información y/o exista una denegación de servicio efectiva. Su finalidad es prevenir ataques de seguridad, teniendo en cuenta los monitoreos constantes que realizan

sobre los dispositivos que se quieren salvaguardar. Estos controles normalmente son identificados en la gestión de incidentes, entre los controles técnicos más utilizados están: Firewall, IDS, IPS, DLP, Antivirus, políticas de seguridad, SIEM, OSSEC, sandbox y EDR [6].

1.2 Estado del Arte

Para la elaboración del estado del arte, se realizó búsqueda de la literatura haciendo uso de bases de datos como: Scopus, IEEE, Scielo, Science.gov y *Google Scholar* en relación con los IOC que pueden ser entregados para proteger servicios web, además se tomaron aquellas investigaciones que trabajaron o implementaron una *honeypot* para generar un modelo de seguridad y reducir niveles de riesgo.

Lance Spitzner, [71] construyó a comienzos del año 2000 una red de seis ordenadores en su lugar de residencia, dicha red la diseño para estudiar el comportamiento y formas de actuación de los atacantes. Fue de los primeros investigadores en adoptar la idea, y hoy es uno de los mayores expertos en *honeypots*, pionero del proyecto *honeynet* (www.honeynet.org). El objetivo de este proyecto es analizar todos los intentos de inicio de sesión, capturando las credenciales utilizadas, la IP y Puerto de donde se intentó realizar la conexión. Adicionalmente, en los casos de conexión exitosa se registraban las rutas y los documentos a los que intentaron y tuvieron acceso.

Para la implementación y fundamentación de la Honeypot Académico en ambiente controlado de la actual investigación, se usa lo establecido por Spitzner, quien analizaba el tráfico sospecho sobre la Red LAN, identificando los vectores de ataque generados por actores internos, para generar los controles pertinentes; en cuanto a la presente investigación, se establece una diferencia, en el la exposición de la *honeypot*, dado que esta, recibió ataques internos y externos.

Según la metodología para la identificación de indicadores de compromiso para la protección de infraestructuras críticas. [6], se realizaron un rastreo de la información junto con la comprensión de las necesidades en el marco de la ciberseguridad de la NIST, implementando un inventario completo y desarrollando una clasificación de activos críticos, a través, de dispositivos avanzados de seguridad perimetral, además, de los procesos definidos de gestión del riesgo de incidentes, lo que permite lograr un modelo de seguridad más eficiente y concreto para la protección de las infraestructuras críticas de las organizaciones.

Según lo planteados por los autores [6], los ataques están basados para infraestructuras críticas, en comparación con la presente investigación, basada en los ataques a los servicios web; asimismo, la metodología para la identificación de indicadores de compromiso para la protección de infraestructuras críticas es una implementación metodológica, mientras el desarrollo de esta investigación tiene una fundamentación técnica. Analizando la protección de las infraestructuras críticas, se puede considerar que se desarrolla una metodología para la identificación de la IOC, no bajo el enfoque técnico que se proponen los fabricantes de tecnologías de seguridad, enfocado a la protección de los procesos de negocios que soportan estas infraestructuras, detectando los diferentes incidentes de seguridad que se presentan sobre los activos críticos de información.

Con base en, el estudio Configuración de *honeypots* adaptativos para análisis de malware, los resultados de los IOC se logran definir utilizando la metodología para análisis de riesgo, que permite identificar los usuarios que interactúan con el proceso, como la plataforma tecnológica asociada a los procesos críticos y la infraestructura de seguridad que protege los procesos [72].

En la tercera jornada Nacional de Investigación en Ciberseguridad plantean que el conocimiento obtenido es empleado para configurar *honeypots* de manera dinámica, permitiendo satisfacer los requisitos necesarios para que el programa maligno pueda desplegar toda su operativa. Los *honeypots* descritos en este proyecto son especializados en determinados protocolos y servicios orientados solo a interacción con código maligno, mediante el cual se detectan patrones de búsqueda, ejecución explicita del código con el que fue diseñado el software malicioso, reacciones sobre el sistema y los vectores de infección de estos. Cabe resaltar que, estos *Honeypots* analizan el comportamiento, los patrones, los puertos y los directorios que afectan el malware, dentro de una red, mientras que la investigación en desarrollo no se orienta hacia el malware, sino que se enfoca en los servicios web. La configuración de *Honeypots* adaptativos para análisis de malware [72] analiza el comportamiento de los patrones de una infección de un equipo por malware y la presente investigación, analiza los vectores de ataques a servicios web.

Por otra parte, brinda una estructura sobre la configuración de servicios trampa, que sirven de referente para la puesta en marcha del objeto de este trabajo de configurar las *Honeypots* adaptativas para el análisis del malware [72].

Quijije, D, realizo un diseño del prototipo de una honeypot virtual, la cual permite mejorar el esquema de seguridad en las redes de la Facultad de Ingeniería en Sistemas Computacionales y Networking de la Universidad de Guayaquil, la honeypot propuesta, estudia el uso, las características y ventajas al momento de establecer una conexión con internet, este ejercicio, permite tener

conocimiento de cómo operan los intrusos, aportando a la mejora de los esquemas de seguridad de la facultad. Las Honeypot son utilizadas como una herramienta de investigación con el propósito de "conocer al enemigo", teniendo en cuenta que al identificarlo y definir las técnicas que usa, se toman medidas que permitan mitigar las vulnerabilidades existentes en cualquier entorno de red.

Según lo planteado por el autor [73], las honeypots son herramientas que atraen al atacante, para aprender de él, rompiendo con los paradigmas establecidos para la seguridad web. El diseño del Prototipo de una Honeypot Virtual, para mejorar el esquema de seguridad en las redes, brinda un análisis sobre los IDS en host y red, detallando la interacción con los componentes y la aplicación de la informática forense sobre estos, presentando la infraestructura, componentes y configuración, de los diversos elementos que hacen parte de esta propuesta. Lo descrito fue logrado y verificado gracias a pruebas de intrusión realizadas que garantizan la operatividad de este prototipo y sus ventajas al momento de usar la información recolectada para proteger el sistema. En los resultados de este diseño se detallan las IP, los puertos y protocolos que más detecto la Honeypot, como tráfico sospechoso e intentos de acceso fallido en los incidentes de seguridad detectados, dichos resultados fundamentan el propósito de esta investigación.

Las principales diferencias entre el trabajo realizado [73] y el presente trabajo se centra en los ataques que infectan los servidores con malware, estableciendo las herramientas, las IP, los protocolos y los puertos utilizados para el ataque, en cambio la investigación desarrollada, se centra, en cómo se presentan, los ataques para los servicios web; cabe resaltar, que el tiempo de pruebas fue más extenso en este proyecto, con una duración de dos meses, en cambio, el estudio de Quijije tuvo un tiempo de prueba de quince días. Otra diferencia abordada entre las investigaciones en mención, es el licenciamiento puesto que, para el estudio realizado en la Facultad de Sistemas de la Universidad de Guayaquil se utilizó un software de prueba, limitado por un máximo de treinta días para su funcionamiento, en comparación a los Indicadores de compromisos basados en ataques a servicios web, se utilizaron herramientas de tipo Open Source, las cuales permite una reducción de costos para el proceso de análisis.

En el artículo "Detección y mitigación de vulnerabilidades día cero", expone como, la curva de protección en el tiempo no se ajusta a la curva exponencial que presentan las amenazas. Para llegar a esta conclusión los autores Guisao y Toro [74], exponen un constante análisis de vulnerabilidades a los diferentes servicios WEB, expuestos en una organización, utilizando una metodología para la detección de vulnerabilidades de redes de datos, la cual se desarrolla en tres fases: 1) Reconocimiento, 2) Numeración de Servicios, 3) Escaneo de Vulnerabilidad. Estas fases permiten

presentar la necesidad de implementar señuelos, que logren captar la atención de los atacantes y que permita proteger los mecanismos de seguridad e infraestructura, mitigando los riesgos de todo ataque evidenciado en los señuelos, logrando que no se materialicen los incidentes detectados a tiempo.

Si bien es cierto, las vulnerabilidades de día cero, tienen un previo estudio por parte de los atacantes, también es cierto que, un correcto monitoreo y una contención oportuna puede lograr proteger los sistemas. Sin embargo, un sistema no logra un cien por ciento de seguridad, en cuanto a los cambios y transformaciones de los atacantes. Se identifica que una detección temprana de vulnerabilidades de día cero, establece tiempos de respuesta rápidas, con acciones que favorecen a otros posibles afectados, evitando el daño de servicios y mitigando el impacto.

Con base en lo expuesto por Guisao y Toro, el propósito del artículo, era detectar y mitigar vulnerabilidades de día cero para servidores de aplicaciones mediante el uso de *Honeypots*, en cambio, para la investigación desarrollada se utilizó la H*oneypots* para identificar indicadores de compromiso en ataques a servicios web.

Trujillano [75], expone los Sistemas Adaptativos de Prevención de Intrusos Mediante *Honeypots*, donde se utiliza la información que recopilan las *Honeypots*, con el fin de, mejorar la seguridad de un sistema informático, por medio de reglas automáticas, además, de un método de investigación y estudio para examinar su efectividad. En cuanto se verificaron los sistemas reales de la Universidad Autónoma de Madrid, se logró identificar que, los ataques que se podrían recibir eran SSH, e inyección de SQL. Con este estudio, se logra comprobar que, para detectar este tipo de ataques con facilidad, las *Honeypots* más adecuadas eran *Cowrie* y *Glastopf* y fueron implementadas a través de una base de datos que guardaba los Logs. Posteriormente, transformaban la información de la base de datos con un archivo de *Python*¸ el cual, clasificaba los ataques como: temporales o definitivos. También se utilizaron los datos de los ataques para generar reglas, las cuales pueden ser incorporadas en los *IPTABLES* de los sistemas Linux, de tal modo, que al llegar las mismas peticiones sean rechazadas por los Sistema Perimetrales.

El resultado de la investigación de Trujillano [75], es la clasificación de los países con mayor número de peticiones, IP y la consolidación de las reglas de tipo permanente y las reglas temporales y definitivas exportadas por el archivo de *Python*. Los Sistemas Adaptativos de Prevención de Intrusos Mediante *Honeypots*, corroboran que las *Honeypots*, brindan información para establecer controles técnicos, y se establece como diferencia que la investigación en mención interpreta la información de manera automática, in tener una fuente de verificación confiable y la presente investigación

interpreta la información, por medio de informes generados por la *Honeypot* y la supervisión de un profesional de Seguridad Informática, asegurando que los indicadores de compromiso no sean falsos positivos.

Moral, C, establece la implementación de una herramienta honeypot (RASPOT) en una plataforma portátil Raspberry para el apoyo a un análisis forense [76], esta herramienta facilita analizar cualquier tipo de red, apoyando en este caso a redes soportadas en IOT (Internet of Things), el autor la probó en dos redes, en la red de la universidad autónoma de Madrid, donde se pasa desapercibida la honeypot al momento de realizar los escaneos dado que su ubicación en la red DMZ permitió un camuflaje de esta, para los atacantes no era fácil ingresar, sin embargo se detectaron un número considerable de registros, aun sabiendo que se tenía el firewall. Fue el caso contrario al implementarlo en una red doméstica, dado que, directamente se podía percibir como los ataques eran atraídos a un mismo dispositivo y se tuvo un número más elevado de registros de ataques.

De estos casos el autor interpreta que la metodología y herramienta utilizada puede ser estudiada más afondo y a una mayor escala con el fin de darle lugar en las empresas y corporaciones, las cuales requieran un producto que no solo escanee sus vulnerabilidades de red, sino que a su vez permita estudiar la forma en que los atacantes quieren realizar en la red sus intrusiones. Como trabajo futuro precisamente se encuentra potencializar el producto y ponerlo en funcionamiento en redes de mayor número de equipos, incluyendo las conexiones WLAN, dado que para este caso solo se tuvo en cuenta conexión LAN. Adicionalmente, exponer un servicio WEB, que cumpla la funcionalidad de la honeypot con el fin de que su uso sea multiplataforma y tenga una mayor conectividad, mitigando su limitación de instalación física [76].

En el análisis de la implantación de un Honeypot, en una plataforma portátil para informática forense (RASPOT), el autor tomo la decisión de renunciar a las herramientas comerciales de Honeypot y desarrollar una propia, donde se les facilitaba para analizar todas las opciones y ver las limitaciones de las cuales carecían, basado en el bajo nivel de la integración de las Honeypot, e incluir las distribuciones que deseaban por su facilidad y comodidad para extraer los datos. La investigación de Moral [76], sustenta que, las herramientas comerciales no brindan una amplia cobertura y están diseñadas para un tema específico, lo cual, permite implementar una infraestructura que emula el comportamiento de una Honeypot, facilitando capturar la información del tema objetivo como lo fue el caso de la presente investigación.

El estudio de las HoneyPots Web como Herramientas de Análisis de Ciberataques sobre una Red de Telefonía Móvil [77] abordan el sector de las telecomunicaciones y las infraestructuras críticas, para realizar la investigación y compresión de los ataques que llegan a las redes de telefonía móvil. Hicieron una evaluación del comportamiento de una infraestructura crítica con un monitoreo controlado y en un ambiente reducido, frente a diferentes casos de uso. Para lo anterior implementaron un *Honeypot*, con los cuales determinan varios vectores de ataques. Utilizaron *Snare*, una herramienta que duplica un sitio web y permite adaptarlo como un sensor de honeypot. *Snare* grafica los datos y resultados de los análisis de eventos que ocurren en las redes móviles, mediante un software llamado *Tanner*. Con este estudio los autores lograron que se registraran ataques de tipo XSS, SQL *Injection*, escaneo de puertos e inclusión de archivos locales. Este estudio amplía los sectores y la aplicabilidad de los sistemas *Honeypot*, mostrando que estas brindan posibilidades de identificar controles técnicos para el desvío de los ataques, detectar y conocer nuevas vulnerabilidades, sirven también para obtener información del atacante y conocer nuevos programas malignos de los cuales no se tengan conocimiento. Con relación a la presente investigación se puede apreciar que en ambas se utilizan herramientas de tipo *Open Source*, y en la implementación se diferencian en que para leer los vectores de ataquen fueron necesarias más de tres herramientas independientes, cuando *Wazuh* puede con toda la interpretación de logs y monitoreo de eventos.

De manera interesante e innovadora los autores [78] logran realizar un estudio en el cual evidencian como los atacantes utilizan los motores de búsqueda para encontrar sistemas vulnerables e información atractiva, como contraseñas, archivos ocultos, u otro tipo de información sensible en Internet. En este trabajo, estudiaron este tipo de ataques desde una perspectiva diferente, teniendo como resultado una *Honeypot* como servidor web, llamada *Dorkpot*, con capacidad para detectar solicitudes relacionadas con *Google Dork* y, de este modo, analizar la naturaleza de tales ataques de tipo reconocimiento. El resultado de este estudio es el top diez de los *Google Dorks* más utilizados para consultar información en las páginas, tal como, contraseñas, archivos de acceso, bases de datos de *WhatsApp*, entre otros. Lo cual invita a las personas administradoras de las páginas *Web* a tener en cuenta las actividades de reconocimiento ejecutadas por los atacantes para obtener archivos expuestos o no ocultos en los servidores y servicios de cámaras *Web* no restringidos. Se incluye esta investigación para ejemplificar un ámbito más donde las honeypot tienen un buen desempeño y que cada vez son más utilizadas por los investigadores para determinar los modos de ataque empleados para hacer daño a un sistema.

Las investigaciones anteriormente mencionadas tienen hallazgos de gran relevancia para las metodologías, modelos e integraciones que permiten conocer diferentes patrones de ataques, fallas,

amenazas y vulnerabilidades asociados a incidentes que atentan contra la confidencialidad, integridad y disponibilidad de la información. Estos hallazgos en su mayoría contemplan vulnerabilidades de día cero que apoyan disminuyendo los tiempos de respuesta de los incidentes y tienen una alta usabilidad de las *honeypot* e IOC como base para el desarrollo de los proyectos, pese a esto, se evidencia el bajo índice de implementación de los controles hallados en las herramientas de seguridad perimetral como firewalls, IDS, IPS, entre otros, argumentando la implementación como trabajo futuro. Conforme con las problemáticas mencionadas, se plantea: teniendo en cuenta el alto índice de ataques informáticos que se llevan a cabo diariamente en la actualidad ¿De qué manera se pueden utilizar *Honeypots* para generar recomendaciones basados en IOC que permitan fortalecer los procesos de seguridad en el diseño y la infraestructura de los servicios web?

2. Metodología y Resultados

La metodología utilizada para lograr definir indicadores de compromiso que, basado en la recolección de datos de una *honeypot* académica, identifiquen ataques de seguridad en servicios Web, generando recomendaciones para controlar y reducir los niveles de riesgo, fue divida en 4 fases como se presenta en la figura 1, fases que describen además el logro de los cuatro (4) objetivos. Fases en las que se describe el procedimiento realizado y se exponen de manera inmediata los resultados hallados en cada una de ellas. En la fase de identificación se realiza la búsqueda de las vulnerabilidades más relevantes aprovechando la información suministrada por fuentes como MITRE, Incibe-Cert, CCN-CERT y OWASP, y en base a su análisis se concluyen las 3 vulnerabilidades más relevantes. En la fase de definición de los IOC se realiza una búsqueda de IOC conocidos, planteando una línea base de la investigación, luego en la fase 3 fue implementada la *honeypot* describiendo los aspectos a tener en cuenta y por último en la fase de evaluación se realizan diferentes ataques de seguridad a la *honeypot*, se identifican los patrones de ataques analizando la información arrojada por la *honeypot*, se configuran los IOC teniendo en cuenta los vectores de ataque, una vez configurados se repite el ataque inicial y se analiza el resultado del laboratorio técnico en base al objetivo general esperado.

Tabla 2.1 Marco Lógico

Título: Indicadores de Compromisos Basados en Ataques a Servicios Web
Problema: Los ataques informáticos crean daños importantes, aún en las empresas que más rigurosidad presentan en su esquema de seguridad, estos ataques mantienen una búsqueda incesante por violar toda estructura organizacional, a través de la tecnología, con diferentes fines que siempre concluyen en el perjuicio de la víctima.
Hipótesis: Si se definen indicadores de compromiso en servicios Web con base en la recolección de datos de una H*oneypot* académica, se comprenden los métodos de ataque más utilizados por los delincuentes informáticos, además se puede ofrecer como resultado las recomendaciones para controlar y reducir los niveles de exposición.
Objetivo General: Definir indicadores de compromiso que, basado en la recolección de datos de una *Honeypot* académica, identifiquen ataques de seguridad en servicios Web, generando recomendaciones para controlar y reducir los niveles de riesgo.

Específicos	Fases	Actividades (metodología)	Entregables (resultados)
1. Identificar las diferentes amenazas a servicios Web y las tres vulnerabilidades más relevantes mediante la caracterización de los diferentes ataques informáticos.	Identificación de las vulnerabilidades de los servicios web	-Búsqueda de entidades enfocadas en el estudio de amenazas y vulnerabilidades existente en cuanto a la seguridad informática. -Clasificación de las amenazas y vulnerabilidades asociadas a los Servicios Web. -Rastreo de los informes periódicos. -Se identifican las tres vulnerabilidades más relevantes. -Se determinan los tipos de ataques que sirven para explotar las tres vulnerabilidades. -Compendio de los tres informes expuestos por MITRE, y de hace un promedio de las tres ponderaciones.	Se obtienen las diferentes amenazas hacia sitios Web, así como las tres (3) vulnerabilidades más relevantes que son: • Asociado a Denegación de Servicios • Asociado a Inyección de código • Asociado a Exposición de datos sensibles
2. Identificar diferentes indicadores de compromisos existentes que permitan, a partir de éstos, el desarrollo o ajustes de un set de indicadores para los servicios Web.	Identificación de indicadores de compromiso existentes	- Se identifican las diferentes empresas del sector público y privado dedicadas a entregar IOC de manera masiva y sin ánimo de lucro. - se realizó un estudio documental de cada proyecto (Alientvault, virus total, MISC Project.) por medio del cual, se pudo identificar su funcionamiento y la forma en	Como resultado final de esta fase 2, se generan los siguientes 5 IOC nombrados así: •Alien Vault IOC.ioc •Alien Vault IOC IP.ioc •Alien Vault IOC URL.ioc •Alien Vault IOC DNS.ioc •Alien Vault IOC HOSTNAME.

		la cual se distribuye la información. -Se extraen los IOC de Alientvault, consumiendo su servicio web con la herramienta Insomnia - Se usa la herramienta MANDIANT y el estándar OpenIOC para escribir los IOC en un formato global.	
3. Implementar un *honeypot* académico en ambiente controlado que permita la captura de los diferentes ataques informáticos y actividades no autorizadas.	Implementación de la H*oneypot* académica	-Búsqueda descriptiva de las diferentes *Honeypo*t comerciales. -Clasificación de la aplicabilidad de las *Honeypo*t comerciales con los Servicios Web. - Las soluciones fueron clasificadas mediante características que posibilitan la recolección de los vectores de ataque a servicios web. -Se selecciona la *Honeypot* que más cumplen con las características y se implementa en un ambiente controlado.	-Se realiza la implementación de H*oneypot* académico en ambiente controlado para la captura de los diferentes ataques informáticos. -Elaboración del Manual de Instalación de *Honeypot*, el cual permite utilizar la topología propuesta a los investigadores y/o implementadores de futuros trabajos.
4. Evaluar los indicadores de compromiso, ejecutando los diferentes ataques informáticos, validando la información	Evaluación de los IOC seleccionados a través de la información de la *Honeypot*	-Se expone la *Honeypot* por un periodo de dos meses, utilizando una DMZ. - Se analizan los ataques recibidos y se configuran un set de IOC, con base en, la recolección de datos de la *Honeypot.*	Se generaron una serie de recomendaciones (controles) que pueden ser aplicados en los servicios Web, con el fin de reducir los diferentes niveles de exposición.

recolectada por la *Honeypot* con respecto a los IOC definidos, generando recomendaciones de controles para la reducción de los riesgos		-Validación de resultados: exposición de los IoC en la *Honeypot.* -Recomendaciones de controles para la reducción de riesgos	

Figura 2-1 Metodología Propuesta

2.1 Fase 1. Identificación de las Vulnerabilidades de los Servicios Web

2.1.1 Metodología

Para identificar las vulnerabilidades de los servicios web, se utilizó el método inductivo, puesto que, se basa en el razonamiento y estudio de procesos específicos con el propósito de identificar y concluir las vulnerabilidades a través de fundamentos teóricos [79]. Cabe resaltar que, el desarrollo de esta fase, permite determinar los organismos de seguridad informática, para determinar las amenazas y vulnerabilidades más relevantes de los servicios web.

Se realiza una indagación de fuentes segundarias, enfocado en reconocer las entidades dedicadas a los estudios de amenazas y vulnerabilidades existentes, que adicionalmente, publiquen informes, de manera periódica; se establecen cuatro entidades de seguridad, OWASP, CCN-CERT, Incibe-Cert y MITRE; después, se clasifica la información, basado en las amenazas y vulnerabilidades asociadas a los Servicios Web. La delimitación de estas entidades de seguridad, permiten establecer las vulnerabilidades que cada entidad tiene en cuenta, en consecuencia, se identificaron las tres vulnerabilidades más relevantes para la aplicabilidad según los entes de seguridad (Ver Tabla 2.3).

De acuerdo con lo establecido el marco teórico, se desarrolló una relación entre los tipos de ataques a los servicios web y las tres vulnerabilidades elegidas, lo que permitió relacionar el tipo de ataque, mediante el cual, se da la posibilidad de explotar dichas vulnerabilidades (Ver Tabla 2.4). Seguidamente, los resultados expuestos por los cuatro entidades de seguridad encontrados, se validaron con respeto a los resultados establecidos por la organización MITRE, para llevar a cabo la identificación de las amenazas a los servicios Web y así confirmar las tres vulnerabilidades determinadas. Por último, se realizó una síntesis de los tres informes expuestos por MITRE y se promedian las puntuaciones otorgadas (Ver Tabla 2.8).

2.1.2 Resultados

Como se indicó en la metodología mediante la investigación realizada se encuentra que existen diferentes entidades de orden público, privado y gubernamental enfocadas en la identificación de las diferentes amenazas, agentes de amenazas y vulnerabilidades existentes. Algunas de estas entidades son CCN-CERT, OWASP, NIST, INCIBE-CERT, ESET, CSIRT (Equipo de Respuesta ante

Emergencias Informáticas) y Sophos Lab. En esta fase se identificaron las amenazas, agentes de amenazas, vulnerabilidades y los diferentes ataques a los servicios web, considerando las entidades que publican periódicamente información de seguridad relacionada con los servicios Web, consultando así las siguientes fuentes: MITRE (a través del portal CWE), CCN-CERT, OWASP, Incibe-Cert.

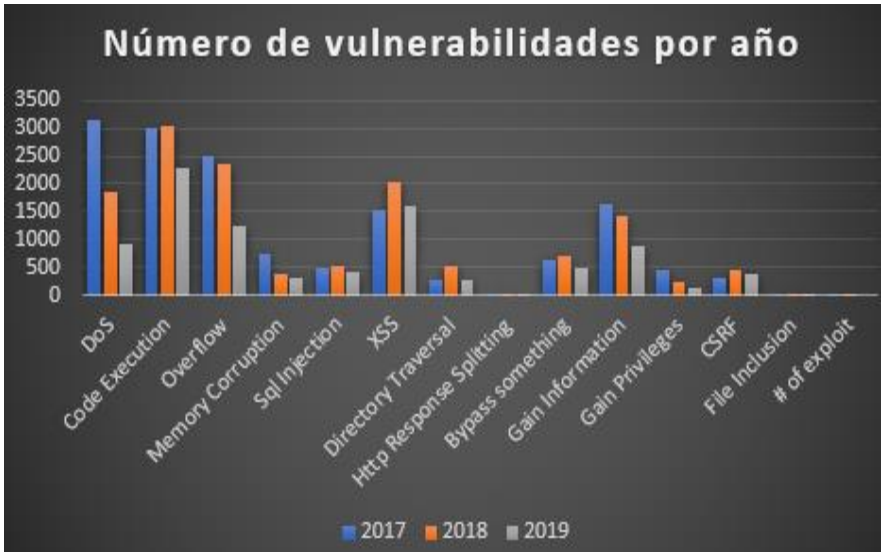

Figura 2-2 Número de Vulnerabilidad por año [80].

En la Figura 2.2 se presenta la información consolidada de manera histórica desde la página oficial de MITRE, donde se tiene en cuenta el comportamiento de las vulnerabilidades en la actualidad; tomando los registros de los tres últimos años.

Como se puede observar en la Figura 2.2, en el año 2017 las vulnerabilidades DoS, Code Execution, Overflow, XSS, y Gain Information, se presentaron en un mayor nivel. En el año 2018, estas vulnerabilidades tuvieron un crecimiento significativo y donde se presentó menor número de vulnerabilidades por año fue en el 2019, este decremento es positivo ya que probablemente se está ejerciendo la protección de vulnerabilidades.

Los organismos de seguridad informática CCN-CERT, INCIBE y OWASP, utilizan como fundamento, las bases de datos de vulnerabilidades CVE y NVD y se exponen los informes anuales,

con base en los incidentes de seguridad reportados que reúnen la información de interés a nivel de seguridad.

Por su parte CCN-CERT publicó entre los años 2017 y 2020 el informe de ciber amenazas y tendencias donde se puede identificar el ranking de amenazas y cómo fue la variación entre este rango de tiempo. Mostrando un top de 15 vulnerabilidades, que se presenta en la Figura 2.3.

Top Threats 2018	Assessed Trends 2018	Change in ranking
1. Malware	→	→
2. Web Based Attacks	↑	→
3. Web Application Attacks	→	→
4. Phishing	↑	→
5. Denial of Service	↑	↑
6. Spam	→	↓
7. Botnets	↑	↑
8. Data Breaches	↑	↑
9. Insider Threat	↓	→
10. Physical manipulation/ damage/ theft/loss	→	→
11. Information Leakage	↑	↑
12. Identity Theft	↑	→
13. Cryptojacking	↑	NEW
14. Ransomware	↓	↓
15. Cyber Espionage	↓	→

Figura 2-3 - Top 15 de amenazas CCN-CERT **[80]**

En la Figura 2.3 se aprecia como en los puestos del 2 al 4 se ubican amenazas relacionadas con las aplicaciones y ataques a servicios web, donde pueden ser aprovechadas las técnicas de ingeniería social y usar el phishing para cometer delitos informáticos.

De los informes expuestos por CCN-CERT entre el 2017 y 2020 [81], [82], [83], [84] también, se logró extraer los diferentes agentes de amenazas y para presentarlos se realiza la tabla 1. Estos agentes de amenaza concuerdan con los agentes de amenaza mencionados los diferentes informes de las demás entidades de ciberseguridad por ello se hace relevante realizar su consolidación.

Tabla 2.2 - Agentes de amenaza según CCN-CERT

Año / Agentes de Amenazas	2017	2018	2019	2020
Estados	X	X	X	X
Organizaciones Criminales	X	X	X	X
Ciberactivismo	x	X	X	X
Ciberterrorismo	X	X	X	X
Script Kiddies	X	X	X	
Actores internos	X	X	X	X
Ciberinvestigadores	X	X		
Organizaciones privadas	X	X		

Por otra parte, Owasp se centra en los riesgos y vulnerabilidades asociadas a servicios Web y genera un ranking o top de 10 vulnerabilidades, donde las califica teniendo en cuenta los riesgos, agentes de amenaza, vectores de ataque, debilidades de seguridad e impacto evaluando cada vulnerabilidad según su explotabilidad, prevalencia y detectabilidad obteniendo así una puntuación para cada una, presentada en la Figura 2.4.

Riesgo	Agentes de Amenaza	Vectores de Ataque	Debilidades de Seguridad		Impacto		Puntuación
		Explotabilidad	Prevalencia	Detectabilidad	Técnico	Negocio	
A1: 2017- Inyección	Específico de la Aplicación	FACIL: 3	COMUN: 2	FACIL: 3	GRAVE: 3	Específico de la Aplicación	8,0
A2: 2017 - Pérdida de Autenticación	Específico de la Aplicación	FACIL: 3	COMUN: 2	PROMEDIO: 2	GRAVE: 3	Específico de la Aplicación	7,0
A3: 2017 - Exposición de Datos Sensibles	Específico de la Aplicación	PROMEDIO: 2	DIFUNDIDO: 3	PROMEDIO: 2	GRAVE: 3	Específico de la Aplicación	7,0
A4: 2017 - Entidad Externa de XML (XXE)	Específico de la Aplicación	PROMEDIO: 2	COMUN: 2	FACIL: 3	GRAVE: 3	Específico de la Aplicación	7,0
A5: 2017 - Pérdida de Control de Acceso	Específico de la Aplicación	PROMEDIO: 2	COMUN: 2	PROMEDIO: 2	GRAVE: 3	Específico de la Aplicación	6,0
A6: 2017 - Configuración de Seguridad Incorrecta	Específico de la Aplicación	FACIL: 3	DIFUNDIDO: 3	FACIL: 3	MODERADO: 2	Específico de la Aplicación	6,0
A7: 2017 - Secuencia de Comandos en Sitios Cruzados (XSS)	Específico de la Aplicación	FACIL: 3	DIFUNDIDO: 3	FACIL: 3	MODERADO: 2	Específico de la Aplicación	6,0
A8: 2017 - Deserialización Insegura	Específico de la Aplicación	DIFICIL: 1	COMUN: 2	PROMEDIO: 2	GRAVE: 3	Específico de la Aplicación	5,0
A9: 2017 - Componentes con Vulnerabilidades Conocidas	Específico de la Aplicación	PROMEDIO: 2	DIFUNDIDO: 3	PROMEDIO: 2	MODERADO: 2	Específico de la Aplicación	4,7
A10: 2017 - Registro y Monitoreo Insuficientes	Específico de la Aplicación	PROMEDIO: 2	DIFUNDIDO: 3	DIFICIL: 1	MODERADO: 2	Específico de la Aplicación	4,0

Figura 2-4 - Top 10 de riesgos más críticos en aplicaciones Web OWASP **[42]**.

En la Figura 2.4 OWASP muestra la relevancia de identificar y controlar principalmente las vulnerabilidades de inyección de código, perdida de autenticación y exposición de datos sensibles en los servicios web. En dicha investigación OWASP registra los modos más comunes de explotar estas vulnerabilidades e identificar cuáles serían los controles para que no sigan siendo aprovechadas por los atacantes y utiliza algunos ejemplos para otorgar un mayor entendimiento.

Por su parte el Incibe Cert publica el boletín el 23 de enero de 2020, donde evalúa las vulnerabilidades más explotadas en las empresas dedicadas al control industrial y como son utilizados los servicios web para consumir y consultar su información, dicho informe se expresa en el top de vulnerabilidades de la Figura 2.5.

Figura 2-5 - Top Ten vulnerabilidades Incibe Cert. **[85]**

Según la información expuesta en la Figura 2.5, se identifica que la mayoría de las vulnerabilidades corresponden a denegación de servicios e intento de robo de credenciales mediante fuerza bruta. Con el fin, de dar sentido a los informes de amenazas y vulnerabilidades anteriormente descritos, se establece la Tabla 2, donde se evidencian las amenazas y vulnerabilidades que son tenidas en consideración por cada entidad. Cabe resaltar, que para su desarrollo se tuvieron en cuenta las vulnerabilidades y amenazas relacionadas con los servicios web, dado que la investigación está orientada a los servicios web.

Tabla 2.3 - Aplicabilidad de las vulnerabilidades según los entes de seguridad.

Páginas Oficiales / **Amenazas Vulnerabilidades**	OWASP	CCN-CERT	INCIBE	MITRE
DoS	X	X	X	X
Ejecución de Código	X			X
Overflow		X		X
Inyección de Código	X	X	X	X
XSS	X			X
Http Response Splitting	X			X
Gain Information		X		X
Gain Privileges		X		X
CSRF			X	X
File Inclusion			X	X
Cryptojacking		X		
Exposición de datos sensibles	X	X	X	X
Robo de Identidad	X	X		X
Deserialización insegura	X			
Componentes con vulnerabilidades conocidas	X			
Pérdida de control de acceso	X			X
Falsificación de petición en sitios cruzados	X		X	

Configuración indebida	X			

Teniendo en cuenta la aplicabilidad e importancia que le dan en los referentes de seguridad, según la Tabla 2.3 es posible determinar que las vulnerabilidades más relevantes son: denegación de servicio, inyección de código y exposición de datos sensibles.

A continuación, en la Tabla 2.4 se determina cuáles de los ataques mencionados en el marco teórico tienen afectación o pueden servir para explotar las tres vulnerabilidades más relevantes.

Tabla 2.4 - Relevancia y Pertinencia de los Ataques a Servicios Web vs las Vulnerabilidades.

Vulnerabilidades / **Tipos de Ataques**	Asociado a Denegación de Servicios	Asociado a Inyección de código	Asociado a Exposición de datos sensibles
Ataque con XML de gran tamaño	X		
Análisis coercitivo	X		
Ataque de entidad y referencia XML	X		
Inundación XML	X		
Criptografía recursiva	X		
Denegación de servicio SOAP	X		
Morphing malicioso	X		
Desvío de enrutamiento	X		
Envenenamiento de esquema	X		
Spoofing WSDL	X		

Vulnerabilidades / **Tipos de Ataques**	Asociado a Denegación de Servicios	Asociado a Inyección de código	Asociado a Exposición de datos sensibles
Reescritura XML		X	X
Exclusión de firma XML			X
Intrusión de Mensajes		X	
Divulgación WSDL			X
Ataques adaptativos de texto cifrado		X	X
Ataque repetitivo		X	
Fuerza Bruta	X	X	X
Inyección XML		X	
Inyección Xpath		X	

En ese sentido, en la Tabla 2.4 se evidencia que, un atacante utiliza la técnica de fuerza bruta se puede explotar las tres vulnerabilidades, esto se logra considerar, si se logra acceder, ya sea, a un servidor principal o a una base de datos, teniendo acceso a la información sensible e incluso a denegar los servicios una vez tengan conocimiento de la estructura y arquitectura de la red. No obstante, los estudios de vulnerabilidades mostrados dentro de esta investigación, en su mayoría se encuentran basados en la información publicada por MITRE desde CWE, quienes adicionalmente emiten en los años 2011, 2019 y 2020 el top 25 de vulnerabilidades, las cuales cuentan con una puntuación, extraída desde la base de datos NVD, por medio de estos, se realizan una puntuación numérica, que corresponde a la gravedad potencial de una vulnerabilidad basada en un conjunto de características sobre la vulnerabilidad.

A continuación, en las Tabla 2.5,2.6,2.7 se encuentran los informes publicados por MITRE en los años 2011, 2019 y 2020 respectivamente.

Tabla 2.5. Top 25 MITRE- CWE 2011 [86].

Rank	ID	Name	Score
[1]	CWE-89	Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection')	93.8
[2]	CWE-78	Improper Neutralization of Special Elements used in an OS Command ('OS Command Injection')	83.3
[3]	CWE-120	Buffer Copy without Checking Size of Input ('Classic Buffer Overflow')	79.0
[4]	CWE-79	Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting')	77.7
[5]	CWE-306	Missing Authentication for Critical Function	76.9
[6]	CWE-862	Missing Authorization	76.8
[7]	CWE-798	Use of Hard-coded Credentials	75.0
[8]	CWE-311	Missing Encryption of Sensitive Data	75.0
[9]	CWE-434	Unrestricted Upload of File with Dangerous Type	74.0
[10]	CWE-807	Reliance on Untrusted Inputs in a Security Decision	73.8
[11]	CWE-250	Execution with Unnecessary Privileges	73.1
[12]	CWE-352	Cross-Site Request Forgery (CSRF)	70.1
[13]	CWE-22	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal')	69.3
[14]	CWE-494	Download of Code Without Integrity Check	68.5
[15]	CWE-863	Incorrect Authorization	67.8
[16]	CWE-829	Inclusion of Functionality from Untrusted Control Sphere	66.0
[17]	CWE-732	Incorrect Permission Assignment for Critical Resource	65.5
[18]	CWE-676	Use of Potentially Dangerous Function	64.6
[19]	CWE-327	Use of a Broken or Risky Cryptographic Algorithm	64.1
[20]	CWE-131	Incorrect Calculation of Buffer Size	62.4
[21]	CWE-307	Improper Restriction of Excessive Authentication Attempts	61.5
[22]	CWE-601	URL Redirection to Untrusted Site ('Open Redirect')	61.1
[23]	CWE-134	Uncontrolled Format String	61.0
[24]	CWE-190	Integer Overflow or Wraparound	60.3
[25]	CWE-759	Use of a One-Way Hash without a Salt	59.9

Tabla 2.6. Top 25 MITRE - CWE 2019 [87].

Rank	ID	Name	Score
[1]	CWE-119	Improper Restriction of Operations within the Bounds of a Memory Buffer	75.56
[2]	CWE-79	Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting')	45.69
[3]	CWE-20	Improper Input Validation	43.61
[4]	CWE-200	Information Exposure	32.12
[5]	CWE-125	Out-of-bounds Read	26.53
[6]	CWE-89	Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection')	24.54
[7]	CWE-416	Use After Free	17.94
[8]	CWE-190	Integer Overflow or Wraparound	17.35
[9]	CWE-352	Cross-Site Request Forgery (CSRF)	15.54
[10]	CWE-22	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal')	14.10
[11]	CWE-78	Improper Neutralization of Special Elements used in an OS Command ('OS Command Injection')	11.47
[12]	CWE-787	Out-of-bounds Write	11.08
[13]	CWE-287	Improper Authentication	10.78
[14]	CWE-476	NULL Pointer Dereference	9.74
[15]	CWE-732	Incorrect Permission Assignment for Critical Resource	6.33
[16]	CWE-434	Unrestricted Upload of File with Dangerous Type	5.50
[17]	CWE-611	Improper Restriction of XML External Entity Reference	5.48
[18]	CWE-94	Improper Control of Generation of Code ('Code Injection')	5.36
[19]	CWE-798	Use of Hard-coded Credentials	5.12
[20]	CWE-400	Uncontrolled Resource Consumption	5.04
[21]	CWE-772	Missing Release of Resource after Effective Lifetime	5.04
[22]	CWE-426	Untrusted Search Path	4.40
[23]	CWE-502	Deserialization of Untrusted Data	4.30
[24]	CWE-269	Improper Privilege Management	4.23
[25]	CWE-295	Improper Certificate Validation	4.06

Tabla 2.7. Top 25 MITRE - CWE 2020 [88]

Rank	ID	Name	Score
[1]	CWE-79	Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting')	46.82
[2]	CWE-787	Out-of-bounds Write	46.17
[3]	CWE-20	Improper Input Validation	33.47
[4]	CWE-125	Out-of-bounds Read	26.50
[5]	CWE-119	Improper Restriction of Operations within the Bounds of a Memory Buffer	23.73
[6]	CWE-89	Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection')	20.69
[7]	CWE-200	Exposure of Sensitive Information to an Unauthorized Actor	19.16
[8]	CWE-416	Use After Free	18.87
[9]	CWE-352	Cross-Site Request Forgery (CSRF)	17.29
[10]	CWE-78	Improper Neutralization of Special Elements used in an OS Command ('OS Command Injection')	16.44
[11]	CWE-190	Integer Overflow or Wraparound	15.81
[12]	CWE-22	Improper Limitation of a Pathname to a Restricted Directory ('Path Traversal')	13.67
[13]	CWE-476	NULL Pointer Dereference	8.35
[14]	CWE-287	Improper Authentication	8.17
[15]	CWE-434	Unrestricted Upload of File with Dangerous Type	7.38
[16]	CWE-732	Incorrect Permission Assignment for Critical Resource	6.95
[17]	CWE-94	Improper Control of Generation of Code ('Code Injection')	6.53
[18]	CWE-522	Insufficiently Protected Credentials	5.49
[19]	CWE-611	Improper Restriction of XML External Entity Reference	5.33
[20]	CWE-798	Use of Hard-coded Credentials	5.19
[21]	CWE-502	Deserialization of Untrusted Data	4.93
[22]	CWE-269	Improper Privilege Management	4.87
[23]	CWE-400	Uncontrolled Resource Consumption	4.14
[24]	CWE-306	Missing Authentication for Critical Function	3.85
[25]	CWE-862	Missing Authorization	3.77

Para obtener las vulnerabilidades más relevantes según MITRE, se realiza una ponderación de cada CWE mediante un promedio, esto según el score y cada año de publicación, expresando los resultados en la Tabla 2.8

Tabla 2.8. Top 25 MITRE - CWE 2011, 2019 y 2020.

CWE ID	SCORE			PROMEDIO
	2011	2019	2020	
CWE-79	77.7	45.69	46.82	56,73
CWE-89	93.8	24.54	20.69	46,34
CWE-78	83.3	11.47	16.44	37,07
CWE-352	70.1	15.54	17.29	34,31
CWE-119	NO PUBLICADO	75.56	23.73	33,09
CWE-22	69.3	14.10	13.67	32,35
CWE-190	60.3	17.35	15.81	29,4
CWE-434	74.0	5.50	7.38	28,96
CWE-798	75.0	5.12	5.19	28,43
CWE-306	76.9	NO PUBLICADO	3.85	26,91
CWE-862	76.8	NO PUBLICADO	3.77	26,85
CWE-120	79.0	NO PUBLICADO	NO PUBLICADO	26,33
CWE-732	65.5	6.33	6.95	26,26
CWE-311	75.0	NO PUBLICADO	NO PUBLICADO	25
CWE-807	73.8	NO PUBLICADO	NO PUBLICADO	24,6
CWE-250	73.1	NO PUBLICADO	NO PUBLICADO	24,36
CWE-494	68.5	NO PUBLICADO	NO PUBLICADO	22,83
CWE-863	67.8	NO PUBLICADO	NO PUBLICADO	22,6
CWE-829	66.0	NO PUBLICADO	NO PUBLICADO	22
CWE-676	64.6	NO PUBLICADO	NO PUBLICADO	21,53
CWE-327	64.1	NO PUBLICADO	NO PUBLICADO	21,36
CWE-131	62.4	NO PUBLICADO	NO PUBLICADO	20,8
CWE-307	61.5	NO PUBLICADO	NO PUBLICADO	20,5
CWE-601	61.1	NO PUBLICADO	NO PUBLICADO	20,36
CWE-134	61.0	NO PUBLICADO	NO PUBLICADO	20,33
CWE-759	59.9	NO PUBLICADO	NO PUBLICADO	19,96
CWE-787	NO PUBLICADO	11.08	46.17	19,08
CWE-125	NO PUBLICADO	26.53	26.50	17,67
CWE-200	NO PUBLICADO	32.12	19.16	17,09
CWE-20	NO PUBLICADO	43.61	33.47	12,69
CWE-416	NO PUBLICADO	17.94	18.87	12,27
CWE-287	NO PUBLICADO	10.78	8.17	6,31
CWE-476	NO PUBLICADO	9.74	8.35	6,03
CWE-94	NO PUBLICADO	5.36	6.53	3,96
CWE-611	NO PUBLICADO	5.48	5.33	3,6
CWE-502	NO PUBLICADO	4.30	4.93	3,07
CWE-400	NO PUBLICADO	5.04	4.14	3,06
CWE-269	NO PUBLICADO	4.23	4.87	3,03
CWE-522	NO PUBLICADO	NO PUBLICADO	5.49	1,83
CWE-772	NO PUBLICADO	5.04	NO PUBLICADO	1,68
CWE-426	NO PUBLICADO	4.40	NO PUBLICADO	1,46
CWE-295	NO PUBLICADO	4.06	NO PUBLICADO	1,35

El resultado de la ponderación arroja que las vulnerabilidades más relevantes y que siempre ha tenido en cuenta MITRE en sus informes son:

CWE 79 - *Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting')*, CWE 89 - *Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection')* y CWE 78 - *Improper Neutralization of Special Elements used in an OS Command ('OS Command Injection'*)
Las cuales tienen directa similitud con las vulnerabilidades anteriormente descritas en la Tabla 2 pero en este caso haciendo relación a la familia de CWE que corresponden.

Por lo anterior, se han obtenidos para esta fase 1, las diferentes amenazas hacia sitios Web, así como las tres (3) vulnerabilidades más relevantes que son:

- Asociado a Denegación de Servicios
- Asociado a Inyección de código
- Asociado a Exposición de datos sensibles

2.2 Fase 2. Identificación de Indicadores de Compromiso Existentes

2.2.1 Metodología

Para determinar los indicadores de compromisos existentes que permiten el desarrollo y los ajustes de un set de indicadores para los servicios Web, se empleó el método deductivo. Este método está basado en el razonamiento desde la deducción propia del ser humano, va de lo general a lo particular por medio de un análisis comprobado de las empresas dedicadas a entregar IOC de manera masiva [79].

Para identificar los Indicadores de Compromiso a Servicio Web, se realiza un rastreo de fuentes secundarias oficiales que entregan IOC de manera masiva y sin ánimo de lucro. Los resultados de esta consulta, permitieron determinar tres proyectos de código libre: Alientvault, virus total, MISC Project. Además, se realizó un estudio documental de cada proyecto, por medio del cual, se pudo identificar su funcionamiento y la forma en la cual se distribuye la información.

Se realizo una búsqueda en las plataformas del Virus Total [89] y MiscProject [90], determinando si un IOC, se encuentra relacionado con los incidentes de seguridad reportados por las empresas que se encontraban adscritas al proyecto MiscProject. Por su parte, el proyecto Alien Vault, sí proporciona IOC, de diferentes tipos, orientados a Servicios Web, para lo cual también fue consultado.

Para obtener los Indicadores de Compromiso, se realizó el registro en Alien Vault, y una revisión bibliográfica, luego se extrajeron con la herramienta insomnia, mediante el consumo del servicio web de la URL(https://insomnia.rest/), posteriormente y con el fin de compartirlos, se utiliza el estándar OpenIOC y la herramienta MANDIANT, por medio de Power QWERTY de Excel, lo que permite automatizar su escritura empleando la herramienta UIPath.

2.2.2. Resultados

Con base en, los resultados obtenidos en la Fase 1 y a partir de las tres vulnerabilidades definidas, se identificaron diferentes IOC existentes. Por lo cual, inicialmente se pretendió utilizar tres grandes proyectos que se centran en documentar los indicadores de compromiso como lo son *Alienvault* OTX, Virus total y MISP Project, aunque, al validar la documentación de dichos proyectos, los cuales exponen su funcionamiento y la forma de consumirlos a través de servicios web constantemente expuestos en internet, se logra evidenciar que, la funcionalidad que brinda Virus Total, como MISP Project, se centra en consultar si un indicador de compromiso está o no registrado como negativo, o si este, se encuentra asociado a listas negras, establecidas por la seguridad de la información, por lo cual, ambos proyectos no fueron tenidos en cuenta, dado que no tienen la información requerida para este proyecto.. Cabe destacar que, para identificar los diferentes indicadores de compromisos, no se cuenta con un sistema para su validación. Por otro lado, el proyecto *Alienvault* permite la validación de diferentes IOC que puedan aplicarse al proyecto y en ese sentido, se hizo uso de dicha plataforma.

En la exploración del proyecto *Alienvault*, se identificó que, al utilizar el ambiente gráfico, ingresando a la URL https://otx.alienvault.com/browse. Una vez iniciada la sesión, y filtrar por indicadores de servicios web, se encontraron los datos registrados en las bases de datos negativas del proyecto, obteniendo como resultado diferentes IP, dominios, nombres de máquina y URL que anteriormente fueron usados para ejecutar un ataque de seguridad en algún servicio de la

infraestructura expuesta por el proyecto, además, hay indicadores de compromiso que son registrados por usuarios finales y empresas.

Para la indagación realizada, se tuvo en cuenta las siguientes cadenas de búsqueda, asociadas a las 3 vulnerabilidades encontradas en la fase 1:

- *denegation of service y brute force (negación de servicio),*
- *Code injection,* SQL *injection (Asociado a la inyección de código)*
- *sensitive* data *exposure (Asociado a la exposición de datos sensibles).*

A continuación, se demuestra la forma en que se obtiene la información de la base de datos AlienVault OTX, con el fin de establecer si hay IoC asociados. Una vez se inicia sesión en la URL de *Alienvault* se selecciona el menú "*Browse*" ubicado en la parte superior de la página, luego, se realizan los filtros correspondientes para que el sistema presente los nombres de máquina asociados a ataques web y se observan los resultados expuestos en la Figura 2.10.

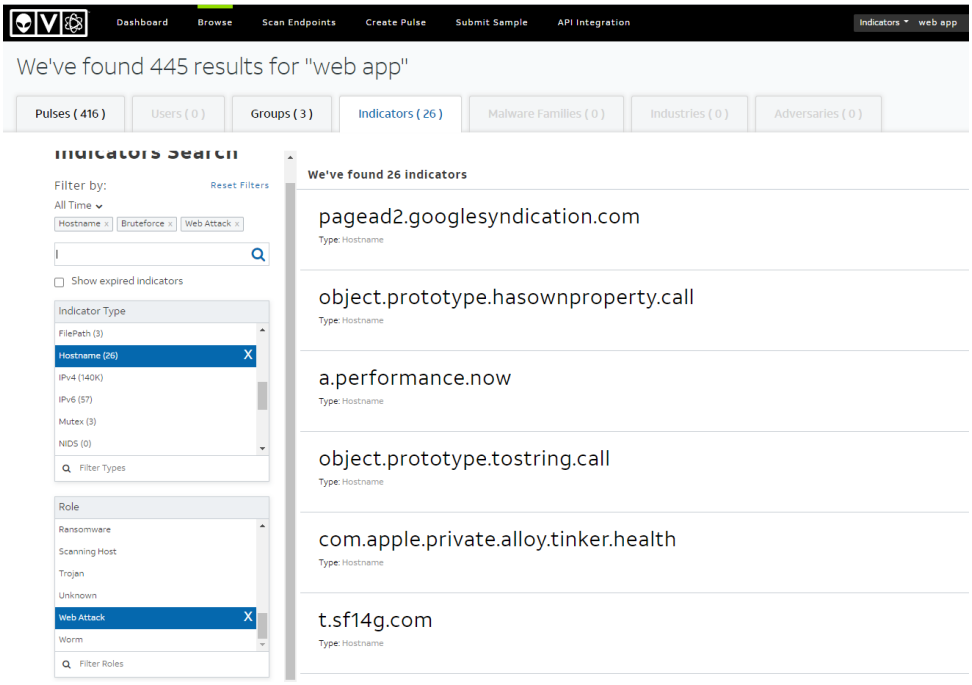

Figura 2-6 - Ejemplo de búsqueda en la interfaz gráfica de la base de datos AlienVault OTX [91].

Según lo establecido en la Figura 2.6, se visualizan los resultados de las búsquedas, pero no es posible realizar ningún tipo de exportación, por ende, se procedió a utilizar otra forma de hallar la información.

Dado que no fue posible la extracción, se consumen los servicios web o también conocidos como API, por medio de un aplicativo cliente denominado REST. Para este caso, se utiliza la herramienta Insomnia, la cual tiene como función, realizar peticiones tipo HTTP, usando parámetros específicos, entre ellos las características de los IOC a consultar, la URL de la API y por supuesto, la llave de conexión que se puede encontrar al momento de iniciar sesión en el aplicativo *Alienvault* e ingresar al submenú API *integration* tal como muestra la Figura 2. 7

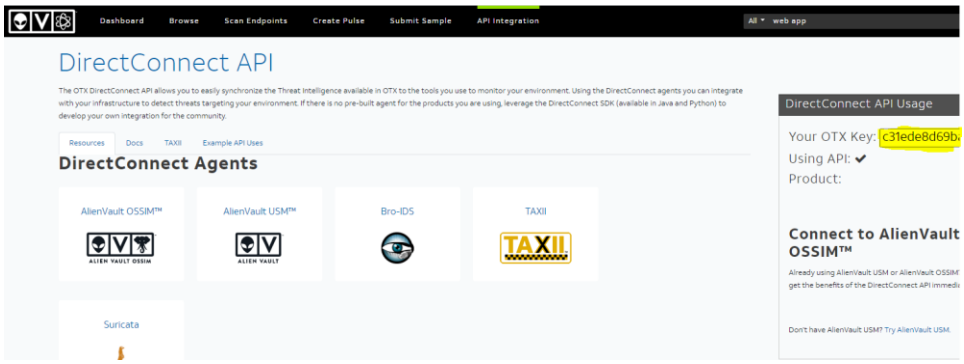

Figura 2-7 - Ambiente donde AlienVault OTX otorga la llave de conexión [92]

En *insomnia* se crea una nueva petición y se le otorga un nombre, a continuación, se presenta como se configura el *Header*. Inicialmente se ingresa el nombre del *Header*, que según la documentación del sitio es "X-OTX-API-KEY" y en el valor se ingresa la cadena de texto proporcionada como llave para la conexión. En el tipo de petición se selecciona Get y se introduce la URL, el cual contiene el API, la cual según la documentación sería: https://otx.alienvault.com/api/v1/indicators/export. Para llevar a cabo los filtros se debe incluir en la URL el signo de interrogación "?" y los filtros correspondientes. Para evidenciar las IPv4, los dominios, nombre de máquina y URL reportados se utiliza el siguiente filtro "*types*=IPv4, domain,hostname,URL" y adicionalmente se establece una fecha de referencia, para que tome solo los registros de esa fecha en adelante, se debe tener en cuenta la función *modified_&*"*since*=2019-01-01T12:35:00+00:00" como se aprecia en la Figura 12. En este caso de estudio, los datos fueron consultados entre el 1 de enero de 2019 y 1 de enero de 2021 con el fin de acotar el número de indicadores de compromiso, dado el excesivo número de registros existente desde el inicio del proyecto *Alienvault*.

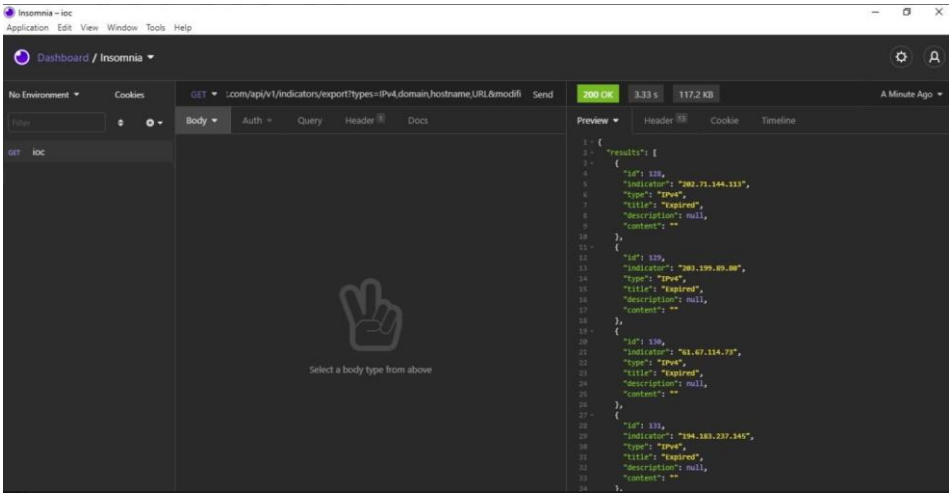

Figura 2-8 - Configuración de la petición en el software Imsomnia.

La Figura 2.8 muestra como son visualizados los resultados de la petición anterior a la fecha de 1 de enero de 2021. El resultante es un *Sting* de 8007 líneas, el cual contiene 999 indicadores de compromiso divido en los tipos IP, dominio, DNS y nombre de máquinas.


```
  "next": "https://otx.alienvault.com/api/v1/indicators/export?modified_since=2020-06-01T12%3A35%3A00%2B00%3A00&limit=10000&page=2&types=IPv4%2Cdomain%2Chostname%2CURL",
  "results": [
    {
      "indicator": "82.221.100.55",
      "description": null,
      "title": null,
      "content": "",
      "type": "IPv4",
      "id": 476
    },
    {
      "indicator": "weksrubaz.ru",
      "description": null,
      "title": null,
      "content": "",
      "type": "domain",
      "id": 2686
    },
    {
      "indicator": "mifastubiv.ru",
      "description": null,
      "title": null,
      "content": "",
      "type": "domain",
      "id": 2690
    },
    {
      "indicator": "adawareblock.com",
      "description": null,
      "title": null,
      "content": "",
      "type": "domain",
      "id": 5295
    },
    {
```


Figura 2-9 - Resultados de la petición en el software Imsomnia.

El String resultante fue registrado en Excel y ordenado mediante una consulta de la extensión de Excel Power Qwerty, para dar más claridad en la exposición de los datos encontrados, se utiliza la Figura 2-10 para expresar el número de indicadores encontrados por tipo.

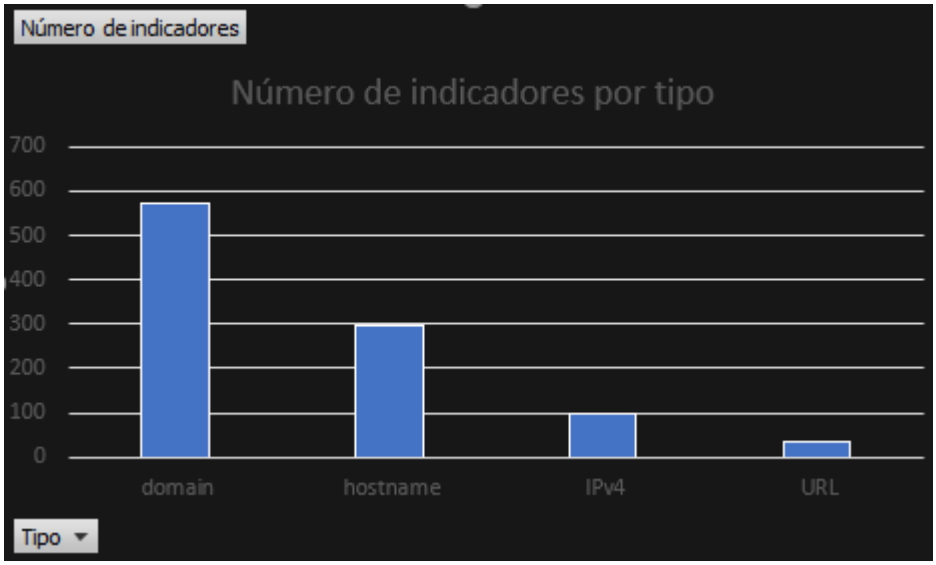

Figura 2-10 - Número de indicadores por tipo

Para identificar los indicadores de compromiso se usó el estándar Open IOC creado por CCN-CERT y la herramienta *Mandiant* IOC versión 2.2.0, donde se construyeron cinco (5) indicadores de compromiso con extensión (IOC). El primero corresponde a la agrupación de todos los indicadores resultantes de la búsqueda en *insomnia*, usando el conector lógico OR, el cual permite indicar que, si uno de estos indicadores llega a un sistema, dicho sistema posiblemente se encuentre bajo un ataque de seguridad y los otro cuatro (4) corresponden a la agrupación por tipo, es decir, uno para las IP, otro para las URL, el siguiente para los nombres de máquina y el ultimo para los DNS.

Por ejemplo, para crear el IOC de las IP encontradas, se ingresa al aplicativo *Mandiant* IOC, se define la ruta donde se van a guardar los indicadores generados y se procede a presionar CTR + N, para que el sistema cree un nuevo indicador. Una vez creado el indicado se selecciona añadir un nuevo ITEM y para el caso de las IP se selecciona el submenú "*PortItem*" y luego Port "*Remote IP*" tal como se muestra en la Figura 2-11. Se ingresa la IP y se repite el proceso para cada una de las IP.

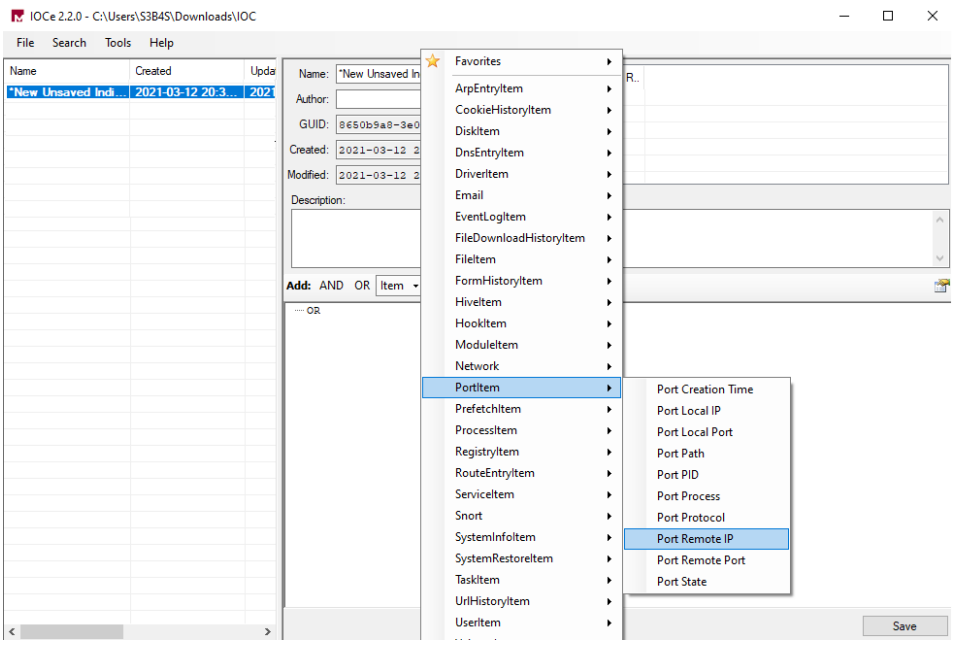

Figura 2-11 - Número de indicadores por tipo.

Teniendo en cuenta la cantidad de registros obtenidos se procedió a desarrollar una automatización, donde la herramienta Uipath que toma el archivo de Excel y genera de manera automática todos los indicadores de compromiso debido a que el tiempo a emplear para configurar 999 en diferentes ocasiones sería muy extenso (al hacerlo de forma automática se genera un proceso más simple). Se tomó esta herramienta teniendo en cuenta la experiencia y conocimientos previos. La automatización de procesos puede ser utilizada basado en los conocimientos de quien la vaya a utilizar.

Como resultado final de esta fase 2, se generan los siguientes 5 IOC nombrados así:

- Alien Vault IOC.ioc
- Alien Vault IOC IP.ioc
- Alien Vault IOC URL.ioc
- Alien Vault IOC DNS.ioc
- Alien Vault IOC HOSTNAME.ioc

El desarrollo de los IOC fue vinculado como evidencia a la investigación dentro de los entregables finales como anexos. Para esta investigación, se considera relevante lo realizado en esta fase, dado que, después de la implementación de la *honeypot*, se realizó el mismo procedimiento para generar

los indicadores de compromiso resultantes de los vectores de ataque encontrados en las peticiones observadas en la infraestructura diseñada.

2.3 Fase 3. Implementación de la *Honeypot* Académica

2.3.1 Metodología

Para la implementación técnica, lo primero es ***seleccionar*** la *Honeypot* que pueda ser usada en las pruebas de seguridad y uso de los IOC, para lo cual, se emplea el método de investigación descriptivo, inicialmente rastreando las diferentes *Honeypot* comerciales, como se observa en el numeral 1.1.3 del estado del arte, dichas soluciones fueron clasificadas mediante características que posibilitan la recolección de los vectores de ataque a servicios web.

Para la definición de las características que debe cumplir la *Honeypot* a ser instalada, se buscaron criterios que logran hallar soluciones multiplataforma, las cuales son: Código abierto, Servicios soportados, Nivel de interacción, Soporte de archivos log, Licenciamiento libre, Disponible en *Dockerhub*, Mapa geográfico de ataques, Base de datos, Configuración de IOC.

Para ampliar la relevancia de las características se describen a continuación cada una de ellas:

- **Código abierto:** Es importante que las soluciones *Honeypot* cuenten con un código abierto con el fin de desarrollar nuevas funcionalidad o integraciones en caso de que se haga necesario, su calificación será un sí o un no, siendo un sí como respuesta tendría mayor relevancia.
- **# Servicios soportados:** Consta de un valor numérico que indica cuantos sensores *Honeypot* le integran o en su caso cuantos protocolos soporta, entre mayor valor, más usabilidad tendría la solución.
- **Nivel de interacción:** Medido entre alto, medio y bajo, siendo alto la mayor calificación, determina la complejidad de los ataques, que tiene la posibilidad de detectar.
- **Soporte de archivos log:** su calificación si/no permite verificar si la herramienta puede registrar los sucesos ocurridos durante el funcionamiento del dispositivo y la interacción con los atacantes.

- **Licenciamiento libre:** Se elige entre sí y no para su ponderación, y básicamente tendrán mayor ventaja las herramientas de código libre ya que para una implementación o investigación posterior no es necesario asumir los costos de licenciamiento y se tiene una mayor usabilidad en el tiempo.
- **Disponible en Dockerhub:** Que cuente con una implementación disponible en Docker hub permite que sea fácilmente desplegado, adicional, siendo calificado con un sí, no serían necesarias herramientas de hardware costosas en su implementación.
- **Mapa geográfico de ataques:** Permite observar el comportamiento de las peticiones revividas según la geolocalización del atacante, permite además identificar si las peticiones son recibidas de lugares donde no se cuentan con servicios o clientes.
- **Base de datos:** Las soluciones que cuenten con el uso de bases de datos permitirán un mayor manejo de registros, lo cual facilita los análisis de la información y fortalece el criterio de las decisiones que se deban tomar en base a los análisis realizados.
- **Configuración de IOC:** Este numeral es de gran importancia ya que no solo pueden ser identificados los resultados de las peticiones recibidas, sino que también, sirve para evaluar su funcionamiento mediante la adaptación de estos, sus posibilidades son si/no.

Luego de la comparación, se selecciona la *Honeypot* que más cumple con las características anteriores, implementando un ambiente controlado que permite la captura de los diferentes ataques informáticos y las actividades no autorizadas, para lo cual, se contó con una configuración de IOC, en los en los temas de interacción, la que tuviese “alto” y cumpliera con las demás características.

Seguidamente, se procedió a la implementación en máquinas virtuales, utilizando virtual Box, y para llevar a cabo la implementación, se diseñó una arquitectura básica que simulara los diferentes servicios web.

La arquitectura contó con los siguientes componentes:

- Internet.
- Red domestica utilizando un router que cuente con configuración de DMZ.
- Estación con Kali Linux.
- Servidor correlacionador de eventos Wazuh.
- Servidor de monitoreo de disponibilidad Zabbix.
- Servidor con Ubuntu donde se encuentra instalado un servicio Web.

- Servidor de aplicaciones vulnerable de OWASP.

Finalmente se realizaron las configuraciones pertinentes para que el servidor Ubuntu fuera monitoreado por el servidor Zabbix y sea reportada toda actividad anormal, respecto a disponibilidad. Además, se configura el servidor web vulnerable, para que todas las peticiones que le ingresen sean monitoreadas e interpretadas por Wazuh, y por último dicha maquina se configura en la DMZ para que cualquier atacante pueda acceder y así observar las peticiones y vectores de ataque utilizados para la explotación de sus vulnerabilidades.

2.3.2 Resultados

Para la implementación de una *Honeypot*, bajo un ambiente controlado que permitiera la captura de los diferentes ataques informáticos y las actividades no autorizadas se realizó un estudio donde se identificaron diferentes soluciones *Honeypot*, clasificadas mediante características que posibilitaron la recolección de los vectores de ataque a servicios web. Para la definición de las características se indagaron los criterios para hallar soluciones multiplataforma, con motor de base de datos, con alto registro de logs, de licenciamiento libre, con posibilidad de configuración de IOC.

Tabla 2.9 - Comparaciones características *Honeypot*.

Características / *Honeypot*	Código abierto	Servicios soportados	Nivel de interacción	Soporte de archivos log	Licenciamiento libre	Disponible en Dockerhub	Mapa geográfico de ataques	Base de datos	Configuración de IOC
Trap-X	No	24	Alta	Si	No	No	Si	Si	Si
Allure	No	16	Media	Si	No	No	Si	Si	No
StrongARM	No	13	Media	Si	No	No	No	Si	No
LogRhyhm	No	10	Alta	Si	No	No	No	Si	No
Dionaea	SI	5	Baja	Si	Si	No	No	No	No
Kippo	Si	2	Baja	Si	Si	No	No	No	No
DejaVu	Si	7	Media	Si	Si	No	No	No	No
HoneyDrive	No	14	Alta	Si	Si	No	Si	No	No

MHN	Si	Ilimitados	Alta	Si	Si	Si	Si	Si	No
Honeypot de alta interacción	Si	Ilimitados	Alta	Si	Si	Si	Si	Si	Si

La Tabla 2.9, expone que las *Honeypot* que reúnen el mayor número de características son Trap-X, MHN (*Modern Honey Network*) y la *honeypot* de alta interacción, no obstante, la implementación de Trap-X no es viable, puesto que, no tiene un código abierto que permita realizar modificaciones o adaptaciones a la medida. Además, es necesaria una suscripción anual, costo que no es factible asumir en un ambiente académico. Por su parte, la MHN es una *honeynet*, la cual, no cuenta con la opción de configurar indicadores de compromisos. No obstante, se elaboró la implementación de una *honeypot* de alta interacción, que posee herramientas de código abierto y licenciamiento libre. Permite soportar e interpretar logs, para ser implementadas bajo un ambiente de contenerización[1], lo que brinda la posibilidad de geolocalización y mapeo de ataques, son integrables, con bases de datos de tipo *open source* y posibilitan la configuración de reglas, como la configuración de IOC. Estas herramientas agrupadas en una misma red loca, pueden suplir la necesidad de identificar los vectores de ataques a servicios web.

Para la implementación de la *honeypot* en un ambiente controlado y para la captación de los ataques informáticos y actividades no autorizadas, se utilizó un equipo de cómputo con un sistema operativo Windows 10, acceso a internet y el programa virtual Box para la ejecución de las máquinas virtuales. Teniendo en cuenta la definición de *honeypot* de alta interacción, la cual indica que, se realizó la implementación de un servicio Web, el cual sirve para comentar y calificar los lugares visitados en la internet sobre un servidor de Ubuntu versión 20.04, adicionalmente se configuró el proyecto de máquinas web vulnerables de owasp con kernel de Ubuntu y mejor conocido como *"Broken Web Aplications Virtual Machine"* , también se usó Zabbix un monitor de disponibilidad y harwdware de red el cual permite comprobar el estado de un servidor en cuanto a procesamiento, memoria RAM, espacio en disco, entre otros.

[1]Es un método de virtualización de nivel de sistema operativo (nivel OS) para implementar y ejecutar aplicaciones distribuidas sin lanzar una máquina virtual completa (VM) para cada aplicación.

Además, se utilizó Wazuh, el cual es una aplicación para el monitoreo de eventos de seguridad y tiene como función correlacionar eventos y cuenta con OSSEC y diferentes decodificadores de logs. Por último, se empleó la herramienta Kali Linux, para realizar pruebas de intrusión. Se configuró una DMZ en la red local, con el fin de recibir diferentes peticiones en el servidor de Owasp, aprovechando sus diferentes vulnerabilidades. En la Figura 2.12 se muestra la arquitectura de red configurada para la prueba de concepto.

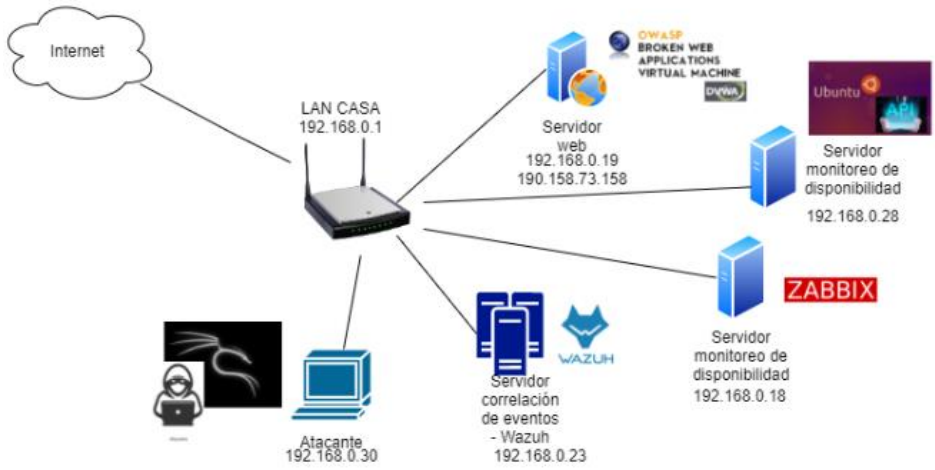

Figura 2-12 - Topología de red prueba de concepto.

El proceso para la implementación de la *Honeypot*, se encuentra documentado en el Anexo 6 “Manual de instalación *Honeypot*”; en dicho documento, adicionalmente, se muestran los sistemas activos y la configuración que se debe tener en cuenta para llevar el proceso en cada máquina virtual.

2.4 Fase 4 Evaluación de los IOC Seleccionados a través de la Información de la *Honeypot*

2.4.1 Metodología

La evaluación de los IoC está asociada a una serie de actividades que permitieron comprobar el funcionamiento de los mismos, esto es, que de acuerdo a los diferentes eventos de seguridad, la herramienta de monitoreo pudiese identificar y mostrar que se están ejecutando ataques informáticos, haciendo uso de los IoC creados, para lo cual, esta fase constó de 3 actividades:

a. Configuración de los IoC
b. Validación de resultados: exposición de los IoC en la Honeypot
c. Recomendaciones de controles para la reducción de riesgos

En ese sentido, para realizar la evaluación de los indicadores de compromiso, se expone la *Honeypot* durante dos meses usando la DMZ contemplada en la topología propuesta, pasados los dos meses, se analizaron los resultados de los ataques recibidos y los vectores de ataque que estos comprenden, se presentan los resultados de los IOC definidos en base a la recolección de datos de la *honeypot* académica, a través de las Figuras 2.24, 2.25, 2.26, 2.27, 2.28, 2.29.

Para dichos IOC, se realiza el proceso de ordenamiento y automatización, para la escritura de los mismos y son compilados en el Anexo 7. Posteriormente, se crean diferentes listas CBD con el fin de clasificar los IOC y tenerlos legibles para el Wazuh, una vez configuradas las listas, se modelan las alertas relacionadas con los IOC. Principalmente, consiste en que, se genere una alerta en caso de que exista una petición que contenga uno de los vectores de ataque conocidos. Cabe clarificar que, los vectores de ataque son todos relacionados con los servicios Web expuestos en la topología. Es posible evidenciar, como las alertas más generadas son las relacionadas con las peticiones recibidas desde China como localidad. Adicionalmente, fue posible generar recomendaciones de controles técnicos que aportan a la reducción de los riesgos de las infraestructuras.

Adicional, se entregan unas recomendaciones de controles que pueden ser aplicados en los sitios y servicios Web, con ello, poder reducir los niveles de exposición. Finalmente, se da la contextualización del cumplimiento del objetivo general.

2.4.2 Configuración de listas IOC

Con el fin de evaluar los indicadores de compromiso, se seleccionan los IOC obtenidos y se realizó un análisis de evaluación, para determinar si son de información importante, o de valor (similar a lo desarrollado en la fase 2). Se configuran los datos obtenidos, como listas negras en la herramienta Wazuh en el módulo CDB, estableciendo reglas, para que cuando se utilicen los datos, como vectores de ataque, el sistema genere una alerta configurada previamente.

Para validar la información recolectada, se expone cómo se realizó la configuración de la lista negra de IP obtenidas. Se ingresa a la máquina virtual en donde está instalado Wazuh, se accede a la carpeta "lists" con el comando "cd /var/ossec/etc/lists", se crea un nuevo archivo con el comando "vi blacklist-IP-Alienvault" y se ingresan las IP obtenidas tal como muestra la Figura 2.13.

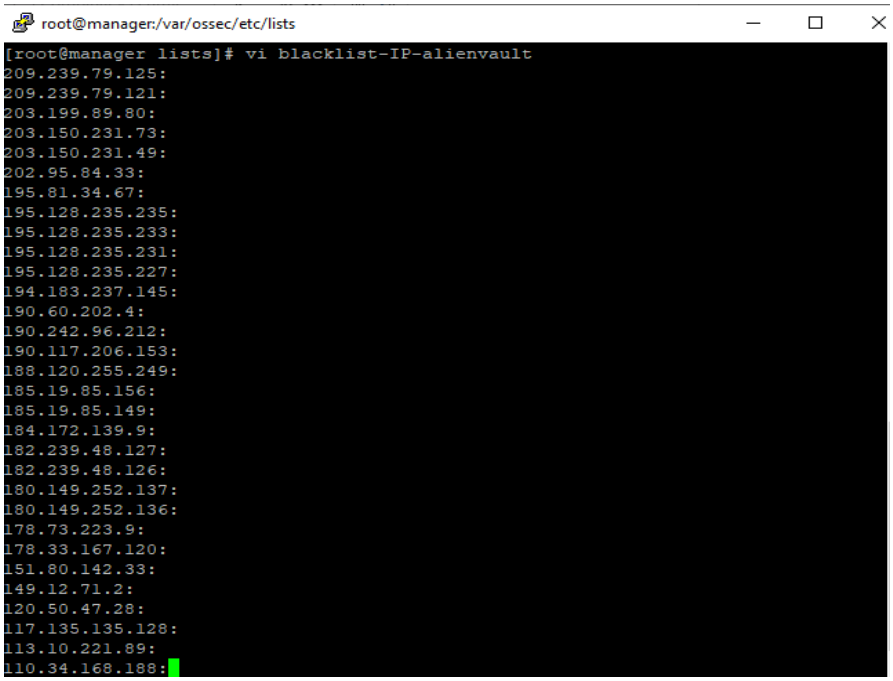

Figura 2-13 - Creación lista negra de IP.

Se ingresa al archivo de configuración de wazuh con el comando "vi /var/ossec/etc/ossec.conf" y en el ruleset, tal como se muestra en la Figura 2.14; se indica la nueva lista a tener en consideración, utilizando el comando "<list>etc/lists/blacklist-IP-alienvault</list>" y posteriormente, se usa el

comando “systemctl restart wazuh-manager” para que el wazuh tome los cambios realizados en ellos archivos de configuración.

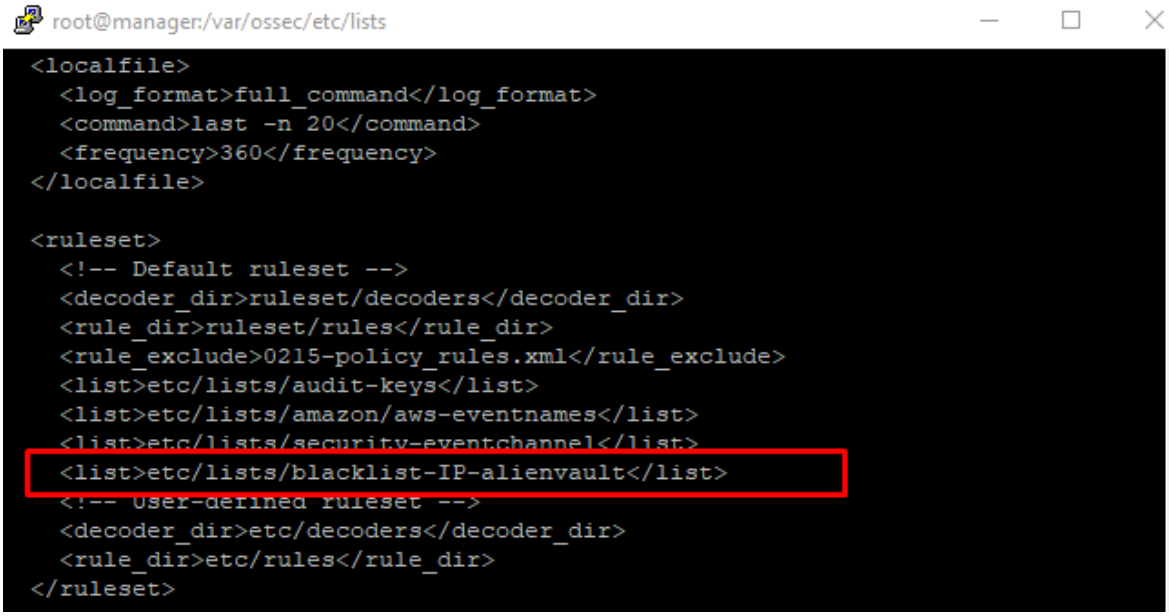

Figura 2-14 - Configuración de archivo de lista negra en Wazuh.

Una vez configurada la regla, se evidencia su existencia en el ambiente grafico de wazuh, ingresando a managment -> CDB lists tal como enseña la Figura 2.15

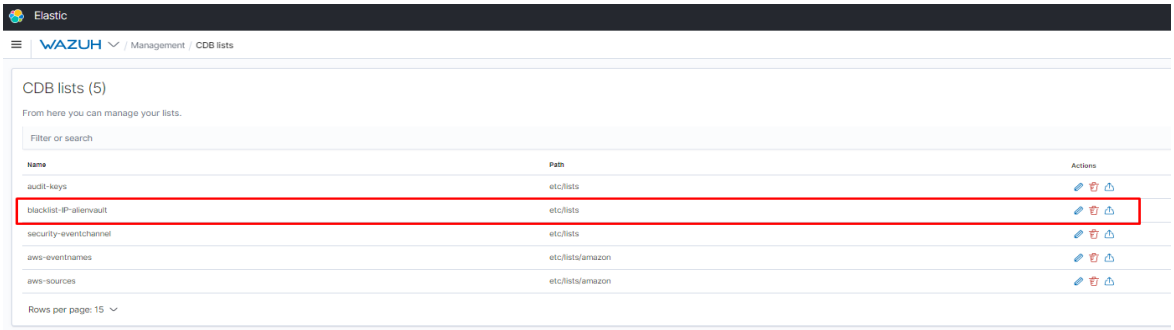

Figura 2-15 - Evidencia de que fue creada la lista en Wazuh.

En la Figura 2.16 se genera la regla que indica, como se realizó la petición desde la base de datos de las listas negras, ingresando a managment -> Rules, en este caso se asocian las listas negras con las peticiones de tipo SSH.

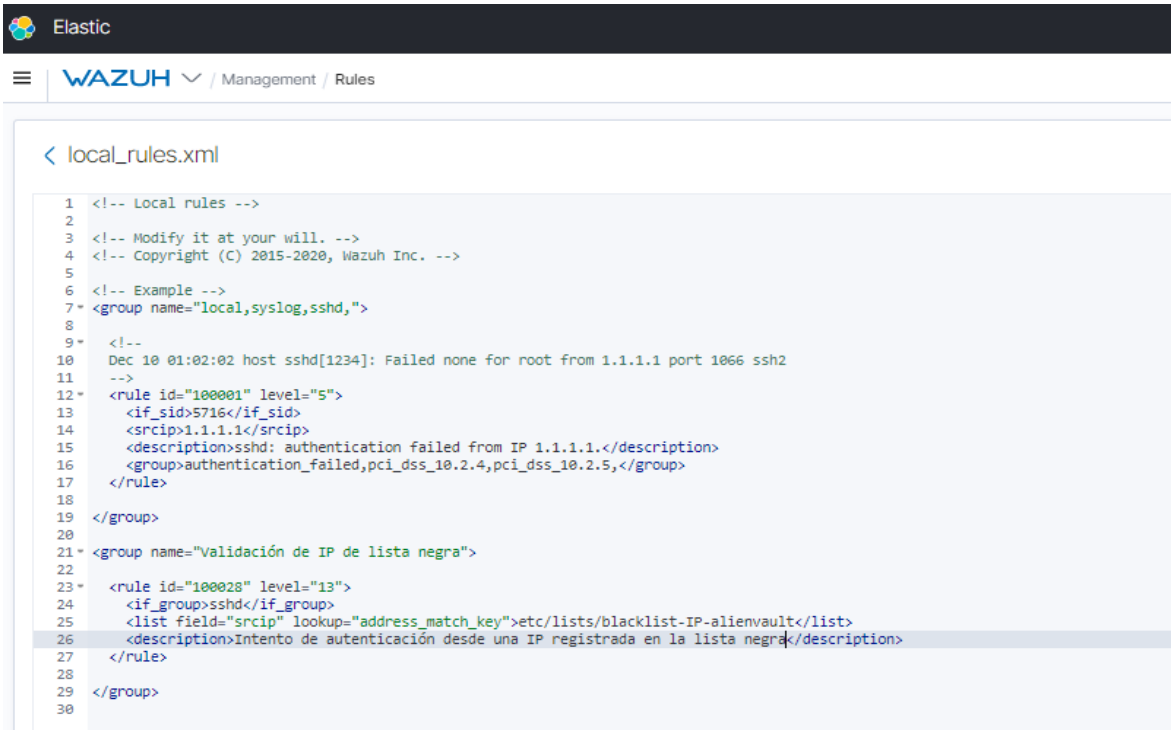

Figura 2-16 - Configuración de regla de lista negra de IP.

Para comprobar el funcionamiento de la regla se consulta la IP del equipo de Windows, expuesta en la Figura 2.17

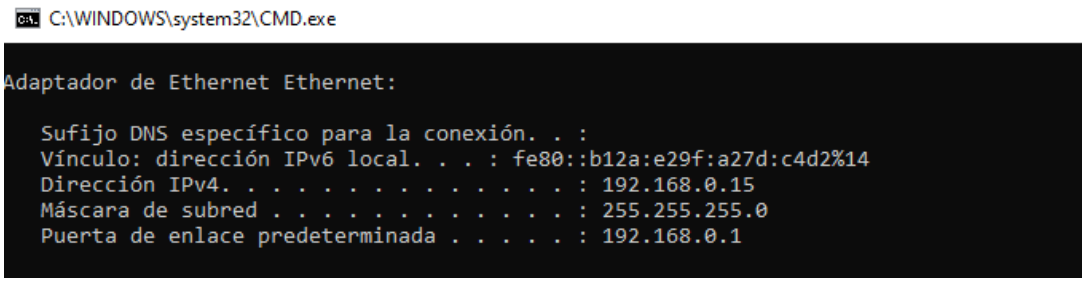

Figura 2-17. Configuración de red equipo Windows 10.

Se añade la IP 192.168.0.15 en la lista negra de IP mediante el ambiente gráfico de Wazuh, utilizando el botón *Add new entry* ubicado en la parte superior derecha. (Ver Figuras 2.18 y 2.19)

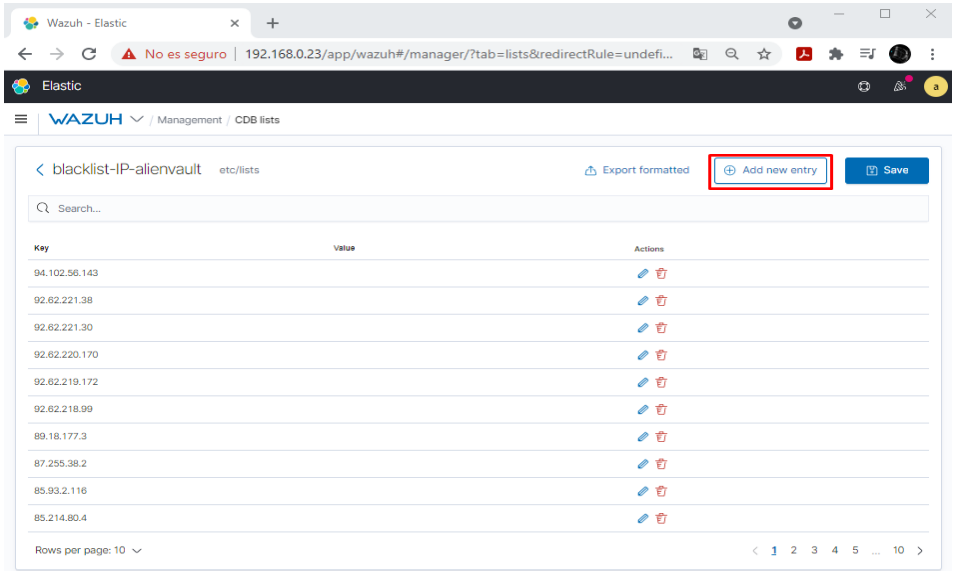

Figura 2-18 - Clic en el botón añadir nueva entrada.

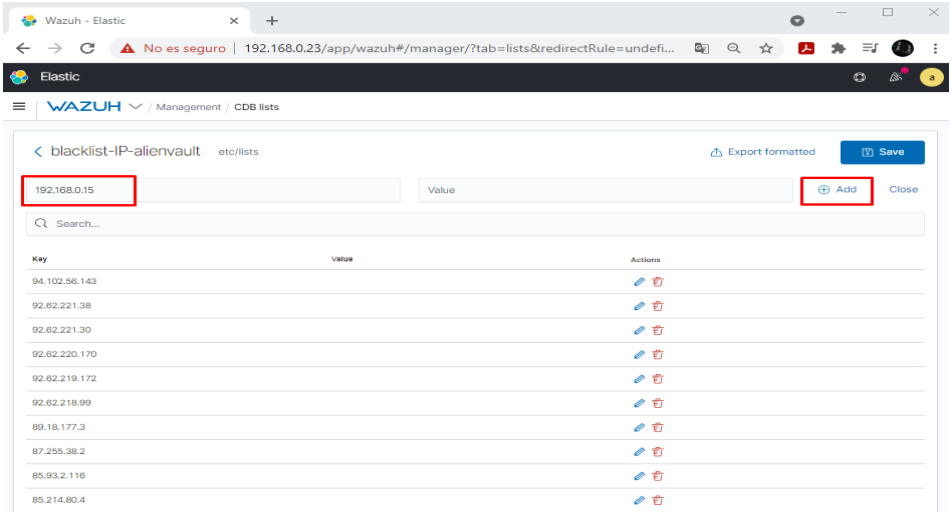

Figura 2-19 - Ingreso de la IP de Windows 10 en la lista negra de IP.

Luego de haber guardado la lista, se busca la IP para verificar la configuración, utilizando el menú de búsqueda, como se ilustra en la Figura 2.20

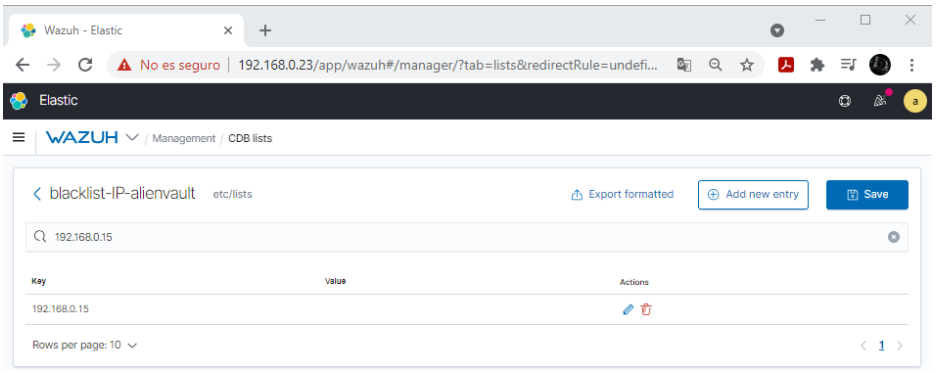

Figura 2-20 - Búsqueda de la IP de Windows 10 en la lista negra de IP.

Al realizar un intento de conexión remota al servidor owaspbwa, por medio del protocolo SSH, se evidencia como en wazuh se activa la regla configurada en el módulo de eventos de seguridad.

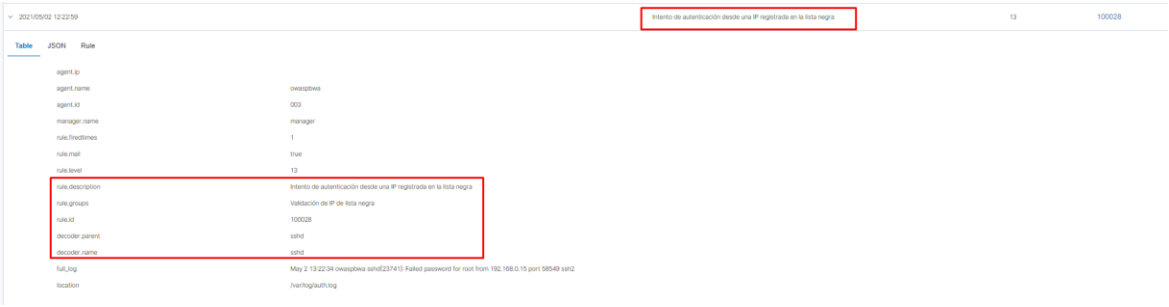

Figura 2-21 - Evidencia funcionamiento de regla creada al realizar una petición ssh desde la IP del equipo Windows 10.

2.4.3 Validación de Resultados

Las alertas fueron configuradas y expuestas por la máquina web vulnerable (*Honeypot*), se monitorea con Wazuh las peticiones entrantes y con Zabbix, la disponibilidad de los servidores expuestos durante 2 meses, donde se obtiene la siguiente información:

La Figura 2.22 expresa los eventos de seguridad recibidos en la plataforma Wazuh para el servidor OWASP.

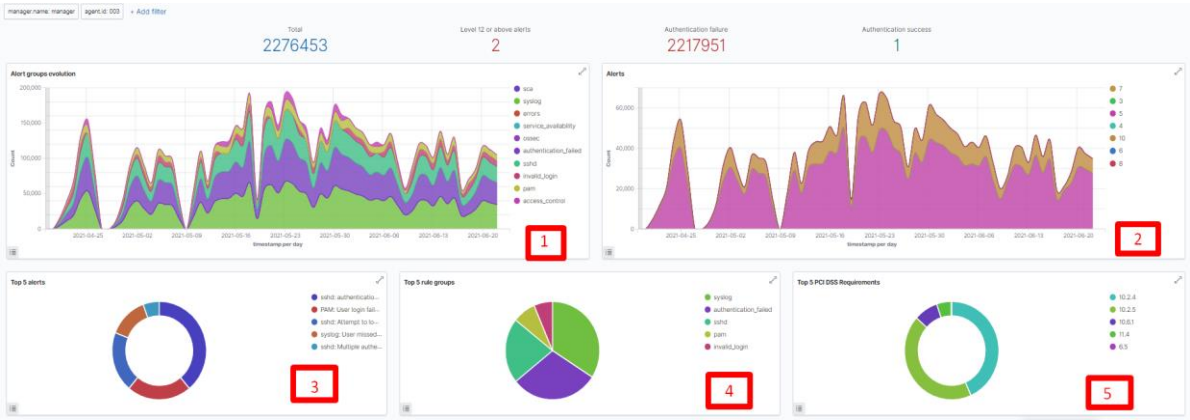

Figura 2-22. Eventos de seguridad registrados en la máquina OWASP.

En total fueron registrados 2´276.453 peticiones. En el panel 1, se encuentran los grupos a los que pertenecen las alertas detectadas, por ejemplo, las peticiones SSH, autenticaciones fallidas, usuarios inválidos, entre otras. En el panel 2, se evidencian las alertas catalogadas por nivel, según el grafico los niveles que mayor registro tuvieron son las alertas catalogadas como nivel 5 y nivel 10. En el panel 3, se demuestra el top 5 de grupos de alertas, de los cuales los tres primeros lugares indican que con un 38.03% el mayor número de peticiones activó la alerta SSH: autenticación fallida; el 22.01% de las peticiones activaron el PAM: cuentas de usuario fallidas; finalmente, el 19.93% de las peticiones corresponde a SSHD: intentos de inicio de sesión utilizando un usuario inexistente.

En el panel 4, figuran los grupos de alerta que tuvieron mayor activación y, en el panel 5, el top de requerimientos para la normativa PCI DSS que mayor impacto tuvieron entre las peticiones. Así mismo, se consulta la regla para la consulta de la lista negra de IP, URL, Hostnames y DNS obtenidos en Alien Vault, con el fin de establecer cuantas peticiones fueron recibidas con alguno de los vectores de ataque configuradas en la CDB.

Con base en lo anterior, se deduce que no hay una relación entre las listas negras y las peticiones recibidas, lo que se interpreta para la presente investigación, que se pueden utilizar indicadores de compromiso recopilados por otras empresas o entidades de seguridad, posiblemente los vectores de ataque no coincidan en la mayoría de los casos, de todos modos, configurarlos puede brindar protección ante una eventualidad.

Se logra evidenciar que los vectores de ataque cambian en el tiempo; para dar un ejemplo de ello, se tomaron las IP más recurrentes en los primeros 7 días (Ver Figura 2.23), y las más recurrentes luego de dos meses (Ver Figura 2.24). De estas dos imágenes se evidencia que las IP cambian a través del

tiempo, esto puede tener dos interpretaciones, la primera es el cambio de atacantes y la segunda, es que los atacantes cambien con frecuencia las IP de donde se generan las peticiones.

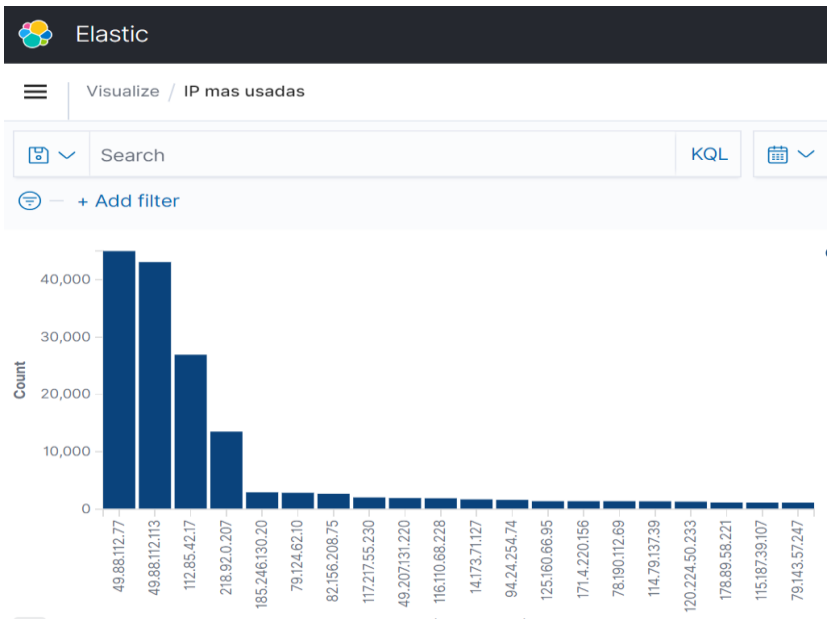

Figura 2-23 – IP's más utilizadas en las peticiones recibidas en los primeros 7 días.

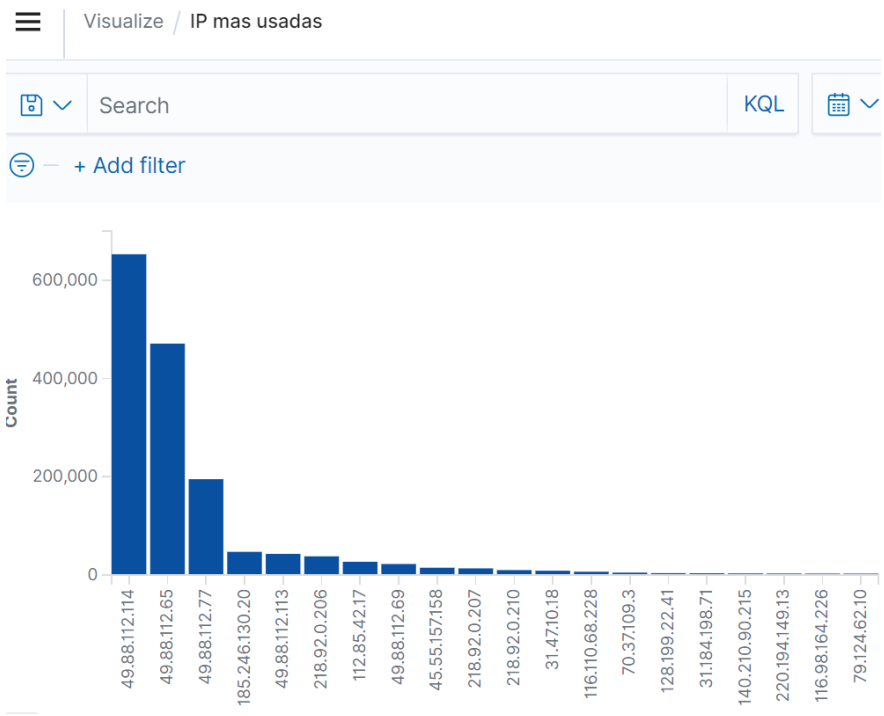

Figura 2-24 – IP's más utilizadas en las peticiones recibidas luego de dos meses.

De lo anterior, se concluye que es importante, tener en cuenta que los IOC publicados por las entidades de ciberseguridad y las actualizaciones o parches de seguridad, también se hace importante monitorear los servidores con que cuenta la red, con el fin de, tomar acciones sobre toda información relevante que se registre, por ejemplo, si ninguna de las IP anteriores hace parte de la infraestructura de la red sería importante bloquearlas usando el firewall para que dichas peticiones no afecten la funcionalidad de la red.

Se expone a continuación los IOC, más representativos, generados con base en la información recolectada en la *honeypot* de alta interacción, entre ellos: usuarios, IP y puertos más utilizadas; países que comúnmente realizan las peticiones; grupos de reglas web con mayor actividad (Access log, attack, sql injection, web).

La Figura 2.25, indica los usuarios más utilizados en las peticiones recibidas. Como recomendación, Si se da el caso, de que alguno de los usuarios utilizados corresponda a un usuario en producción, se debe cambiar por lo menos la contraseña y preferiblemente el usuario.

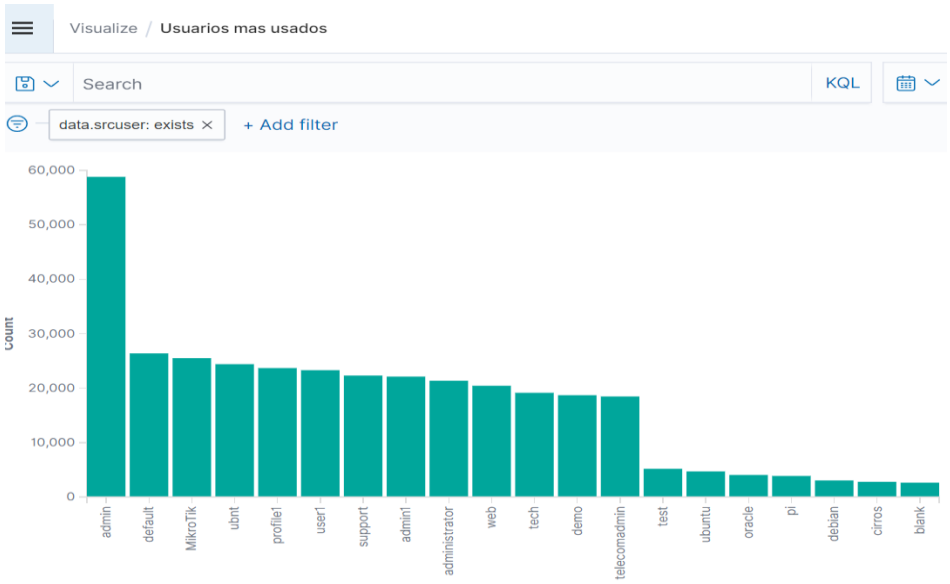

Figura 2-25 – Usuarios más utilizados en las peticiones recibidas.

En la Figura 2.26, se representan los puertos más utilizados por los atacantes. En este caso, si se identifica que alguno de los puertos corresponde a un servicio fundamental, debería ser cambiado o asegurado mediante un firewall.

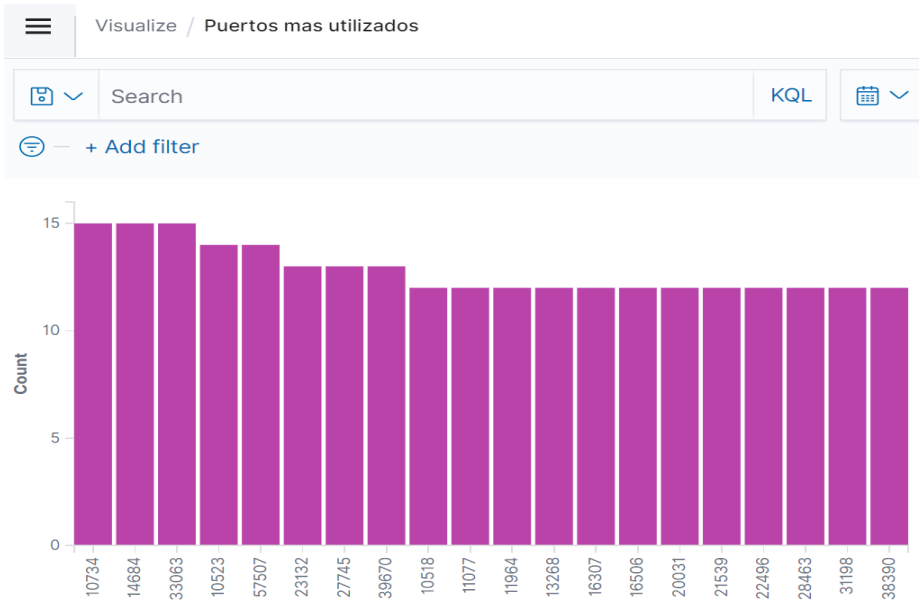

Figura 2-26 - Puertos más utilizados en las peticiones recibidas

Las reglas con mayor actividad son expuestas en la Figura 2.27, con base en esta información, se puede identificar, que, a partir de este tipo de ataques, se debe reforzar la seguridad; para este caso en especial, debería limitarse las peticiones fallidas o el número de equipos que pueden usar el servicio SSH de un servidor.

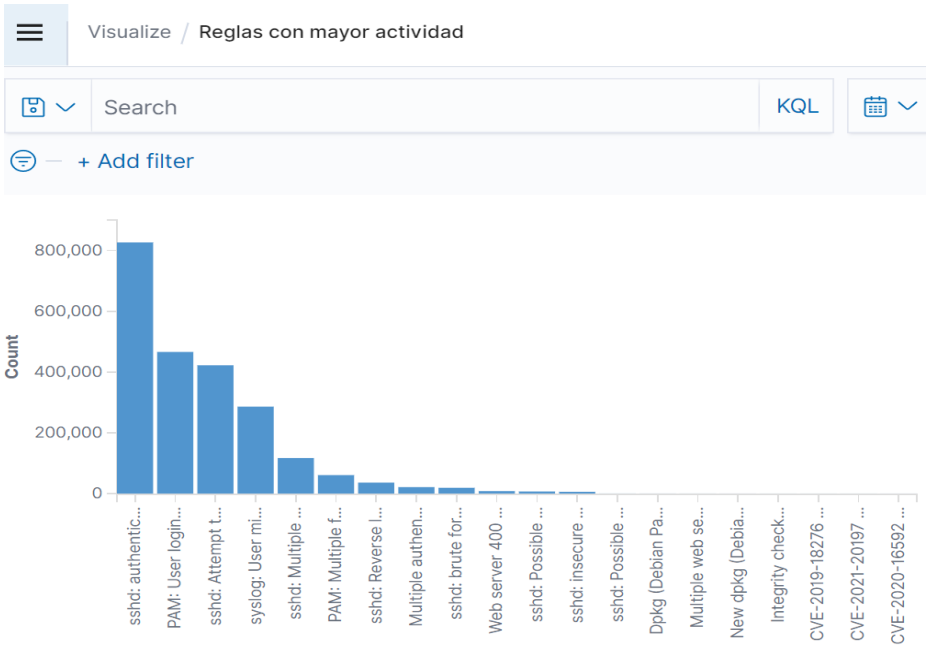

Figura 2-27 – Reglas con mayor actividad en las peticiones recibidas.

La Figura 2.28, otorga detalle de los países con mayor actividad registrada. Es posible limitar las peticiones de acuerdo con la geolocalización, Por ejemplo, si una compañía no cuenta con cliente, ni sedes en China, no debería aceptar las peticiones. Siguiendo con el ejemplo, en este caso el no aceptar peticiones de China, ahorraría a la red un poco más de 1´500.000 peticiones.

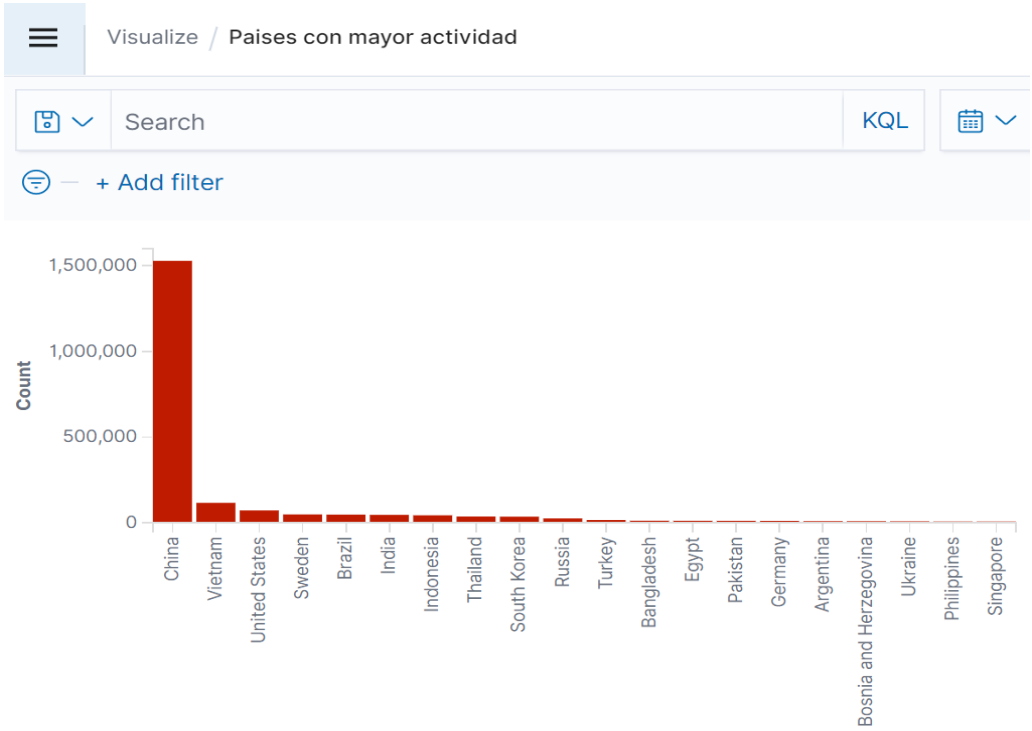

Figura 2-28 – Países con mayor actividad en las peticiones recibidas.

Los grupos de reglas web con mayor actividad expresados en la Figura 2.29, pueden ser utilizados para reconocer las técnicas de ataque y reforzar los controles e incluso implementar los controles faltantes, de acuerdo con el tipo de ataque recibido, por ejemplo, en este caso que las peticiones son de tipo web debería ser implementado un WAF para tener mayor trazabilidad de las peticiones recibidas.

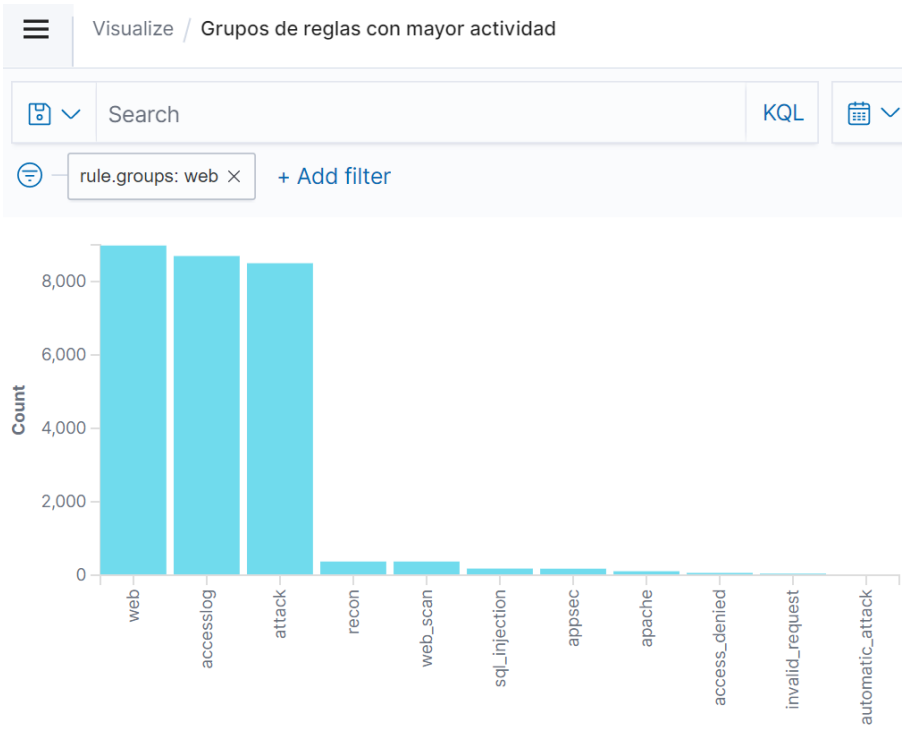

Figura 2-29 – Grupos de reglas web con mayor actividad en las peticiones recibidas.

Se logra evidenciar como con la topología propuesta es posible obtener IOC de tipo geolocalización (irregularidades geográficas), tráfico de red inusual, trafico irregular de puertos, consultas a base de datos desde IP no autorizadas, peticiones de autenticación fallida, IOC de disponibilidad (zabbix), cambios en archivos de configuración.

A nivel general, las reglas con mayor actividad son las peticiones SSH, lo cual se puede presentar dado que, si los ciberdelincuentes lograran acceder a las máquinas y servicios, tendrían acceso a todo el sistema y, por ende, a su información de bases de datos y su lógica de negocio.

Para ilustrar un ejemplo de la funcionalidad que brinda el servidor de disponibilidad Zabbix dentro de la topología propuesta, se muestra en las Figuras 2.30, 2.31, 2.32 cómo se evidencia la trazabilidad de la disponibilidad del servidor.

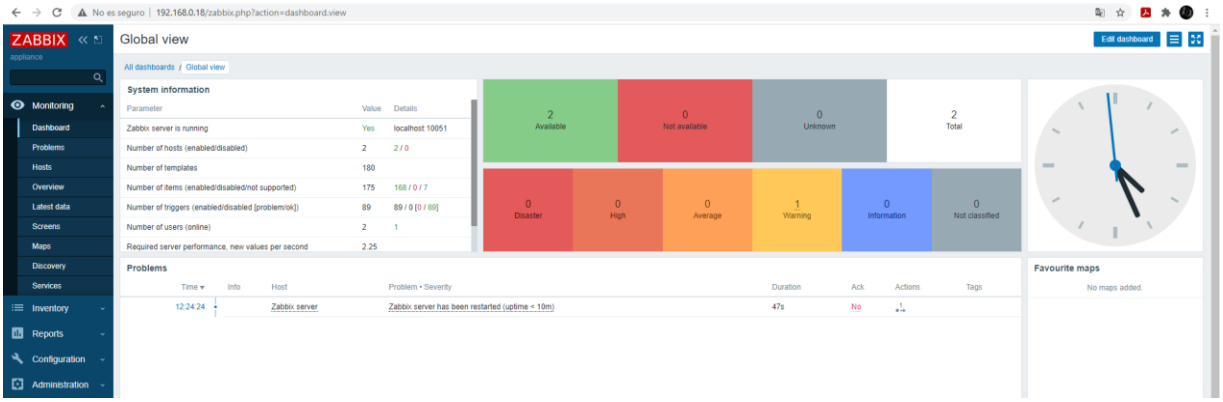

Figura 2-30 – Reporte Zabbix de reinicio en la máquina Ubuntu 20.04.

En la Figura 2.30, se puede observar que el servidor Zabbix reporta, a través de un correo electrónico, un reinicio del servidor Ubuntu, cuya duración de indisponibilidad fue de 47 segundos. Estas alertas se dan de manera proactiva sin que la persona encargada de monitoreo tenga que estar pendiente. En este mismo tablero central se encuentra la información relacionada con los *n* servidores que se encuentren configurados y estén siendo monitoreados. Es posible que sean enviadas este tipo de notificaciones de acuerdo con los picos de comportamiento de la memoria RAM, el disco duro, cambios en el nombre de máquina, entre otros.

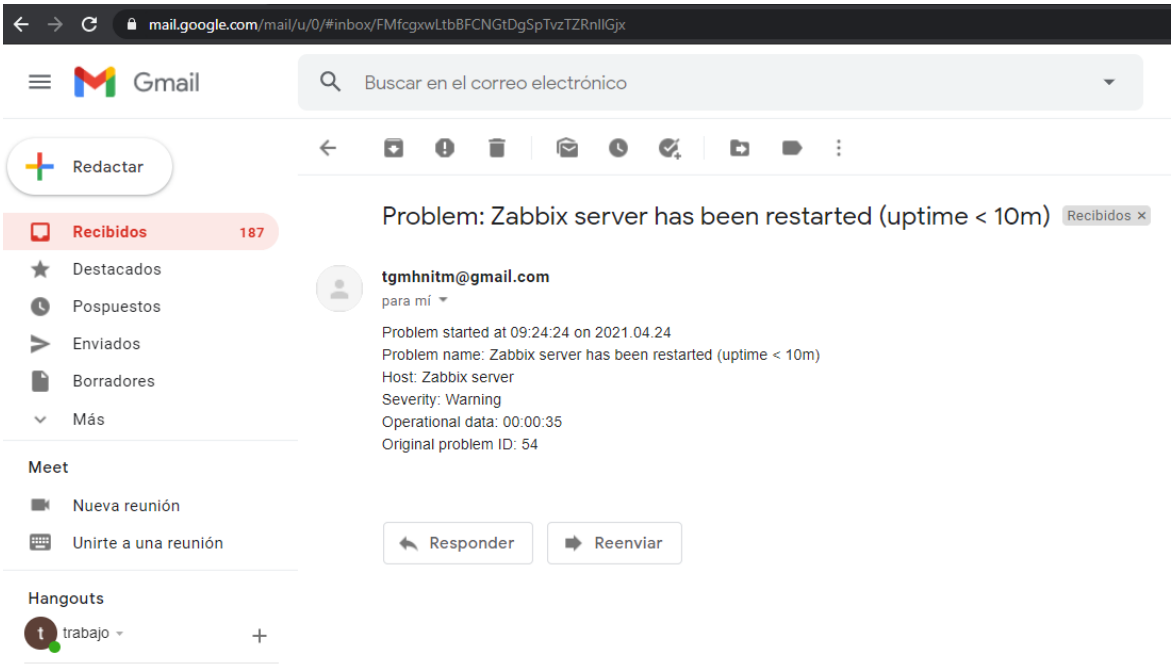

Figura 2-31 – Correo electrónico enviado por el servidor Zabbix informando el reinicio.

La Figura 2.31 es un ejemplo del correo automático que es enviado por el servidor Zabbix, indicando que el servidor Ubuntu que fue reiniciado; en este correo se reporta la alerta que indica la hora de inicio de la novedad encontrada.

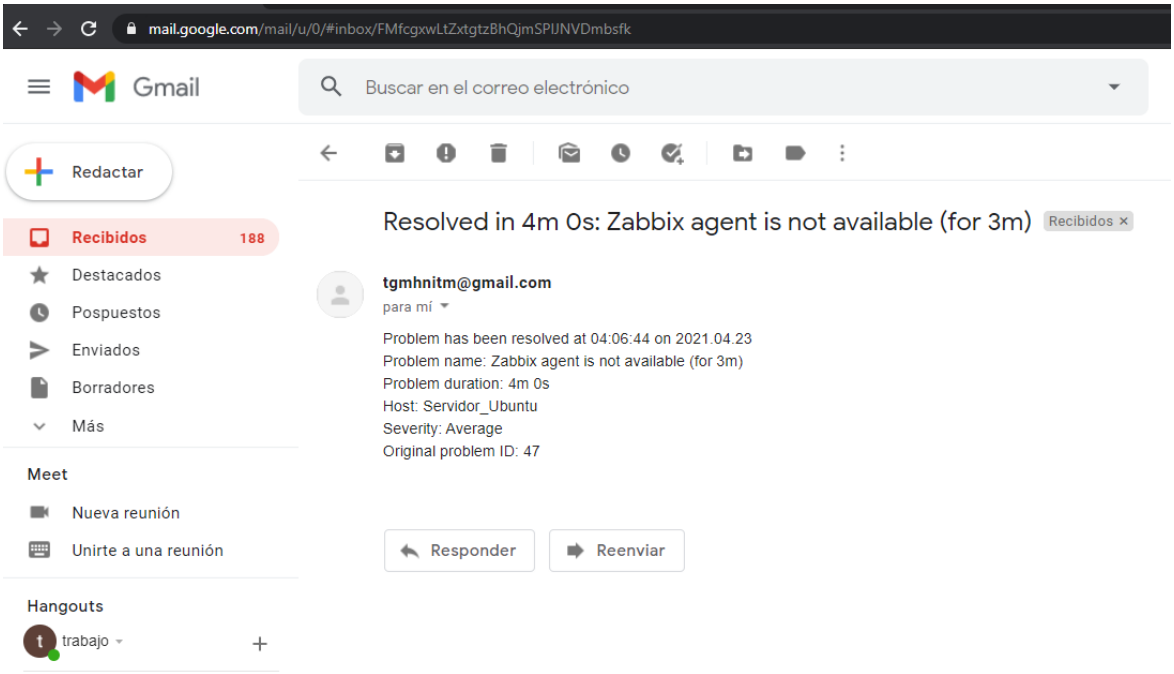

Figura **2-32** - Correo electrónico enviado por el servidor Zabbix informando el reestablecimiento.

En la Figura 2.32 el sistema notifica que ya fue reestablecido el servicio del servidor indicando la duración del evento y, la fecha y hora en que fue resuelto.

Se encontraron diferentes ventajas en el uso de la topología propuesta, con lo que es posible obtener datos tales como:

- Los benchmarking de los servidores donde se puede evidenciar que configuraciones se deben realizar.
- Las vulnerabilidades CVE obtenidas desde la NVD detectadas en los sistemas operativos.
- Tipificación según diferentes estándares de seguridad PCI DSS lo cual sirve para identificar los IOC según el mercado (banca, salud...)
- Tipos de ataque según el estándar de mittre atack donde se puede evidenciar la forma de tratar dichos ataques.
- En los ataques recibidos se pueden evidenciar los comandos realizados y una geolocalización que se puede utilizar.

- Reportes por vista lo cual fue generado diariamente.
- Creación de listas blancas y listas negras con el fin de realizar reglas que alerten, reglas que pueden contar con un nivel.
- El sistema cuenta con un módulo de active response donde pueden generarse bloqueos para los IOC encontrados tipo firewall.
- Línea base y comportamiento normal de los servidores en cuanto a disponibilidad.
- Mensajería tanto en Zabbix como en Wazuh para las alertas más relevantes para los agentes monitoreados.
- Consultas personalizadas en las peticiones recibidas.

2.4.4 Recomendaciones de Controles para la Reducción de Riesgos

Con base en los resultados establecidos por la presente investigación, se citan a continuación los controles técnicos que las empresas deberían seguir para asegurar constantemente su información.

- Investigar constantemente nuevos vectores de ataque y nuevas vulnerabilidades, mediante la consulta en páginas oficiales como Alien Vault OTX, CVE y Mittre Attack, con fuentes de información de alta calidad, ya que estos vectores de ataque son de gran importancia y tienen relevancia dentro de los controles de reducción de riesgos.
- Se recomienda bloquear el top 20 de IP, relacionadas con las peticiones malintencionadas recibidas, dichas IP se encuentran en la Figura 28, principalmente las IP 49.88.112.114; 49.88.112.65;49.88.112.77 ya que registran de 200 mil peticiones en adelante.
- En caso de contar con usuarios de nombres admin, default y MikroTik, bien sea en el sistema o en las bases de datos se recomienda sean modificados o reemplazados en lo posible, teniendo en cuenta que estos nombres de usuario registran entre 20.000 y 60.000 peticiones maliciosas.
- Dado el alto número de peticiones mal intencionadas recibidas desde el país de China, se recomienda bloquear todas las peticiones que lleguen desde dicha localidad siempre y cuando no se cuente con servidores o servicios ubicados en dicho país.
- Escanear y gestionar de manera proactiva las vulnerabilidades de la red, software y los dispositivos asociados a las funciones de la compañía, validando la solución de dichas vulnerabilidades, aplicando los correctivos y configuraciones correspondientes para mitigar las posibilidades de que exploten las vulnerabilidades, con procesos de inspección y manejo correcto de dicha proactividad para el correcto uso.

- Mantener los sistemas actualizados y licenciados según corresponda para, de esta manera, cerrar posibles brechas de seguridad.
- Configuración de puertos verificando que no correspondan al top 20 de puertos entregados en la figura 30, ni sean utilizados por defecto, configurarlos de manera que solo las máquinas permitidas los accedan y no publicarlos a internet de no ser necesario, dado que como se ha mencionado se tienen en cuenta variables como seguridad y privacidad.
- La implementación de un WAF es fundamental para evitar ataques de tipo web, para este caso se recomienda utilizar fail2band, modsecurity y cloudflare el cual permite proteger una url de manera gratuita evitando ataques de tipo DoS distribuida, XSS, SQL injection, con la intención de brindar eficiencia y eficacia en dicha implementación.
- Habilitar un IPS como pfsense para filtrar el tráfico de red, con el fin de prevenir los ataques mediante la implementación de reglas Yara, utilizando la información recolectada en los indicadores de compromiso.
- Si se utiliza la metodología propuesta se recomienda optimizar las máquinas según el módulo de benchmarking que entrega el Wazuh, esto se hace con el fin de fortalecer la seguridad de cada servidor. Si no se utiliza la topología puede usar una herramienta de benchmarking de código libre para ejecutar esta actividad.
- Utilizar un monitoreo de directorios usando herramientas como Wazuh, donde alertaría si hay cambios por parte de usuario no permitidos en una carpeta o ruta determinada mediante el módulo de integrity monitoring. Esto permitirá evidenciar qué tan permeados se encuentran los sistemas en caso de recibir un evento de seguridad.
- Instalar en los sistemas operativos software antivirus o antimalware con el fin de disminuir el riesgo de infecciones de malware.
- Utilizar técnicas como port nocking para caracterizar las peticiones normales de la red en los servidores y el resto sean denegadas.
- Realizar copias de seguridad y backup de las bases de datos e información manejada dentro de la organización. Esto con el fin de recuperar fácilmente el sistema en caso de que este sea comprometido.
- Configurar las listas negras de manera dinámica, con el fin que estas puedan ser automáticamente actualizadas y gestionadas.
- Implementar herramientas *opensource* que permitan sistemas de virtualización gratuitos como *Proxmox* el cual, adicionalmente, permite un *gateway* para el correo electrónico y un servidor de *backups.*

Dadas las anteriores actividades son sus respectivos resultados, se pudo validar el funcionamiento de los IOC, a través de la configuración de estos, exposición de las *Honeypot* a internar y la recolección de las alertas generadas por diferentes ataques informáticos, finalmente se entregan una serie de controles como recomendaciones a ser aplicadas.

2.4.5 Resultados objetivo general

De acuerdo con la ejecución de las diferentes fases y sus resultados, y conforme a la declaración del objetivo general "*Definir indicadores de compromiso que, basado en la recolección de datos de una Honeypot académica, identifiquen ataques de seguridad en servicios Web, generando recomendaciones para controlar y reducir los niveles de riesgo*", a continuación de relata su cumplimiento:

a. Se realizó una búsqueda en diferentes fuentes, de estas se obtuvieron los ataques informáticos y las 3 vulnerabilidades más relevantes, con lo cual, se definieron un grupo de 5 indicadores de compromiso asociados a los eventos de seguridad.
b. Se hizo una selección de una *Honeypot* que pudiese albergar servicios Web a ser atacados, con el fin de lograr configurar los IOC definidos.
c. Se realizó la configuración del *Honeypot*, los IOC y un sistema de monitoreo con Wazuh, que permitió, a través de la exposición de servicios Web en Internet, capturar diferentes ataques informáticos, los cuales fueron identificados a través de los IOC configurados, generando graficas del comportamiento registrado, con ello, se evaluó que el sistema es funcional en una *Honeypot* académica.
d. Finalmente, se han generado una serie de recomendaciones (controles) que pueden ser aplicados en los servicios Web, con el fin de reducir los diferentes niveles de exposición.

3. Conclusiones y Recomendaciones

3.1 Conclusiones

A partir de la realización de este proyecto, se logra definir diferentes indicadores de compromiso, obtenidos a partir de fuentes abiertas y de la recolección de datos de una *Honeypot académica*, que estuvo de forma funcional durante 2 meses, con esto, se logró identificar los ataques de seguridad en servicios Web, generando recomendaciones para controlar y reducir los niveles de exposición. Lo que permitió definir los indicadores de compromiso, bajo la recolección de datos de dicha *Honeypot,* e identificando los ataques de Seguridad Web.

Con base en, el desarrollo del estudio, se logra identificar un conjunto de vulnerabilidades y amenazas para los servicios web, a partir del análisis de los informes de ciber amenazas y tendencias, publicados periódicamente por entidades como OWASP, CCN-CERT, Incibe-Cert, MITRE, como resultado de este proceso de identificación, se reconoce que las tres vulnerabilidades más relevantes: Denegación de servicio; Inyección de código y exposición de datos sensibles. Este hallazgo, es fundamental, dado que, a partir de la identificación de estas vulnerabilidades, se pueden establecer qué tipo de ataques pueden ser generados para aprovechar dichas vulnerabilidades. Al entender los ataques que pueden ser generados, se facilita la creación de controles técnicos, lo que permite, fortalecer los mecanismos de defensa que se implementan para estos sistemas.

De igual modo, en el estudio se identificaron indicadores de compromiso existentes, clasificados en cuatro tipos, a saber: IP, URL, DNS, HOSTNAMES, además, se obtuvo un set de indicadores de compromiso basados en la recolección de datos de una *Honeypot* académica (*Honeypot* de alta interacción), los cuales son entregados en los anexos, identificando usuarios, IP, puertos y países comúnmente utilizados para la realización de ataques a servicios Web. Este proceso posibilitó la generación de recomendaciones técnicas para controlar y reducir los niveles de exposición. Cabe resaltar, que este tipo de procesos, tienen un nivel de complejidad alto y se requiere tener manejo en dicha área para no agravar situaciones.

De acuerdo al estudio realizado, las diferentes soluciones *Honeypot* que normalmente se encuentran en el mercado, son diseñadas con un objetivo previamente definido, el cual puede ser estudiar un tipo de protocolo, un tipo de ataque, comportamientos de red referente a un puerto o una configuración de seguridad específica, aun así, fue posible evidenciar que, usar una solución

independiente puede otorgar mayor libertad y mayores resultados en la trazabilidad y en la obtención de los vectores de ataque.

El proceso evaluativo, se llevó a cabo en los diferentes IOC, que fueron definidos, implementando reglas que identificaban si una petición contenía dentro de sus vectores de ataque alguna de las IP, URL, *Hostnames* y DNS obtenidos en *Alien Vault*, se logra probar que dentro de las más de dos millones de peticiones recibidas no se encontraron los IOC previamente definidos, lo cual describe claramente que, más que configurar IOC definidos por entidades de seguridad, se hace importarte el monitoreo constante de los servidores y la red. Según el análisis de los resultados obtenidos, fue posible generar recomendaciones para la reducción de los riesgos, recomendaciones que se encuentran en el numeral 2.4.3 de la metodología y resultados. Adicionalmente, concluye que, para obtener un efecto positivo en los sistemas de información y definir controles eficientes, para contrarrestar los ataques es pertinente su configuración en el software apropiado, por ejemplo: las IP, identificadas como maliciosas deben configurarse en un firewall, los vectores de ataque de tipo http deben ser configurados en un Web *Application Firewall* y así mismo los hash y herramientas utilizadas por los atacantes, deben ser restringidos en un proxy.

Esta investigación, tiene gran importancia en materia de estudio para los profesionales cuyo perfil corresponda a seguridad informática, puesto que, permite conocer este tema, ampliando sus conocimientos y aportando claridad a los aprendizajes adquiridos por el lector en su proceso académico.

3.2 Recomendaciones

Se sugiere que, en las infraestructuras que utilicen servicios web, se contemplen y se corrijan las tres vulnerabilidades reconocidas por MITRE, OWASP, Incibe-Cert y CCN-CERT. También, se debería tener en cuenta, que el manejo de cada topología debe ser único, por lo tanto, cada entidad, empresa u organización, debe tener una gestión de vulnerabilidad, basado en los servicios o la información que manejen.

Aún cuando el gremio de la seguridad informática, ofrece diferentes Indicadores de Compromiso, se sugiere el uso de una Honeypot, para establecer IOC. Con base en estos IOC, el profesional en seguridad identifique los controles técnicos, verdaderamente eficientes para el manejo de sus incidentes.

Teniendo en cuenta los resultados obtenidos en la presente investigación, para futuros trabajos, se recomienda implementar la topología propuesta dentro de un ambiente productivo de una Institución Académica o una compañía, con el fin de, identificar su funcionalidad en un ambiente industrial, relacionando las vulnerabilidades más relevantes encontradas y partiendo desde las recomendaciones de controles y los IOC que fueron definidos.

Las principales recomendaciones referentes al fortalecimiento de la seguridad de los sistemas perimetrales, es gestionar los controles técnicos considerados como necesarios para desarrollar e implementar los IOC.

4. Glosario

Agente de amenaza

También conocido como ciber atacante es una persona o grupo de personas que se encuentran de manera constante en búsqueda de generar incidentes de seguridad, utilizando tácticas, técnicas y herramientas que permitan tanto identificar como aprovechar las vulnerabilidades de los sistemas objetivo, ya sea para lucrarse o con el fin de comprometer y hacer daños a sus víctimas [93].

Amenaza

Corresponde a cualquier evento que, por medio de técnicas básicas o sofisticadas de ciberataques, ocasione un efecto negativo en los bienes, el personal o en el operar de las organizaciones (incluyendo su prestigio, finalidad, tarea u objetivo corporativo), perjudicando los pilares de la estabilidad de la información [94].

Backdoors

Es un tipo de malware utilizado en los ataques persistentes el cual permite tener acceso a un dispositivo sin que el propietario o usuario deba autorizar la conexión, logrando usar la información y las herramientas para fines determinados. Esta técnica es utilizada desde 1997 y a la fecha se evidencia la instalación de puertas traseras [6].

Backup

Es una técnica que busca evitar la indisponibilidad, perdida, destrucción, modificación de archivos dentro de un dispositivo electrónico, en donde básicamente se hace una copia de la información original, permitiendo adicionalmente una pronta recuperación de desastres y una recuperación de los datos en caso de ser necesario [95].

Benchmarking

Metodología utilizada para determinar que los dispositivos y software se encuentran correctamente configurados, según los lineamientos básicos de seguridad definidos por un fabricante o configuraciones encontradas de CVE o CWE para evitar vulnerabilidades en los dispositivos [96].

Criptografía

Se emplea para el intercambio de datos de manera segura, de forma que puedan ser comprendidos por entidades a quienes van dirigidos y poseen los medios para descifrarlo, con el fin que los datos originales sean ilegibles para quien desconozca la forma en que fue cifrada la información. Con esta técnica se apoya el pilar de la seguridad confidencialidad [97].

CVE (*Common Vulnerabilities and Exposures*)

En español exposiciones y vulnerabilidades comunes, es un repositorio donde son compartidas públicamente las vulnerabilidades o exposiciones conocidas. [98].

Desde MITRE se realizan diferentes muestreos de datos según las vulnerabilidades reportadas y documentadas por las entidades y empresas de seguridad, las cuales presentan una candidatura y atraviesan por tres etapas para la publicación de la vulnerabilidad. El primer paso es el registro tratamiento donde se estudia, luego se procede con la asignación CVE-ID que es el identificador de la vulnerabilidad y por último está la etapa de publicación. En este marco de trabajo se organiza la información, se caracteriza y surgen diferentes informes y estadísticas [98].

CWE (*Common Weakness Enumeration*)

En español enumeración de debilidades habituales, es una lista generalizada de debilidades en programa y hardware donde no se tiene presente ningún producto o abastecedor en concreto, sino que agrupa o categoriza las vulnerabilidades concretas CVE´s [99].

Defacement

Ataque dirigido a los sitios web para modificar total o parcialmente la apariencia de los sitios, se usa alguna vulnerabilidad para realizar los cambios, bien sea para ejecutar un phishing, una suplantación o para capturar información relevante de un usuario o un grupo de interés determinado [100].

Firewall

También llamado cortafuegos es un dispositivo de seguridad perimetral el cual puede ser físico (dispositivo) o lógico (software) que permite el filtrado del tráfico entre dos redes, normalmente es donde se definen las reglas y políticas de seguridad de las compañías estableciendo cuales sitios en internet pueden o no ser visitados, cuales puertos escuchados o consumidos y/o realizar bloqueos de IP malintencionadas [101].

Gateway

También llamado puerta de enlace es el dispositivo encargado de la comunicación bidireccional o unidireccional entre dos redes. En caso de que las tecnologías de comunicación no compaginen se encarga de estandarizarlas con el fin que exista una conexión efectiva [102].

Hardening

Procedimiento mediante el cual se busca minimizar la posibilidad de que sea aprovechada una o más vulnerabilidades, en la que básicamente se desactivan los servicios que no son necesarios dentro de los servidores y dispositivos de seguridad perimetral, adicionalmente, se gestiona el parcheado y la actualización de estos [103].

IDS

Traducido al español, sistema de detección de intrusos, es un software que se utiliza para identificar posibles ataques de seguridad hacia los dispositivos de la red, mediante el tráfico de red establece actividades anómalas o sospechosas y se basa en un set de reglas previamente configuradas, donde pueden ser utilizados marcos como las reglas YARA para la protección de la red [97].

IPS

Conocido como sistema de prevención de intrusos, tiene integradas algunas funciones de firewall y de IDS con el fin de tomar acciones de prevención sobre posibles anomalías o ataques de seguridad detectadas dentro de una red [97].

Iptables

Se utiliza para filtrar el tráfico de red que permite una maquina con sistema operativo Linux, tiene la capacidad de agregar, eliminar o modificar paquetes de red, según los filtros que se requieran. Tiene compatibilidad con el protocolo IPv4 y IPv6 y está disponible en Linux a partir de la distribución 2.4 en adelante para el uso de personal administrador o usuario final [104].

Log

Evidencia resultante de todas las transacciones operadas en una aplicación determinada, dentro de seguridad informática es un registro de los eventos que ocurren dentro de los sistemas perimetrales o la red de una organización. Con el análisis de estos registros es posible determinar la interacción que hubo entre sistemas, individuos y/o la red [105].

Malware

Todo firmware o software mal intencionado diseñado con el fin de someter, dañar o cambiar la morfología original de los datos dentro de un celular, computador, servidor o dispositivo electrónico. Algunos de los tipos más conocidos son backdoor, gusanos, troyanos, spyware [106].

NVD (*National Vulnerability Database*)

Esta información puntúa las vulnerabilidades teniendo presente el efecto que tienen la posibilidad de producir al ser explotadas, e incluye listas de verificación de estabilidad, debilidades de estabilidad en el programa y configuraciones no idóneas tanto en hardware como en programa [107].

OpenSource

Código abierto en el idioma español, hace referencia a la característica de entregar el código de manera que cualquier persona tenga la facultad de verlo, editarlo y distribuirlo sin restricción alguna. Esto ha creado a través de los tiempos todo un movimiento colaborativo, donde se comparten las soluciones útiles para un gremio o funcionalidad especifica [108].

OSSEC

Sistema de detección de intrusos, que monitorea una maquina final, sea virtual, física o servidor identificando posibles ataques informáticos hardware, firmas y malware. Normalmente OpenSource y capaz de proyectar en un cuadro de mando o gestión la información de los Logs encontrados [109].

Pentesting

Practica que se utiliza para encontrar las vulnerabilidades de un software o hardware, la cual consiste en simular un ataque en busca de configuraciones deficientes, determinar la viabilidad de un ataque y proponer en base al resultado acciones técnicas que mitiguen la explotación de las vulnerabilidades encontradas [110].

Proxy

Software encargado de filtrar las peticiones entre la red LAN e internet de manera bidireccional, en este dispositivo es posible configurar reglas que impiden accesos no autorizados desde internet hacia la red privada o el acceso de los usuarios de la red privada a páginas web no deseadas [111].

Ransomware

Es un tipo de malware o software malicioso, que se distribuye mediante páginas web, correos, dispositivos de almacenamiento extraíble e instalación de software. Su objetivo principal es cifrar la información de la víctima, utilizando el cifrado asimétrico, el cual requiere dos claves para descifrar los datos, una clave se encuentra en la maquina y la otra solo es conocida por el atacante. Una vez cifrada la información el atacante contacta a la víctima, relacionando el valor que se espera a cambio de la clave faltante [112].

Spoofing

Ataque basado en la suplantación, donde el atacante busca reemplazar el DNS, página web, email y/o IP por uno malicioso, con el fin de inyectar código, instalar un backdoor, o infectar un dispositivo con malware para un fin determinado [97].

VPN

Por sus siglas, Virtual Private Network, es una comunicación cifrada entre dos redes LAN lograda mediante una red pública o internet. Usualmente utilizada por las compañías para que sus colaboradores tengan acceso a red corporativa sin comprometer la misma [97].

Vulnerabilidad

Corresponde a los puntos de quiebre, falta de configuración o aspectos no cubiertos a nivel de seguridad, los cuales pueden ser aprovechados por un ciber atacante para un fin específico. A nivel de seguridad de la información las vulnerabilidades se encuentran indexadas mediante las bases de datos CVE, CWE, NVD, las cuales básicamente describen cada vulnerabilidad y/o familia de vulnerabilidades, otorgándoles un identificador, una descripción, una posible solución en caso de que haya sido identificada y por su puesto las referencias de donde se encuentra la información con mayor amplitud [113].

5. Anexos

Anexo 1 - Alien Vault IOC.ioc

Anexo 2 - Alien Vault IOC IP.ioc

Anexo 3 - Alien Vault IOC URL.ioc

Anexo 4 – Alien Vault IOC DNS.ioc

Anexo 5 – Alien Vault IOC HOSTNAME.ioc

Anexo 6 – Manual de instalación Honeypot

Anexo 7 – Honeypot IOC.ioc

Anexo 8 – Honeypot IOC usuarios.ioc

Anexo 9 – Honeypot IOC IP.ioc

Anexo 10 – Honeypot IOC Puertos.ioc

Anexo 11 – Honeypot IOC Paises.ioc

6. Bibliografía

[1] F. B. Jurado, «Diseño de una arquitectura de integración de servicios Web,» 2019. [En línea]. Available: https://ruidera.uclm.es/xmlui/bitstream/handle/10578/23091/TFG_FIDELBueno.pdf?sequence=1&isAllowed=y.

[2] BBC, «"12 ataques por segundo": cuáles son los países de América Latina más amenazados por "malware",» 6 Septiembre 2016. [En línea]. Available: http://www.bbc.com/mundo/noticias-37286420.

[3] CCN-CERT, «CIBERAMENZAS Y TENDENCIAS,» 2017. [En línea]. Available: https://www.ccn-cert.cni.es/informes/informes-ccn-cert-publicos/2221-ccn-cert-ia-16-17-ciberamenzas-y-tendencias-edicion-2017-resumen-ejecutivo-1/file.html.

[4] ACIS: ASOCIACIÓN COLOMBIANA DE INGENIEROS DE SISTEMAS, «Profesionales En Seguridad De La Información:¿ Evolución O Revolución?,» 14 Noviembre 2019. [En línea]. Available: https://sistemas.acis.org.co/index.php/sistemas/issue/view/7.

[5] M. Fraunholz, H. Zimmermann y D. Schotten, «An adaptive honeypot configuration, deployment and maintenance strategy,» de *2017 19th International Conference on Advanced Communication Technology (ICACT)*, Bongpyeong, 2017.

[6] M. L. G. M. A. Ávila, «Metodología para la identificación de indicadores de compromiso para la protección de infraestructuras críticas,» Bliblioteca Digital Universidad de Alcalá, Abril 2018. [En línea]. Available: https://ebuah.uah.es/dspace/bitstream/handle/10017/33004/TFM_Avila_Granada_2018.pdf?sequence=1.

[7] CCN-CERT, «GUÍA DE SEGURIDAD (CCN-STIC-423) INDICADORES DE COMPROMISO (IOC),» Octubre 2015. [En línea]. Available: https://www.ccn-

cert.cni.es/series-ccn-stic/guias-de-acceso-publico-ccn-stic/1090-ccn-stic-423-indicadores-de-compromiso/file.html.

[8] OASIS-OPEN, «OASIS Advances Automated Cyber Threat Intelligence Sharing with STIX, TAXII, CybOX,» 16 Julio 2015. [En línea]. Available: https://www.oasis-open.org/news/pr/oasis-advances-automated-cyber-threat-intelligence-sharing-with-stix-taxii-cybox.

[9] CYWARE, «Cyber Threat Intelligence: What is CybOX? How do you use a CybOX object?,» 7 Mayo 2019. [En línea]. Available: https://cyware.com/educational-guides/cyber-threat-intelligence/what-is-cybox-how-do-you-use-a-cybox-object-af90.

[10] MITRE, «MAEC: Malware Attribute Enumeration and Characterization,» 2019. [En línea]. Available: https://maecproject.github.io/about-maec/.

[11] CSIRT, «CIF: Collective Intelligent Framework,» 1 Junio 2016. [En línea]. Available: https://www.csirt.gob.cl/.

[12] AT&T CYBERSECURITY, «AT&T Alien Labs Open Threat Exchange,» 1 Enero 2020. [En línea]. Available: https://cybersecurity.att.com/open-threat-exchange.

[13] VERIZON, «VERIS FRAMEWORK: the vocabulary for event recording and incident sharing,» 28 Octubre 2019. [En línea]. Available: http://veriscommunity.net/.

[14] UNIVERSIDAD DE ALICANTE, «Proyecto de aplicaciones Web,» 2015. [En línea]. Available: http://expertojava.ua.es/experto/restringido/2015-16/proyint/proyint.pdf.

[15] IBM, «Servicios Web aspectos basicos,» 9 Febrero 2015. [En línea]. Available: https://www.ibm.com/developerworks/ssa/library/ws-restful/ws-restful-pdf.pdf.

[16] A. Männikkö, «WS-* Web Services and Their Suitability for Modern,» University Of OULU, Enero 2020. [En línea]. Available: http://jultika.oulu.fi/files/nbnfioulu-202002181158.pdf.

[17] CRYPTOMEX, «Desarrollo de aplicaciones web distribuidas,» 22 Julio 2017. [En línea]. Available: http://cryptomex.org/SlidesAplicsDist/XML-RPC.pdf.

[18] IDECA, «Guía práctica para la creación de servicios Web bajo los principios de interoperabilidad Web segura,» 19 Febrero 2016. [En línea]. Available: https://ideca.gov.co/sites/default/files/documentacion/instructivoguia-practica-creacion-servicios-web.pdf.

[19] CINVESTAV, «Servicios Web:,» Laboratorio de Tecnologías de Información, 10 Marzo 2015. [En línea]. Available: https://www.tamps.cinvestav.mx/~vjsosa/clases/sd/ServiciosWeb_Axis_REST.pdf.

[20] P. K. a. K. P. Kour, «SQL injection: Study and augmentation,» *2015 International Conference on Signal Processing, Computing and Control (ISPCC)*, pp. 102-107, 2016.

[21] J. M. J. B. K. Anton, «Proactive controls for developers,» OWASP, 2018. [En línea]. Available: https://github.com/OWASP/www-project-proactive-controls/raw/master/v3/OWASP_Top_10_Proactive_Controls_V3.pdf.

[22] O. Mahjoubi, «Detección de vulnerabilidades y generación de alertas de seguridad para aplicaciones web,» *Universitat Oberta de Catalunya (UOC)*, 2019.

[23] M. E. G. J. L. F. P. A. Grassi, «Digital Identity Guidelines,» NIST, June 2017. [En línea]. Available: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-63-3.pdf.

[24] OWASP, «A3:2017-Sensitive Data Exposure,» 2017.

[25] MITRE, «Exposure of private personal information to an unauthorized actor,» 20 Febrero 2017. [En línea]. Available: https://cwe.mitre.org/data/definitions/359.html.

[26] OWASP, «A4:2017-XML External Entities (XXE),» *OWASP Top Ten 2017*, 2017.

[27] CWE, «Improper Restriction of XML External Entity Reference,» 2016. [En línea]. Available: https://cwe.mitre.org/data/definitions/611.html.

[28] OWASP, «A5:2017-Broken Access Control,» *OWASP Top Ten 2017*, 2017.

[29] CWE, «Improper Access Control,» 2015. [En línea]. Available: https://cwe.mitre.org/data/definitions/284.html.

[30] OWASP, «A6:2017-Security Misconfiguration,» *OWASP Top Ten 2017*, 2017.

[31] NIST, «Guide to General Server Security - Guidelines for Checklist Users and Developers,» Febrero 2018. [En línea]. Available: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-70r4.pdf.

[32] D. F. D. E. M. N. R. A. P. W. E. Burr, «Electronic Authentication Guideline,» NIST, June 2017. [En línea]. Available: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-63-2.pdf.

[33] OWASP, «A7:2017-Cross-Site Scripting (XSS),» *OWASP Top Ten 2017*, 2017.

[34] CWE, «Deserialization of Untrusted Data,» CWE, 2016. [En línea]. Available: https://cwe.mitre.org/data/definitions/502.html.

[35] OWASP, «A8:2017-Insecure Deserialization,» *OWASP Top Ten 2017*, 2017.

[36] CVE, «MITRE Common Vulnerabilities and Exposures (CVE) Search,» Octubre 2019. [En línea]. Available: https://www.cvedetails.com/version-search.php. [Último acceso: Noviembre 2019].

[37] OWASP, «A10:2017-Insufficient Logging & Monitoring,» *OWASP Top Ten 2017*, 2017.

[38] OWASP, «Application Security Verification Standard,» Marzo 2019. [En línea]. Available: https://github.com/OWASP/ASVS/raw/master/4.0/OWASP%20Application%20Security%20Verification%20Standard%204.0-en.pdf.

[39] MITRE, «CWE CATEGORY: Cryptographic Issues,» CWE, 20 Agosto 2020. [En línea]. Available: https://cwe.mitre.org/data/definitions/310.html.

[40] Y. P. D. I. R. M. C. D. G. P. P. B. C.M. Legón Pérez, «Ataques "Side Channels": Una amenaza real a las implementaciones de algoritmos criptográficos,» de *Mathematical Models for Security*, Havana, 2016.

[41] MITRE, «CWE-639: Authorization Bypass Through User-Controlled Key,» 25 Junio 2020. [En línea]. Available: https://cwe.mitre.org/data/definitions/639.html.

[42] OWASP, «Los diez riesgoa más críticos en Aplicaciones Web,» CreativeCommons, California, 2017.

[43] C. M. V. Punitha, «Detection of Coercive Parsing Attack in XML Requests using Machine Learning Techniques,» 3 Septiembre 2018. [En línea]. Available: http://isyou.info/inpra/papers/inpra-v6n3-02.pdf. [Último acceso: 3 Noviembre 2019].

[44] R. S. A. Alasri, «International Journal of Engineering & Technology,» Septiembre 2018. [En línea]. Available: https://www.sciencepubco.com/index.php/ijet/article/view/20570.

[45] S. Jan, «Automated and Effective Security Testing for XML-based Vulnerabilities,» 31 Agosto 2017. [En línea]. Available: https://pdfs.semanticscholar.org/aa5d/758e4fc3b8f9692743ea7cf88c74c9e82375.pdf.

[46] A. Dhanapal y P. Nithyanandam, «An effective mechanism to regenerate HTTP flooding DDoS attack using real time data set,» IEEE, Julio 2017. [En línea]. Available: https://ieeexplore.ieee.org/abstract/document/8342626.

[47] M. A. a. H. R. S. Baraka, «An Ontology-Based Approach for Detecting SOAP Message Attacks,» *2018 International Journal on Web Service Computing (IJWSC)*, vol. 9, nº 3/4, pp. 5-10, 2018.

[48] P. V. B. B. V. V. A. M. Y. S. Chakaravarthi, «WEB SERVICE REGISTRATION AND ROUTING SYSTEM AND INTER WEB PROXY SERVICE MODEL PREVENTS THE MESSAGE ALTERATION ATTACKS, MAN-IN-THE MIDDLE ATTACKS,» de *2017 International Conference on Information Communication and Embedded Systems (ICICES)*, Chennai, India, 2017.

[49] D. C. A. S. S. Nath, «Web service performance enhancement for portable devices modifying SOAP security principle,» de *20th International Conference of Computer and Information Technology (ICCIT)*, Dhaka, 2017.

[50] A. M. A. S. M. Babamir, «A cryptography approach on security layer of web service,» de *IEEE 10th International Conference on Application of Information and Communication Technologies (AICT)*, Baku, 2016.

[51] A. S. a. L. S. Scott, «Web Services Policy Generation Based on SLA Requirements,» de *IEEE 3rd International Conference on Collaboration and Internet Computing (CIC)*, California, USA, 2017.

[52] G. W. A. I. E. Fray, «New XML Signature Scheme That is Resistant to Some Attacks,» *in IEEE Access*, vol. 8, nº 1, pp. 35815-35831, 28 Enero 2020.

[53] S. C. A. R. K. L. K. S. Stricot-Tarboton, «Taxonomy of Man-in-the-Middle Attacks on HTTPS,» de *IEEE Trustcom/BigDataSE/ISPA*, Hamilton, New Zealand, 2016.

[54] P. N. A. D. P. B. Jagruti, «A Survey on Webservice Security Techniques,» de *4th International Conference on Computing Communication and Automation (ICCCA)*, Greater Noida, India, 2018.

[55] M. I. B. a. M. F. Hassan, «Adaptive security architecture for protecting RESTful web services in enterprise computing environment,» 27 Noviembre 2017. [En línea]. Available: https://doi.org/10.1007/s11761-017-0221-1.

[56] A. P. A. A. A. L. B. S. Jan, «Automatic Generation of Tests to Exploit XML Injection Vulnerabilities in Web Applications,» *IEEE Transactions on Software Engineering*, vol. 5, nº 4, pp. 335-362, 4 Abril 2019.

[57] S. D. A. P. K.Pranathi, «Attacks on Web Application Caused by Cross Site Scripting,» de *Second International Conference on Electronics, Communication and Aerospace Technology (ICECA)*, Andhra Pradesh, India , 2018 .

[58] V. R. M. A. K. Jevitha, «Web Services Attacks and Security- A Systematic Literature Review,» de *6th International Conference On Advances In Computing & Communications, ICACC*, Cochin, India, 2016.

[59] Kaspersky, «Tendencias de ciberseguridad ¿Qué es un honeypot?,» 2020. [En línea]. Available: https://latam.kaspersky.com/resource-center/threats/what-is-a-honeypot.

[60] INCIBE-CERT, «Honeypot, una herramienta para conocer al enemigo,» 14 Junio 2018. [En línea]. Available: https://www.incibe-cert.es/blog/honeypot-herramienta-conocer-al-enemigo.

[61] TRAP X SECURITY, «TRAP X SECURITY,» 8 Mayo 2020. [En línea]. Available: https://trapx.com/.

[62] CyberDB, «Deception Technology Facts and Emerging Vendors,» 27 Septiembre 2016. [En línea]. Available: http://cyberdb.co/wp-content/uploads/pdf/DeceptionTech%20white%20paper%20v4.pdf. [Último acceso: 5 Enero 2020].

[63] WatchGuard , «WatchGuard Technologies Acquires Percipient Networks and Adds Security at the DNS Layer to Company’s SMB Security Platform,» 17 Enero 2018. [En línea]. Available: https://www.watchguard.com/wgrd-about/press-releases/watchguard-technologies-acquires-percipient-networks.

[64] LogRhythm, «LogRhythm Security Intelligent Platform,» Febrero 2020. [En línea]. Available: https://logrhythm.com/.

[65] Dionaea, «Dionaea's documentation,» 1 Mayo 2017. [En línea]. Available: https://dionaea.readthedocs.io/en/latest/index.html. [Último acceso: 8 Mayo 2020].

[66] HARDSOFT SECURITY, «Explicación de Honeypot e instalación de kippo y kippo-graph.,» 25 Junio 2018. [En línea]. Available: https://hardsoftsecurity.es/index.php/2018/06/25/explicacion-de-honeypot-e-instalacion-de-kippo-y-kippo-graph/.

[67] B. Patel ANd H. Ramadoss, «DejaVU - Open Source Deception Framework,» 9 Mayo 2020. [En línea]. Available: https://github.com/bhdresh/Dejavu/blob/master/README.md. [Último acceso: 28 Mayo 2020].

[68] F. Y. Liow, «Development of an intrusion detection system and the deployment of a honey net,» Nanyang Technological University, 28 Mayo 2020. [En línea]. Available: https://hdl.handle.net/10356/140327.

[69] A. F. N. H. a. R. B. B. H. Wafi, «Implementation of a modern security systems honeypot Honey Network on wireless networks,» de *International Young Engineers Forum (YEF-ECE)*, Jakarta, Indonesia , 2017.

[70] M. Leguizamón, M. Bonilla y C. León, «Análisis de ataques informáticos mediante Honeypots en la Universidad Distrital Francisco José de Caldas,» 2020. [En línea]. Available: https://doi.org/10.25100/iyc.v22i2.8483.

[71] L. Spitzner, «Honeytokens: The other honeypot.,» 2020. [En línea]. Available: https://community.broadcom.com/symantecenterprise/communities/community-home/librarydocuments/viewdocument?DocumentKey=74450cf5-2f11-48c5-8d92-4687f5978988&CommunityKey=1ecf5f55-9545-44d6-b0f4-4e4a7f5f5e68&tab=librarydocuments.

[72] A. N. G. Fernandez, «Configuración de honeypots adaptativos para análisis de malware,» de *III Jornadas Nacionales de Investigación en Ciberseguridad*, 2017.

[73] D. D. E. Quijije, «Diseño del Prototipo de una Honeypot Virtual que Permita Mejorar el Esquema de Seguridad en Las redes de la Carrera de Ingeniería en sistemas Computacionales y Networking de la Universidad de Guayaquil.,» Septiembre 2011. [En línea]. Available: http://repositorio.ug.edu.ec/handle/redug/6776.

[74] J. C. T. R. J. S. Guisao, «Detección y mitigación de vulnerabilidades día cero,» Tecnólogico de Anquioquia, Octubre 2014. [En línea]. Available: http://ojs.tdea.edu.co/index.php/cuadernoactiva/article/download/201/218/.

[75] D. M. Trujillano, «Sistema adaptativo de prevención de intrusos mediante Honeypots,» UAM. Departamento de Ingeniería Informática, Julio 2016. [En línea]. Available: https://repositorio.uam.es/handle/10486/676764.

[76] C. E. Moral, «Estudio y análisis de la implantación de un honeypot en una plataforma portátil para informática forense (RASPOT),» UAM. Departamento de Ingeniería Informática, Julio 2014. [En línea]. Available: https://repositorio.uam.es/handle/10486/662509.

[77] C. T. S. A. F. P. E. E. Casanovas, «HoneyPots Web como Herramientas de Análisis de Ciberataques sobre una Red de Telefonía Móvil,» UNDEF. Universidad de la Defensa Nacional, 2017. [En línea]. Available: https://rdu.iua.edu.ar/handle/123456789/2111.

[78] E. L. T. H. F. Quinkert, «DorkPot: A Honeypot-based Analysis of Google Dorks,» Ruhr-University Bochum, 26 Febrero 2019. [En línea]. Available: https://www.syssec.ruhr-uni-bochum.de/media/emma/veroeffentlichungen/2019/02/26/DorkPot-MADWeb2019.pdf.

[79] T. C. A. Bernal, «Metodología de la investigación: para la administración, economía, humanidades y ciencias sociales.,» *Pearson Educación*, 2006.

[80] «The ultimate security vulnerability data source,» [En línea]. Available: https://www.cvedetails.com/vulnerabilities-by-types.php..

[81] CCN-CERT, «ciberamenzas-y-tendencias-edicion-2017,» 08 Junio 2017. [En línea]. Available: https://www.ccn-cert.cni.es/informes/informes-ccn-cert-publicos/2221-ccn-cert-ia-16-17-ciberamenzas-y-tendencias-edicion-2017-resumen-ejecutivo-1/file.html.

[82] CCN-CERT, «ciberamenzas y tendencias edicion 2018,» 10 Mayo 2018. [En línea]. Available: https://www.ccn-cert.cni.es/informes/informes-ccn-cert-publicos/2835-ccn-cert-ia-09-18-ciberamenzas-y-tendencias-edicion-2018-1/file.html.

[83] CCN-CERT, «ciberamenazas y tendencias resumen ejecutivo 2019,» 10 Junio 2019. [En línea]. Available: https://www.ccn-cert.cni.es/en/reports/public/3767-ccn-cert-ia-13-19-ciberamenazas-y-tendencias-resumen-ejecutivo-2019/file.html.

[84] CCE-CERT, «ciberamenazas y tendencias edicion 2020,» 29 Septiembre 2020. [En línea]. Available: https://www.ccn-cert.cni.es/informes/informes-ccn-cert-publicos/5377-ccn-cert-ia-13-20-ciberamenazas-y-tendencias-edicion-2020/file.html.

[85] « Instituto Nacional de Ciberseguridad de España Centro de respuestas a incidentes de seguridad.,» 2019. [En línea]. Available: https://www.incibe-cert.es/blog/seguridad-industrial-2019-cifras.

[86] «CWE,» 2011. [En línea]. Available: https://cwe.mitre.org/top25/archive/2011/2011_cwe_sans_top25.html. 2011 CWE/SANS Top 25..

[87] «MITRE - CWE 2019 Top 25,» 2019. [En línea]. Available: https://cwe.mitre.org/top25/archive/2019/2019_cwe_top25.html.

[88] «CWE MITRE 2020 Top 25,» 2020. [En línea]. Available: https://cwe.mitre.org/top25/archive/2020/2020_cwe_top25.html.

[89] «VirusTotal API v3 Overview,» 2020. [En línea]. Available: https://developers.virustotal.com/reference/overview.

[90] «MISP - Open Source Threat Intelligence Platform,» 2020. [En línea]. Available: https://www.misp-project.org/documentation/.

[91] «Interfaz gráfica Alienvault. Fuente de investigación de ataques informáticos,» 2020. [En línea]. Available: https://otx.alienvault.com/browse.

[92] «AlienVault OTX,» 2020. [En línea]. Available: https://otx.alienvault.com/api.

[93] NIST, «Nist Glosary,» 2020. [En línea]. Available: https://csrc.nist.gov/glossary/term/Threat_Agent_Source.

[94] NIST, «Nist Glosary,» 2020. [En línea]. Available: https://csrc.nist.gov/glossary/term/threat.

[95] NIST, «Nist Glosary,» 2020. [En línea]. Available: https://csrc.nist.gov/glossary/term/backup.

[96] NIST, «Extensible Configuration Checklist Description Format (XCCDF),» 16 Julio 2021. [En línea]. Available: https://csrc.nist.gov/projects/security-content-automation-protocol/specifications/xccdf.

[97] INCIBE, «Glosario de términos de seguridad,» 2017. [En línea]. Available: https://www.incibe.es/sites/default/files/contenidos/guias/doc/guia_glosario_ciberseguridad_metad.pdf.

[98] MITRE, «CVE,» 1 Febrero 2020. [En línea]. Available: https://cve.mitre.org/cve/cna/rules.html#Section_4_1_qualifications.

[99] CWE MITRE, «Common Weakness Enumeration,» 19 Julio 2006. [En línea]. Available: https://cwe.mitre.org/data/definitions/89.html.

[100] INCIBE, «Defacement Incibe,» 05 Mayo 2021. [En línea]. Available: https://www.incibe.es/aprendeciberseguridad/defacement.

[101] NIST, «Nist Glorssary,» 2020. [En línea]. Available: https://csrc.nist.gov/glossary/term/firewall.

[102] NIST, «NIST Glossary,» 2020. [En línea]. Available: https://csrc.nist.gov/glossary/term/gateway.

[103] NIST, «Nist Glossary,» 2020. [En línea]. Available: https://csrc.nist.gov/glossary/term/hardening.

[104] NETFILTER, «Netfilter Projects,» 2017. [En línea]. Available: https://www.netfilter.org/projects/iptables/index.html.

[105] NIST, «Nist Glosary,» 2020. [En línea]. Available: https://csrc.nist.gov/glossary/term/log.

[106] NIST, «Nist Glossary,» 2020. [En línea]. Available: https://csrc.nist.gov/glossary/term/malware.

[107] NIST, «National Vulnerability Database,» 15 Enero 2021. [En línea]. Available: https://nvd.nist.gov/.

[108] T. Correia, I. Pedrosa y C. J. Costa, «Software Open Source em Auditoria,» 2018. [En línea]. Available: https://ieeexplore-ieee-org.itm.elogim.com:2443/stamp/stamp.jsp?tp=&arnumber=8399428.

[109] R. Venkatesan, D. R. Devi, R. Keerthana y A. A. Kumar, «A NOVEL APPROACH FOR DETECTING DDoS ATTACK IN H-IDS USING ASSOCIATION RULE,» 2018. [En línea]. Available: https://ieeexplore-ieee-org.itm.elogim.com:2443/stamp/stamp.jsp?tp=&arnumber=8541174.

[110] NIST, «Nist Glossary,» 2020. [En línea]. Available: https://csrc.nist.gov/glossary/term/penetration_testing.

[111] NIST, «Nist Glossary,» 2020. [En línea]. Available: https://csrc.nist.gov/glossary/term/proxy.

[112] OPTIVE, «Cybersecurity Dictionary Ransomware,» 2020. [En línea]. Available: https://www.optiv.com/cybersecurity-dictionary/ransomware.

[113] NIST, «Nist Glosary,» 2020. [En línea]. Available: https://csrc.nist.gov/glossary/term/vulnerability.

[114] CYBSEC, «CYBSEC Security Systems: HONEYPOTS,» 2017. [En línea]. Available: http://www.cybsec.com/upload/ESPE_Honeypots.pdf.

[115] D. A. P. S. W. D. Yu, «Software Vulnerability Analysis for Web Services Software Systems,» de *11th IEEE Symposium on Computers and Communications (ISCC'06)*, Cagliari, Italy, 2006.

[116] IONOS, «Honeypot: seguridad informática para detectar amenazas,» 08 Agosto 2017. [En línea]. Available: https://www.ionos.es/digitalguide/servidores/seguridad/honeypot-seguridad-informatica-para-detectar-amenazas/.