**Ingeniería en computación**



*Proyecto Integrador*

**Implementación de una solución para gestión de eventos de seguridad de una red de datos**

*Autores*

Sergio David Figueroa y Federico Nicolás Sepúlveda

*Director*

Mgter. Ing. Miguel Ángel Solinas

# 

# Resumen

# Índice

[**Resumen**](#_kol2h6qgysv9) **2**

[**Índice**](#_klumoiel3imn) **2**

[**Índice de figuras**](#_udaw4l426ne) **4**

[**Índice de tablas**](#_7129zk3d87tq) **4**

[**Agradecimientos**](#_kbys14dyjqjf) **4**

[**Glosario**](#_clves114mmg4) **4**

[**Introducción**](#_msgxh4ei5l0s) **8**

[**Objetivo General**](#_78o1ww6a21kk) **8**

[Motivación](#_6in6ygfnp5ic) 8

[**Marco Teórico**](#_d9ffu4qmfjif) **10**

[Personal](#_5uf9go6vrxbl) 11

[Procesos](#_uav59ix7td1f) 13

[Tecnología](#_uxifivrhnbyi) 14

[Agregando contexto a los incidentes](#_mutedczcpitr) 15

[Definición de conductas normales](#_v38go8e8oip2) 16

[Inteligencia de amenazas](#_zf482rispops) 16

[Obstáculos para el manejo eficiente de incidentes del CSIRT](#_549bommxiogm) 16

[Ámbitos de actuación de los CSIRT](#_vv7uqf175hj2) 17

[Estado de la ciberseguridad en Argentina](#_yfm8mk50o7e7) 19

[Demanda de ciberseguridad en Argentina](#_q5byxa3cwh8b) 21

[SIEM: Definición y funciones](#_fcw24esumwr1) 23

[Soluciones disponibles](#_l27jioqn7rm) 26

[Soluciones comerciales](#_qfy2fmcg4w6m) 28

[Soluciones gratuitas y de código abierto](#_6o7syte386lv) 32

[AlienVault OSSIM](#_rfumn1y3jjr5) 33

[Graylog](#_52ie70upam3c) 34

[Elastic Stack](#_ujjvw6kyonbv) 34

[Security Onion](#_jkv2vhvda6xt) 34

[Corolario](#_ixc2mn3rcrvj) 35

[**Descripción de Requerimientos**](#_86v24fz9rll5) **36**

[Requerimientos funcionales del SIEM](#_pxpn8qxmf2xg) 36

[Requerimientos no funcionales](#_m534rw324oov) 36

[Análisis de riesgos](#_fj7tmur7lrs4) 37

[Síntesis](#_keapnm1iowx3) 42

[**Iteración 1: “Elección de la solución”**](#_g08juwy58nbz) **42**

[Security Onion como sistema de gestión de eventos](#_6xxk90feqwsh) 42

[Arquitectura del sistema de gestión de eventos](#_gbvih2nkchgr) 44

[Arquitectura de alto nivel](#_5tjrs2d54xrk) 44

[Arquitectura de despliegue](#_u3jf3sp8ta81) 46

[Tipo de Nodos](#_kn9i2cd2rftp) 46

[Tipos de Arquitectura](#_86kyhzv16i7a) 48

[Recibiendo, procesando y visualizando eventos: La pila Elastic](#_g14uofnzjar) 50

[Logstash](#_aems0za6gauc) 51

[Elasticsearch](#_zf9h5d2oz4ft) 52

[Kibana](#_7vgz3mvtajep) 53

[Elastic, ElastAlert, TheHive y Cortex](#_74h0r22ou7ny) 54

[Analizando y clasificando eventos: ElastAlert](#_pqzcyedw8mey) 54

[El panel de control general: TheHive y Cortex](#_lq03o39atr0w) 55

[Automatizando acciones: TheHiveHooks](#_y03qr0todr02) 64

[Integración con los sistemas de detección](#_bto1am9p5t5z) 66

[Suricata, Snort y Ossec](#_5el7xg6aq71u) 66

[Arquitectura del despliegue](#_juvwd52ve8ho) 68

[**Iteración 2: “Configuración y despliegue en un ambiente de prueba”**](#_xdnrscy4e329) **68**

[Configuración del ambiente de prueba](#_5b96qatzvv33) 69

[Configuración del entorno de virtualización](#_vbv9ozuusue0) 70

[Definición y configuración de las redes a observar](#_4gy5pwndbx0t) 70

[Configuración inicial del sistema base](#_pr1m3xgke3pt) 72

[Instalación y configuración de Security Onion](#_dhdn30qeke8m) 72

[Instalación y configuración de TheHive - Cortex](#_5m89g5riahoi) 74

[Configuración de acciones automáticas](#_3p0estdaov1m) 75

[**Iteración 3: “Reportes de incidentes y acciones automáticas”**](#_a0fkq9pxl854) **76**

[Análisis de prioridades de los incidentes](#_csgr7xq0owzd) 76

[Automatización de acciones](#_d4bzhip9yjkb) 78

[**Iteración 4: “Testing del sistema”**](#_eojcxztr0qz5) **79**

[Test de detección de ataques varios](#_sozhocrqz06a) 79

[Test de reportes de incidentes](#_143fjw3fk3os) 79

[Test de acciones automáticas](#_qu0xa8uwm63h) 79

[**Conclusión**](#_e2lxx5wud3th) **80**

[**Futuros trabajos**](#_eomtxgzcuip2) **81**

[**Bibliografía**](#_l9ifxkyatvuo) **81**

[**Anexos**](#_mm3s3xi5bc29) **88**

# Índice de figuras

[Figura 2](#_i1jwb047kfjj)

# Índice de tablas

# Agradecimientos

# Glosario

CCNA: acrónimo en inglés de Cisco Certified Network Associate, un certificado de validación profesional emitido por la corporación Cisco para técnicos que operan sus productos.

CERT: siglas en inglés de Computer Emergency Response Team, en español equipo de respuesta a incidentes de computadoras. Término registrado comercialmente por la universidad estadounidense de Carnegie Mellon.

CPU: siglas en inglés de Central Processing Unit, en español Unidad de Procesamiento Central.

Creative Commons: licencia que permite a cualquier usuario copiar, reproducir, adaptar, distribuir, traducir y desarrollar los contenidos multimedia sin costo alguno. La utilización de contenido que se encuentra bajo esta licencia implica reconocer al autor original.

CSIRT: siglas en inglés de Computer Security Incident Response Team, en español equipo de respuesta a incidentes de seguridad de computación. Es el equipo de profesionales, sistemas y toda la infraestructura (hardware y software) de detección y respuesta a incidentes de ciberseguridad de una organización.

DDoS: siglas en inglés de Denied Distribution Of Service, en español denegación distribuida de servicio.

DNS: siglas en inglés de Domain Name Service, en español Servicio de Nombres de Dominio, es un protocolo de red de la capa de aplicación.

Dirección MAC: siglas en inglés de Media Access Control, en español control de acceso a medios, es un conjunto de bytes que constituyen la dirección física (única) que identifica a un dispositivo conectado a una red.

EMCFFAA: Estado Mayor Conjunto de las Fuerzas Armadas Argentinas.

Firewall: Un firewall es un dispositivo basado en un software o hardware o ambos. Bloquea o permite el tráfico de red, basándose en una serie de reglas dinámicas predefinidas y políticas.

Gbps: siglas de gigabit por segundo, es una especificación técnica de la medida del ancho de banda y / o velocidad de transmisión, dependiendo del contexto.

GNU: acrónimo recursivo en inglés de “GNU is Not Unix”, en español GNU No es Unix.

GPL: siglas en inglés de General Public Licence, en español licencia pública general, es un tipo de licencia GNU.

GUI: siglas en inglés de Graphical User Interface, en español Interfaz gráfica de Usuario.

HIDS: siglas en inglés de Host Intrusion Detection System, en español sistema de detección de intrusiones en un host o punto final.

IDS: siglas en inglés de Intrusion Detection System, en español sistema de detección de intrusiones. Es un componente de software destinado al procesamiento de firmas basadas en la información recolectada del tráfico de red, mediante una sonda colocada en un enlace de la infraestructura de comunicaciones de datos. Los IDS son un componente vital de un CSIRT debido a que realizan la identificación a priori de eventos en el tráfico de datos y su clasificación como un incidente.

IMAP: siglas en inglés de Internet Message Access Protocol. Este protocolo de aplicación, permite a los usuarios acceder a sus e-mails directamente en el servidor y sólo descargar, hacia la máquina local, los mensajes y archivos adjuntos que le resulten de interés.

Información normalizada: el objetivo es modificar los mensajes de diferentes fuentes de manera tal que se adapten a un modelo de datos común.

Infraestructura de IT: corresponde a la infraestructura de tecnologías de la información (servidores, switches, routers, etc) de una organización.

IPV4 e IVP6: siglas en inglés de los protocolos de Internet versiones 4 y 6, respectivamente.

IPS: siglas en inglés de Intrusion Protection System, en español sistema de protección de intrusiones.

Licencia GFDL: siglas de GNU Free Documentation License*.* Está orientado a permitir que un manual, un libro de texto o cualquier otro documento escrito sea libre en el sentido de su difusión, copias, modificaciones y comercialización.

Licencia AGPL: siglas de GNU Affero General Public License. Esta licencia asegura los derechos de autor sobre el software y da permisos legales para la copia, distribución y modificaciones del mismo. En caso de modificaciones se debe poner a disposición de la comunidad el código fuente con dichos cambios.

Licencia APACHE: licencia de software libre permisiva creada por la Apache Software Foundation. Se diferencia de otros tipos de licencias ya que no exige copyleft en el software donde se aplica.

Licencia BSD: siglas en inglés de Berkeley Software Distribution, licencia de software libre desarrollada en dicha universidad homónima de Estados Unidos.

Linux: núcleo de código (kernel) abierto de familias del mismo nombre de sistemas operativos de software libre.

Log: equivalente en inglés a “registro” en español. Término utilizado específicamente para registros de datos con un formato definido.

Malware: software malicioso diseñado para identificar y / o explotar vulnerabilidades en los sistemas de una víctima: sistemas operativos, drivers, cualquier tipo de software, dispositivos, etc. Las consecuencias implican desde el malfuncionamiento del software o dispositivo afectado, robo o pérdida de información, hasta la inutilización total del hardware o sistema infectado.

MIT: siglas en inglés de Massachusetts Institute of Technology, universidad de los Estados Unidos cuyo nombre es usado para un tipo de licencia de código libre desarrollada en esa universidad.

MSSP: siglas en inglés de managed security service provider, en español proveedores de servicios de seguridad gestionados. Empresas que prestan servicios de seguridad informática a organizaciones.

NIDS: siglas en inglés de Network Intrusion Detection System, en español sistema de detección de intrusiones a nivel de red.

NIPS: siglas en inglés de Network Intrusion Protection System, en español sistema de protección de intrusiones a nivel de red.

NSM: siglas en inglés de Network Security Monitoring, en español monitoreo de seguridad de redes.

RAM: siglas en inglés de Random Access Memory, en español memoria de acceso aleatorio.

Ransomware: Software malicioso, que en un dispositivo puede bloquear la interfaz de usuario o cifrar las información que se encuentra en el disco y posteriormente solicitarle a la víctima un pago para recuperar los datos.

SaaS: siglas en inglés de Software as a Service, en español software como servicio, es un modelo de negocio de software a cuyo despliegue y funcionalidades están disponibles a la medida de la demanda del cliente.

SNMP:

STDIN:

SYN: bit usado en el protocolo TCP para indicar la sincronización del número de secuencia al comienzo de una comunicación utilizando el protocolo antes mencionado.

TCP: siglas en inglés de Transmission Control Protocol, en español protocolo de control de transmisión. Uno de los protocolos fundamentales en la comunicación de datos.

UDP: siglas en inglés de User Datagram Protocol. Es un protocolo que permite la transmisión sin conexión de datagramas en redes basadas en IP.

VPN: siglas en inglés de Virtual Private Network, en español red privada local

# Introducción

Este proyecto consiste en la implementación de un Sistema de Administración de Eventos y Seguridad de la Información (SIEM, por sus siglas en inglés) para la Universidad Nacional de Córdoba. Un SIEM consiste en varias herramientas como bases de datos, filtros para normalizar la información, tablero para visualizar los datos y generador de alertas entre otras. Por otro lado tenemos el monitoreo de la red en tiempo real que utiliza un Sistema de Detección de Intrusiones (IDS, por sus siglas en inglés). Este último envía los datos ya procesados al SIEM para que los almacene en la base de datos.

Además como se pretende que el SIEM funcione dentro de un Equipo de Respuesta a Incidentes de Seguridad Informática (CSIRT, por sus siglas en inglés) se necesita integrar un gestor de incidentes. Este último sirve para tener un registro de los incidentes ocurridos, permite administrar las tareas del equipo de analistas, compartir y solicitar información con otros CSIRT entre otras funciones.

## Objetivo General

El objetivo de esta tesis es el desarrollo e implantación de un sistema SIEM dentro del proyecto general de la creación del CSIRT de la Universidad Nacional de Córdoba, con el fin de otorgar al mencionado centro de respuesta, el instrumento capaz de obtener, analizar y presentar datos sobre las amenazas detectadas por los demás subsistemas del CSIRT.

## Motivación

La tecnología y la digitalización de la información convierten a los datos en un activo muy importante de las organizaciones y de los individuos en general. Es fundamental saber cómo proteger los datos para evitar ser víctima de un ciberdelito o parte involuntaria de una ciber operación a gran escala. A pesar de que actualmente las técnicas de seguridad hacia los datos y la infraestructura de redes están en auge, las herramientas de seguridad como firewalls, IDS y otras que permiten prevenir ataques informáticos no son suficientes para mitigar y tener un seguimiento de actividades maliciosas o potencialmente maliciosas para lograr fortalecer la infraestructura y prevenir futuros incidentes. Resulta necesario contar con un sistema global que permita integrar un variado conjunto de utilidades que brindan soluciones puntuales y específicas, para crear una defensa inteligente y eficiente de los activos de información de una organización.

Actualmente la infraestructura de red y los sistemas asociados conviven en un ambiente de *saturación de la información* que implica un alto costo de procesamiento y ponen a prueba permanentemente a los sistemas encargados de la optimización de los recursos de hardware y software con los que cuenta la infraestructura, tales como uso de CPU y memorias RAM de routers, switches y servidores, el almacenamiento secundario donde el desafío de retener un ingente volumen de datos generados por el exponencial y siempre creciente tráfico de la red amenaza constantemente con el colapso de los medios disponibles sin importar su capacidad de almacenamiento, entre otros problemas, configuran una avalancha constante de información que sería imposible de analizar siquiera una parte de ella en un momento determinado utilizando métodos que impliquen el procesamiento en bruto.

En esta situación, sería imposible distinguir un evento puntual y nocivo dentro de esta cantidad gigantesca de información que se genera permanentemente en la red, de un evento normal o de tráfico legítimo y en caso de identificar un potencial incidente, este tendría unas probabilidades muy altas de ser un falso positivo. Esto último es característico de los sistemas basados en el análisis de firmas, como los IDS, IPS o antivirus. Es necesario diseñar, desarrollar, implementar, configurar y probar un sistema capaz de orquestar un gran abanico de herramientas diseñadas cada una con un objetivo puntual, combinando las capacidades de todos sus subsistemas para identificar eficientemente las amenazas reales y responder en consecuencia, minimizando los falsos positivos y daños colaterales.

# 

# Marco Teórico

Las infracciones a las políticas de seguridad y los ataques han concentrado la atención sobre las capacidades de detección, investigación y mitigación de incidentes de seguridad de la información en las organizaciones. Si bien no siempre es posible evitar un incidente de seguridad, es necesario detectar y responder rápidamente para minimizar el daño. Para ello, es preciso realizar inversiones inteligentes basadas en un plan de seguridad que comprenda la realidad y necesidades específicas de la organización, ya que un gran monto de dinero o equipos adquiridos por si mismos no garantizan una mayor protección.

Este plan debe incluir personal especializado, procedimientos e infraestructura adaptados a la organización, con una gestión de objetivos a cumplir a corto, mediano y largo plazo.

Para las organizaciones que no cuentan con una capacidad de manejo de incidentes, la creación desde cero de un Computer Security Incident Response Team (CSIRT) puede ser un proceso complejo y costoso. Sin embargo, no es necesario una gran inversión para obtener las capacidades elementales ofrecidas por un CSIRT, ya que es posible desarrollar una solución específica y a escala de la organización.

Una vez identificadas las necesidades de la organización, el proceso de creación del CSIRT requiere de la creación, colaboración y comunicación entre los tres pilares que lo componen: el personal, la tecnología y los procesos, como se muestra en la Figura 1.

|  |
| --- |
|  |
| Figura 1: Pilares de un CSIRT |

El CSIRT debe tener una perspectiva flexible y escalable para mantener el ritmo de las tácticas de los adversarios, acompañando el crecimiento y evolución de la organización.

## Personal

En cuanto al personal, estos comprenden tanto a los encargados de dar respuesta a los incidentes como a los analistas del CSIRT. Si bien la propia organización puede designar a sus integrantes para asumir estas funciones, existen otras alternativas como la tercerización mediante empresas especializadas que proveen el servicio de Managed Security Service Provider (MSSP) o contratar especialistas en respuesta a incidentes en el caso de una emergencia o un problema complejo. Otra vía consiste en la creación de equipos híbridos compuestos por personal perteneciente a la organización y especialistas externos.

De acuerdo a una encuesta del SANS Institute del año 2014 [1], el 61 % de las organizaciones relevadas manifestaron haber recurrido a personal de emergencia para cubrir incidentes críticos y el 58 % tenía un equipo de respuesta propio. Por lo que las organizaciones no siempre cubren sus necesidades con miembros de su propio personal y en algunos casos las tareas recaen por completo en los servicios de terceros. Esto se debe a que, sin importar la estructura del equipo, el personal de un CSIRT debe contar con el entrenamiento necesario para tratar con los cambios en las amenazas a las que se enfrenta. En la Tabla 1 se muestran las responsabilidades y la formación requerida para cada uno de los integrantes de un CSIRT.

|  |  |  |
| --- | --- | --- |
| Título profesional | Tarea | Entrenamiento requerido |
| Nivel 1 - Analista de alertas | Supervisa continuamente la cola de alertas, monitorea el estado de los sensores y los puntos finales, clasifica las alertas de seguridad y recopila los datos necesarios para iniciar el trabajo de Nivel 2. | Procedimientos de triage de alerta y detección de intrusos. Gestión de redes, información de seguridad y eventos. Capacitación en investigación basada en host. |
| Nivel 2 - Analista de respuesta a incidentes | Realiza un análisis profundo de incidentes al correlacionar datos de varias fuentes y determina si un sistema crítico o un conjunto de datos se ha visto afectado. Asesora sobre su remediación. | Análisis avanzado de forensia de redes y basado en host. Procedimientos de respuesta a incidentes, revisiones de registros, evaluación básica de malware e inteligencia de amenazas. |
| Nivel 3 - Especialista en la materia | Se trata de un conjunto de especialistas que cubren distintas áreas de un CSIRT.  Actúan como “cazadores” de amenazas, sin esperar que se intensifiquen los incidentes. Se encuentra estrechamente involucrado en el desarrollo, ajuste e implementación de análisis de detección de amenazas. | Entrenamiento avanzado en detección de anomalías. Entrenamiento específico en herramientas para la agregación y análisis de datos e inteligencia de amenazas.  Poseen un conocimiento profundo en áreas como redes, puntos finales, inteligencia de amenazas, forensia e ingeniería inversa de malware, así como la infraestructura de IT subyacente.. |
| Director del CSIRT | Administra recursos para incluir personal, presupuesto, programación de turnos y estrategias para cumplir con los acuerdos de nivel de servicio. Se comunica con la gerencia y sirve como persona de contacto en el caso de incidentes críticos. Proporciona una dirección general para el CSIRT. | Gestión de proyectos, formación en gestión de respuesta a incidentes, habilidades generales de gestión de personas y comunicación institucional. |
| Tabla 1: Integrantes de un CSIRT y sus funciones | | |

Para organizar el trabajo de los analistas, un CSIRT necesita un director que coordine los múltiples esfuerzos dentro y fuera del equipo. Su responsabilidad es dirigir el trabajo y organizar los recursos con el fin de detectar, investigar y priorizar incidentes que puedan impactar en la organización. Otra de las misiones asignadas al director consiste en desarrollar un modelo de flujo de trabajo e implementar procedimientos operativos estandarizados, para el proceso de manipulación de incidentes, que guíen a los analistas en la clasificación y respuesta apropiada.

En la Figura 2 se observa un modelo de organización de un CSIRT.

|  |
| --- |
|  |
| Figura 2: Organización de un CSIRT |

## Procesos

Para estandarizar las acciones que pueden tomar los analistas del CSIRT y asegurar que no se perderán tareas importantes en el camino, es necesario definir procesos repetibles de clasificación de incidentes e investigación. Al crear un flujo repetible de gestión de incidentes, se definen las responsabilidades y acciones de los miembros del equipo: desde la creación de una alerta y evaluación por analistas de nivel 1, hasta el tratamiento del incidente por parte del personal de los niveles superiores. Como consecuencia, la segmentación del proceso permite una gestión eficiente de los recursos del CSIRT.

Uno de los modelos de procesos de respuesta a incidentes más utilizado es el modelo DOE/CIAC [2], que consiste en seis etapas: preparación, identificación, contención, erradicación, recuperación y lecciones aprendidas.

## Tecnología

En el núcleo de un CSIRT se encuentran las tecnologías de recolección de datos, agregación, detección, análisis y administración. En cuanto a la recolección, se trata de un sistema de monitoreo que obtiene sus datos a partir de un conjunto variado de fuentes como puntos finales (PC, dispositivos móviles, servidores, etc), redes, generadores de logs y eventos. Como resultado de la disponibilidad de los datos, antes y durante el incidente, los analistas pueden utilizar el sistema de monitoreo como una herramienta de investigación, revisando las actividades sospechosas del incidente en curso. Por otro lado, el sistema de monitoreo puede ser utilizado para generar la respuesta al incidente y potencialmente mitigar sus causas.

Un aspecto importante a considerar es la compatibilidad de las tecnologías empleadas, en particular si la organización ya cuenta con una herramienta de monitoreo existente y se busca incorporar nuevas soluciones para integrarlas a los sistemas en servicio. En la Figura 3 se ejemplifica la necesidad de compatibilidad entre sistemas y componentes.

|  |
| --- |
|  |
| Figura 3: Compatibilidad entre tecnologías de detección |

### Agregando contexto a los incidentes

La incorporación de inteligencia de amenazas y otras informaciones de contexto tales como activos e identidades, contribuye al proceso de investigación del analista de un CSIRT. En determinados casos, la información inicial que está asociada a una alerta puede ser muy limitada, por ejemplo la dirección IP del punto final sospechoso es insuficiente por sí sola para tomar una decisión.

Para que los analistas puedan investigar un incidente, generalmente necesitan más información, por ejemplo los nombres del dueño y de dominio de la máquina, registros DHCP para mapear la IP con el host al momento del incidente, etc. Si el sistema de monitoreo incorpora información de identidad y de los activos de información, entre otros datos de contexto, le permitirá al analista ahorrar tiempo y esfuerzo para priorizar los incidentes y elaborar la respuesta más apropiada.

### Definición de conductas normales

Dado que es posible observar patrones de comportamiento en usuarios, aplicaciones, infraestructura, redes y dispositivos, es útil establecer una referencia o línea base de la actividad de lo que se considerará un comportamiento normal. Esto facilitará la detección de conductas sospechosas anticipando posibles amenazas.

Un sistema de monitoreo configurado y con una base de referencia adecuada estará en condiciones de enviar alertas confiables al analista de primer nivel. Esto obtiene especial relevancia, ya que de acuerdo al citado informe del SANS Institute del año 2014 [1], uno de los principales desafíos en la utilización de registros de eventos de seguridad, es la incapacidad de distinguir actividades sospechosas de las normales. La ausencia de una referencia de “normalidad” es un obstáculo común al que se enfrentan las empresas de monitoreo y muchas organizaciones.

La mejor práctica es utilizar plataformas que pueden crear líneas o patrones de referencia mediante el monitoreo de la red y la actividad de los puntos finales durante un periodo de tiempo.

### Inteligencia de amenazas

Los CSIRT bien establecidos o maduros desarrollan continuamente la capacidad de aprovechar la inteligencia de información proveniente tanto de sus incidentes pasados como de fuentes de inteligencia compartidas. Ejemplos de estas últimas son los proveedores especializados, CSIRT aliados, divisiones policiales de cibercrimen, organizaciones de intercambio de información como la Information Technology - Information Sharing and Analysis Center (IT-ISAC) [3], etc.

La capacidad de utilizar la inteligencia de amenazas contribuye a mejorar la precisión de la detección. De esta manera, es posible detectar patrones de amenazas ocultas en puntos finales, logs y registros de red, reduciendo las oportunidades de desarrollo de un ataque.

### Obstáculos para el manejo eficiente de incidentes del CSIRT

Algunos de los obstáculos que deben ser evitados por un CSIRT son aquellos que generan cuellos de botella en el proceso de respuesta a incidentes. Este proceso que consiste en el traslado de un incidente entre los sucesivos niveles del CSIRT, eventualmente puede generar “ruido blanco”: la presencia de una gran cantidad de alertas de poca importancia y / o falsos positivos. De prolongarse esta situación en el tiempo, se produce un fenómeno llamado “fatiga de alertas” que afecta a los analistas provocando una disminución en sus capacidades de atender incidentes prioritarios.

Al momento de elegir una herramienta de monitoreo, se debe considerar que incluya entre sus características la personalización del umbral de alertas y la posibilidad de combinar distintas alertas en un mismo incidente. Una herramienta de este tipo permite a los analistas clasificar las alertas más rápido, reduciendo las capas de evaluación necesarias antes de que el evento pueda ser confirmado y mitigado.

## Ámbitos de actuación de los CSIRT

En la actualidad existen en todo el mundo CSIRT pertenecientes a organizaciones que responden a distintos ámbitos de la sociedad y de diferente naturaleza (pública o privada). En términos generales, estos equipos se clasifican dependiendo de la comunidad a la que atienden, diferenciándose entre:

* **CSIRT para el sector de PYMES:** En este caso, el tamaño de las empresas hace poco viable que las organizaciones de este sector puedan implementar de forma individual las funciones de un CSIRT. Por lo tanto, surge la necesidad de unificar esfuerzos y servicios en un solo equipo capaz de dar soporte a varias empresas. La naturaleza de estos CSIRT puede ser pública o privada, dependiendo del contexto en el que se encuentren estas compañías.
* **CSIRT académico:** El área de responsabilidad de este tipo de equipos se circunscribe a instituciones académicas. Su tamaño, por lo tanto, puede variar dependiendo de las dimensiones de la comunidad, condicionando los servicios que ofrezcan, el modo en que lo hagan y su grado de intervención.
* **CSIRT comercial:** estos centros prestan distintos servicios a cambio de una contraprestación económica. Se trata de empresas especializadas en la industria de la ciberseguridad, que habitualmente utilizan acuerdos de servicios específicos con cada cliente.
* **CSIRT de proveedor:** se centra en los productos o servicios específicos de un proveedor. Su objetivo es proveer servicios y soluciones para eliminar o reducir el impacto negativo de las vulnerabilidades en estos últimos, ya sea un producto tecnológico o un servicio TIC.
* **CSIRT del sector militar:** Prestan servicios a organizaciones militares, con responsabilidades en infraestructuras TIC necesarias para la Defensa. Su comunidad está conformada por las instituciones militares y de entidades estrechamente relacionadas con éstas. Por ejemplo, en nuestro país el Comando Conjunto de Ciberdefensa [11], es el encargado de la defensa de la infraestructura de redes y activos de la información de las Fuerzas Armadas.
* **CSIRT para protección de infraestructuras críticas:** Los CSIRT de este sector se centran principalmente en la protección de las infraestructuras críticas y de los activos de información asociados. Ejemplos de infraestructuras críticas son las centrales y redes de energía, telecomunicaciones, sistema financiero, sector sanitario, agua, transportes, industria nuclear, etc.
* **CSIRT gubernamental:** Bajo esta denominación se sitúan los equipos cuyo principal objetivo es asegurar la infraestructura TIC de un Gobierno/Estado y los servicios ofrecidos a la población. La Comunidad a la que están dirigidos son las administraciones públicas y sus distintos organismos. Estos CSIRT gubernamentales generalmente forman parte de las instituciones del Estado.
* **CSIRT Nacional:** Este es un equipo con responsabilidad general de coordinación sobre todos los sectores y tiene una amplia responsabilidad sobre prácticamente todos los CSIRT tratados anteriormente. Este centro funciona como punto focal de contacto tanto en el entorno nacional como para requerimientos internacionales. ENISA, la Agencia de Ciberseguridad para la Unión Europea, define en un documento [4] elaborado en diciembre de 2009, a este tipo de CSIRT como “aquel que actúa como el Point of Contact (POC) con otros equipos nacionales y/o internacionales. De hecho, podría considerarse como CSIRT del último recurso, por su papel de coordinación”. Cada nación define la misión de estas unidades y establece sus operaciones, su organización y su imperativo legal en base a las necesidades del país y su comunidad. En Argentina, esta responsabilidad es asumida por la Dirección Nacional de Ciberseguridad [5].

|  |
| --- |
|  |
| Figura 4: CSIRT Nacional como centro de coordinación |

Frente a esto, todos los CSIRT nacionales tienen un objetivo común, mantener seguras las redes de sus países. De este modo podemos concluir que, aunque cada uno utiliza herramientas y procedimientos diferentes, todos comparten el mismo objetivo:

* + Designar un punto de contacto para la coordinación de la respuesta a incidentes.
  + Construir y mantener una red de contactos extensa, tanto nacional como internacional.
  + Monitorización de la situación actual y mejora de la concientización.

Es importante destacar que la constitución de un CSIRT nacional o gubernamental, no es la única medida a tener en cuenta en una estrategia de ciberseguridad completa por parte de un Estado. Es una parte importante de la misma teniendo en cuenta, además, que este tipo de equipos deberían asumir la responsabilidad de la Protección de las Infraestructuras Críticas de Información (CIIP) del país.

### Estado de la ciberseguridad en Argentina

Los orígenes de la gestión de la seguridad informática en nuestro país pueden rastrearse en el decreto 856/98 [6] y a la Resolución de la Secretaría de la Función Pública (SFP), organismo dependiente de la Jefatura de Gabinete de Ministros, Res SFP 81/99 [7]. Esta Resolución establece la reorganización de la Subsecretaría de Tecnologías Informáticas y el “Reglamento de Operación del ArCERT", donde se indican los requisitos y condiciones de operación de la Coordinación de Emergencia en Redes Teleinformáticas - ArCERT, y las Políticas de Seguridad del mismo.

El objetivo principal del ArCERTfue coordinar y colaborar en los esfuerzos orientados a elevar los umbrales de seguridad en los recursos y en los sistemas de información en el ámbito de la Administración Pública Nacional (APN). Para esto se estableció una estrategia de coordinación, asesoramiento y capacitación hacia los organismos públicos en la gestión de la problemática de seguridad.

Sus funciones [7] eran:

1. Proveer un servicio especializado de asesoramiento en seguridad de redes.
2. Promover la coordinación entre los organismos de la Administración Pública Nacional para prevenir, detectar, manejar y recuperar incidentes de seguridad.
3. Centralizar los reportes sobre incidentes de seguridad ocurridos en la APN y facilitar el intercambio de información para afrontarlos.
4. Actuar como repositorio de toda la información sobre incidentes de seguridad, herramientas, técnicas de protección y defensa.

En el marco de las funciones nombradas, el ArCERT realizó actividades de investigación de amenazas y nuevas soluciones disponibles, divulgación de incidentes y soluciones así como capacitaciones de seguridad en redes y seminarios de actualización periódicos.

El año 2011 marcó el comienzo de un largo proceso de reestructuración de los organismos del Estado Nacional relativos a la ciberseguridad, que incluyó el reordenamiento de estructuras internas, la creación de nuevas dependencias y el reemplazo o absorción de unidades preexistentes por las de nueva formación. Ese mismo año, mediante la resolución de la Jefatura del Gabinete de Ministros, Res JGM Nº 580/2011 [8] se creó el Programa de Infraestructuras Criticas de Informacion y Ciberseguridad, que declaró como finalidad “Impulsar la creación y adopción de un marco regulatorio específico que propicie la identificación y protección de las Infraestructuras estratégicas y críticas del Sector Público Nacional, los organismos interjurisdiccionales y las organizaciones civiles y del sector privado que así lo requieran”.

En el año 2013 y mediante el artículo 1º de la Disposición Nº 2/2013 [9] de la Oficina Nacional de Tecnologías de Información (ONTI), se creó el Instituto de Ciencias e Ingeniería de Computación - Computer Emergency Response Team (ICIC-CERT) que reemplazó al ArCERT. Este nuevo CERT heredaba parte de las responsabilidades del original, a la par que otros artículos de la referida disposición creaban nuevos grupos de trabajo especializados que ampliaban las capacidades, funciones y responsabilidades concentradas originalmente en el ArCERT, tales como el grupo ICIC - GAP (Grupo de Acción Preventiva, art. 3º) con funciones similares a las encargadas al ArCERT pero enfocadas a monitorear “los servicios que el Sector Público Nacional brinda a través de la red de Internet y aquellos que se identifiquen como Infraestructura Crítica para la prevención de posibles fallas de Seguridad”, el grupo “ICIC - GICI” (Grupo de Infraestructuras Críticas de Información, art. 4º) especializado en el desarrollo y aplicación de nuevas tecnologías para el monitoreo, simulación y respuesta a incidentes en la red de infraestructuras críticas, establecer prioridades y planes estratégicos para liderar el abordaje de la ciberseguridad para la protección de este tipo de infraestructuras, coordinar la implementación de ejercicios de respuesta ante la eventualidad de un intento de vulneración de estos activos, entre otras. Finalmente, se creó el grupo de trabajo “ICIC - INTERNET SANO” (art. 7º) con el objetivo específico de ocuparse de las tareas de difusión y capacitación que anteriormente le correspondía al ArCERT. En cuanto al programa de infraestructuras críticas, podemos mencionar la adhesión de la Universidad Nacional de Córdoba, cuando el 15 de Julio de 2014, mediante la Resolución 1221 [10] firmada por el Rector Tamarit, en su artículo N°1 “Hacer lugar a lo solicitado a fS.1 por la Prosecretaría de Informática y, en consecuencia, adherir al "Programa Nacional de Infraestructuras Críticas de Información y Ciberseguridad"...”

El Estado Nacional siguió actualizando sus políticas en los años siguientes, creando nuevos centros de respuesta a incidentes y actualizando la normativa vigente. Algunos de los ejemplos son la creación del Comando Conjunto de Ciberdefensa de las Fuerzas Armadas [11], el “MING-CSIRT” [12] del Ministerio de Seguridad de la Nación, la Dirección Nacional de Ciberseguridad[5] y sus correspondientes unidades de gobierno.

|  |  |
| --- | --- |
|  |  |
| Figura 5: Insignias de CSIRTS argentinos: A la izquierda la correspondiente al Comando Conjunto de Ciberdefensa y a la derecha la del CSIRT de la provincia de Neuquén. | |

Por otro lado, los Estados Provinciales, universidades y empresas también han desarrollado e implantado CSIRTs en sus organizaciones. Algunos ejemplos de esto son el “BA-CSIRT”[13] de la Ciudad de Buenos Aires, el “CSIRT-NQN”[14] del gobierno de la provincia de Neuquén, “CERT UNLP”[15] de la Universidad Nacional de La Plata o el que dispone NIC Argentina[16] destinado a la infraestructura crítica de DNS. En el caso de las empresas, podemos mencionar a los CSIRT de las redes bancarias Link[17] y Banelco[18].

#### Demanda de ciberseguridad en Argentina

Nuestro país, de manera análoga a los demás países de la región, ha experimentado un crecimiento exponencial de incidentes de ciberseguridad a lo largo de las últimas dos décadas. Esto afecta a individuos, empresas, universidades, infraestructuras críticas y a los organismos de diferentes niveles del gobierno. Este incremento exponencial en la demanda de ciberseguridad no ha encontrado una respuesta adecuada del Estado Argentino. Un ejemplo lo constituyen los 1590 millones de ataques[19] que sufrió el sistema bancario argentino en el año 2019 o los 187 millones de ciberataques entre enero y marzo del 2020 con un incremento del 131% solo para el mes de marzo, según la plataforma “Fortinet Threat Intelligence Insider Latin America”[20].

En la Figura 6 se observan las diez amenazas más frecuentes y la cantidad de ataques producidos en Argentina por ellas, en el periodo comprendido entre los meses de abril y junio de 2020. Los tres primeros corresponden al troyano de puertas traseras “DoublePulsar Backdoor”, negociación de cifrados SSL anónimos e intentos de ataque contra una vulnerabilidad de divulgación de información en el servidor SMB de Microsoft Windows. La fuente es un informe de Fortinet Threat Intelligence Insider Latin America [20].

|  |
| --- |
|  |
| Figura 6: Cantidad de intentos de intrusión según la amenaza. Abril - Junio de 2020. [20] |

Hay informes del gobierno nacional [21] disponibles desde el 2016 que anticipan esta tendencia, en particular el Ministerio de Modernización indicó que el país registró 5.400 millones de dólares en pérdidas atribuidas a incidentes de ciberseguridad. Esto representa el 1% del PBI del mismo año. La Figura 7 del mencionado informe muestra los tipos de incidentes más frecuentes del año 2016

|  |
| --- |
|  |
| Figura 7: Porcentaje de tipos de incidentes durante 2016.[21] |

Este costo, enorme tanto en relación a la producción nacional como en forma cuantitativa, resalta la necesidad cada vez más importante de desarrollar políticas públicas activas en forma de inversiones en la ampliación de las capacidades de análisis y respuesta a amenazas de ciberseguridad en los organismos de la Administración Pública.

Resulta por lo tanto, necesaria la promoción de iniciativas privadas junto a la concientización de los riesgos existentes, el fomento de investigaciones académicas y la formación de recursos humanos capaces de afrontar estos desafíos, junto a una legislación adecuada y actualizada, para responder al constante aumento de amenazas y complejos escenarios que implica la creciente actividad en línea de industrias, organismos del Estado, comercio y desarrollo de la sociedad en Internet.

## SIEM: Definición y funciones

Las capacidades que ofrece un CSIRT para las organizaciones, en términos de prevención y mitigación de incidentes informáticos, se basan en los tres pilares mencionados en las secciones anteriores: el personal, la tecnología y los procesos.

En cuanto a la tecnología, es la que permite llevar a cabo las tareas de recolección de datos, agregación, detección, análisis y administración. Estas tecnologías se encuentran dentro del marco operacional del *Security Information and Event Management* (SIEM) como entidad dentro de la estructura de organización de un CSIRT.

El proceso de monitoreo de la seguridad de una red de datos compleja requiere recopilar diferentes tipos de datos para detectar, verificar y contener acciones ofensivas. Las tecnologías del SIEM proporcionan informes en tiempo real y análisis de eventos de seguridad a largo plazo, como se muestra en la Figura 8. Todo esto ayuda a la tarea de un analista de ciberseguridad cuando debe verificar acciones ofensivas sobre la red de una organización.

El término SIEM fue acuñado en 2005 por los analistas Amrit Williams y Mark Nicolett de la compañía estadounidense Gartner[23], una empresa especializada en investigación y consultoría de incidentes de seguridad, unificando los acrónimos en inglés SIM (security information management) y SEM (security events management) para describir metodologías muy similares pero ligeramente diferentes de ciberseguridad. Esta superposición de tareas hizo evidente que un nuevo término podría englobar ambos conjuntos de funciones, con el fin de disponer de un único acrónimo que pudiese identificar a una plataforma capaz de resolver los objetivos de los sistemas predecesores.

De esta manera, el nuevo acrónimo SIEM significa Administración de Eventos de Seguridad de la Información, por sus siglas en inglés. Como plataforma que combina las funciones de los sistemas anteriormente descritos, sus capacidades comprenden la siguiente lista de tareas:

* Recolectar, analizar y presentar de manera eficiente datos relacionados a la seguridad.
* Análisis en tiempo real de eventos de seguridad
* Generar reportes y almacenar datos relacionados a la seguridad
* Administración de niveles y tipos de acceso e identidad.
* Auditoría de registros
* Respuesta a incidentes y operaciones de seguridad.

Como se puede observar en la Figura 8, para cumplir con las tareas anteriormente descritas, un SIEM obtiene su información de diversas fuentes:

* Inteligencia de Amenazas, identidades y logs: la revisión de amenazas pasadas y la inteligencia compartida por otras organizaciones aliadas contribuyen a detectar más rápido patrones anormales de comportamiento.
* Telemetría de Netflow: es un protocolo de recopilacion de informacion del flujo de red IP.
* Captura de paquetes: replicación de paquetes de información con el fin de correlacionar posibles amenazas en el flujo de red.
* Dispositivos Antimalware: componentes especializados en la detección de malware, localizados en puntos finales.
* IDS (HIDS y NIDS): Sistemas de detección de intrusiones, orientados a redes (NIDS) o puntos finales (HIDS).
* Firewalls: software que bloquea y filtra conexiones con origen desde el sistema hacia el exterior y viceversa.
* IPS: Sistema de protección de intrusiones. Ofrecen protección activa frente a comportamientos inusuales ya que pueden tomar acciones programadas para evitar un intento de intrusión.
* Syslog: Syslog es un protocolo para el tratamiento de logs en un formato estandarizado, empleado fundamentalmente en el monitoreo de la integridad de servidores.

|  |
| --- |
|  |
| Figura 8: Funciones y flujos de datos de un SIEM. |

La secuencia del tratamiento que recibe la información dentro un SIEM está representada en la Figura 9, donde se muestran los cuatro pasos que forman el proceso de respuesta a incidentes del SIEM. La secuencia se detalla a continuación:

* Paso 1: inicialmente es necesario recolectar datos provenientes de diversas fuentes, tales como sistemas IDS, firewall, syslog, switches (protocolos SNMP), entre otros. Como estos datos tienen formatos disímiles entre sí, es necesario un posterior proceso de normalización.
* Paso 2: a medida que los datos arriban al SIEM, es necesario someterlos a un proceso de normalización. Este proceso consiste en extraer la información contenida en los distintos mensajes, para almacenarla en un formato estandard. Con el objetivo de reducir el volumen de almacenamiento, la información resultante es sometida a un proceso de agregación, que separa los datos importantes de los secundarios.
* Paso 3: es posible analizar los datos resultantes en busca de patrones de actividad inusual. Esta búsqueda se puede realizar de manera manual por un analista o en forma autónoma en el caso de disponer de herramientas que utilicen inteligencia de amenazas.
* Paso 4: con los resultados del análisis del paso anterior, es posible determinar la existencia de un intento de intrusión u otros tipos de incidentes. De confirmarse este último caso, es posible enviar alertas a los responsables de tomar decisiones para mitigar el incidente.

|  |
| --- |
|  |
| Figura 9: Proceso de identificación y respuesta a incidentes de un SIEM |

El objetivo, por lo tanto, de un sistema SIEM dentro del CSIRT es concentrar la información proveniente de múltiples fuentes independientes, procesar los datos y realizar un análisis centralizado.

## Soluciones disponibles

Las principales soluciones que existen en el mercado, sean gratuitas, de software libre, pagas o de código propietario, se presentan a continuación.

Se realizó un relevamiento en sitios especializados y de referencia para el mercado mundial, con el objetivo de identificar las herramientas que actualmente dominan el mercado global de SIEM disponibles comercialmente. En el sitio web de Gartner[24] se encontró un listado de los productos ordenados por valoración de los usuarios de dicha página. El resultado de los diez primeros productos junto a sus desarrolladores, se muestra en la Tabla 1.

|  |  |  |
| --- | --- | --- |
| Posición | Nombre de la solución | Empresa desarrolladora |
| 1) | QRadar SIEM[25] | IBM |
| 2) | ManageEngine ADAudit Plus[26] | ManageEngine |
| 3) | LogRhythm NextGen SIEM Platform[27] | LogRhythm |
| 4) | LogPoint - SIEM[28] | LogPoint |
| 5) | McAfee Enterprise Security Manager[29] | McAfee |
| 6) | ArcSight Enterprise Security Manager (ESM)[30] | Micro Focus |
| 7) | InsightIDR[31] | Rapid7 |
| 8) | Elastic (ELK) Stack[32] | Elastic |
| 9) | Splunk Enterprise[33] | Splunk |
| 10) | Exabeam Security Management Platform[34] | Exabeam |
| Tabla 1: Ranking Gartner[24] de soluciones SIEM 2020 | | |

En base a la información recolectada, se procedió a clasificar las herramientas bajo el criterio de la disponibilidad del código: propietarias o libres y el resultado se muestra en la Tabla 2.

|  |  |
| --- | --- |
| **Soluciones Pagas y Propietarias** | **Soluciones Libres y Gratuitas** |
| Splunk[33] |  |
| McAfee Enterprise Security Manager[29] |  |
| AlienVault USM[35] | Graylog[36] |
| QRadar SIEM [25] | Elastic (ELK) Stack[32] |
| ManageEngine ADAudit Plus[26] | AlienVault OSSIM[37] |
| LogRhythm NextGen SIEM Platform[27] | Security Onion[38] |
| LogPoint - SIEM[28] | Sweet Security[39] |
| ArcSight Enterprise Security Manager (ESM)[30] |  |
| InsightIDR[31] |  |
| Tabla 2: Comparación entre distintos tipos de soluciones SIEM | |

### Soluciones comerciales

En nuestro análisis de las soluciones comerciales disponibles, se decidió describir la situación del mercado internacional en función de los siguientes criterios:

Usuarios (descritos a través de los sectores industriales a los que pertenecen), la distribución geográfica de los usuarios a nivel global y luego repetimos el análisis con un énfasis en América Latina.

Respecto de los sectores industriales que hacen uso de sistemas de ciberseguridad y generan demandas de nuevas soluciones, se observó que el sector financiero a nivel mundial es el que lidera el consumo de soluciones SIEM, con la mayoría de los desarrolladores teniendo como clientes principales a empresas y organizaciones de este sector. Esto se explica dado el alto nivel de digitalización de la banca y los servicios financieros, que por su masividad y naturaleza son objetivos prioritarios para cualquier atacante en el ciberespacio.

Finalmente, se hizo una comparación por características entre los principales productos SIEM del mercado internacional, en base a las revisiones de sus usuarios en distintos medios, en particular se tuvo referencia a las publicaciones de “Gartner”[24] y “Markets & Markets”[4b]. Los resultados se encuentran en la Tabla 3.

|  |
| --- |
|  |
| Tabla 3. Valoración de las características |

En esta tabla comparativa se observó que FortiSIEM y ADAudit Plus, de las compañías Fortinet[40] y ManageEngine[26] respectivamente, son las soluciones mejor valoradas.

|  |
| --- |
| Gráfico |
| Figura 10: Porcentaje de adopción según el sector industrial |

En las Figuras 10 y Figura 11 se aprecia que el sector financiero es el principal demandante de estas soluciones a nivel global, seguido por las industrias manufactureras y de salud. Los sectores industriales se destacan con diferencia respecto al resto en términos de adopción de soluciones SIEM. Esto se debe a que la automatización de la industria primero y su evolución al actual modelo de “industrias 4.0”[41], con cadenas de producción, montaje y ensamble distribuidas geográficamente alrededor del globo, implica uso masivo de sensores, redes y datos intrínsecos a cada fase de producción. Esto produjo la necesidad de contar con soluciones de ciberseguridad para evitar incidentes que impliquen el posible robo de información crítica o secretos industriales. De manera análoga al sector manufacturero, las clínicas, hospitales[42] y centros de salud han sufrido el impacto de la digitalización de sus procesos tanto en el hardware médico, el almacenamiento y distribución de la información como en la protección de los sensibles datos privados de los pacientes y la estricta normativa que los regula.

En la Figura 11 se observa que las soluciones QRADAR[25], LogPoint[28] y FortiSIEM[40] concentran su demanda en Europa, Medio Oriente y África. En el continente americano existe un contraste entre los mercados de América del Norte y de América Latina. En el norte del continente la mayor parte del mercado es atendido por las soluciones de Enterprise Security Manager[30], InsightIDR[31] y AlienVault USM[35], mientras que la plataforma FortiSIEM es la preferida en Iberoamérica.

Sobre nuestra región, en la Figura 12 se observó que FortiSIEM acapara el 34% del mercado, seguido de Enterprise Security Manager, ADAudit Plus y NextGen SIEM Platform con un 17, 14 y 13 % del mercado, respectivamente.

|  |
| --- |
| Gráfico |
| Figura 11: Porcentaje de despliegue según área geográfica |

|  |
| --- |
| Gráfico |
| Figura 12 mercado de sistemas SIEM comerciales en América Latina |

### Soluciones gratuitas y de código abierto

Como consecuencia del relevamiento de las soluciones libres que se encuentran disponibles, se observó que la oferta de productos que cubren estas necesidades es acotada. Incluso en algunos casos, hay proyectos que se encuentran en estado de abandono, como por ejemplo *Sweet Security*[39]*.* Este proyecto no cuenta con soporte desde el año 2017. Por otro lado, las versiones abiertas de productos propietarios presentan serias limitaciones respecto de su equivalente comercial, lo que dificulta su consideración como alternativa viable para una organización.

Sin embargo, existen soluciones íntegramente libres capaces de cumplir adecuadamente las misiones de un SIEM, constituyendo una opción válida y competitiva frente a los principales productos comerciales.

Al igual que en el caso de los productos comerciales, se realizó un análisis sobre los sectores industriales a nivel mundial donde predominan las soluciones gratuitas. Como resultado, se observó un uso intensivo en las áreas de servicios IT, esto se debe a que en estas áreas predominan empresas desarrolladoras de tecnología de la información y entes gubernamentales, por lo que en ambos casos cuentan con el personal y recursos necesarios para modificar las soluciones a la medida de sus objetivos e infraestructura. En la Figura 13 se puede observar el porcentaje de adopción por área en la industria.

|  |
| --- |
| Gráfico |
| Figura 13: Porcentaje de adopción según el sector industrial para el caso de las soluciones libres. Gartner 2019[24] |

Por otro lado, se realizó el estudio de la región y el porcentaje de uso de las soluciones. Se observó que la pila Elastic tiene una presencia significativa en todas las regiones, como se puede ver en la Figura 14.

|  |
| --- |
| Gráfico |
| Figura 14 Porcentaje segun la region de implementación |

Otra conclusión que se puede destacar del gráfico es que en américa del norte y del sur la solución SIEM más utilizada es AlienVault OSSIM[37], mientras que en Europa, Medio Oriente y África es Graylog[36] así como en Asia/Pacífico optan por la pila Elastic[32].

#### AlienVault OSSIM

AlienVault OSSIM[37] es una solución desarrollada por la compañía estadounidense de telecomunicaciones AT & T. Se trata de una alternativa libre a AlienVault USM, que es la solución comercializada por la empresa. Permite la recopilación, normalización y correlación de eventos, ofreciendo un subconjunto de las capacidades de la variante comercial. El fabricante recomienda esta solución para profesionales IT de organizaciones pequeñas y de pocos recursos, investigadores de seguridad y miembros de la comunidad académica.

Sus capacidades de monitoreo incluyen entornos físicos y virtuales pero no permite un despliegue descentralizado, ya que solo admite la instalación en un único servidor. Entre sus limitaciones se cuentan la ausencia de gestión de logs y la capacidad de detección en puntos finales. A pesar de esto, es la solución SIEM más utilizada en América Latina.

#### Graylog

Graylog[36] es una solución SIEM desarrollada por la compañía del mismo nombre. Dispone de una arquitectura enfocada a la administración centralizada de logs y, a diferencia de otros sistemas, es capaz de recibir y procesar datos en distintos formatos tales como Syslog, NetFlow, JSON, entre otros. Permite una arquitectura de despliegue descentralizada, donde la recolección, normalización y correlación de eventos se llevan a cabo de manera centralizada. Utiliza una base de datos de elasticsearch y es una solución apta para escalar al tamaño que requiera cualquier organización.

Entre sus principales ventajas se encuentra su motor de procesamiento de logs, que permite consultar y procesar información de una manera más eficiente que otras soluciones disponibles. Otra de sus ventajas es la tolerancia a fallos, que previene la pérdida de datos en el caso de producirse algún problema de red. Esto facilita el almacenamiento distribuido de datos y el balanceo de carga en las bases de datos. Como se observa en la Figura 14, Graylog es la solución SIEM más utilizada en Europa, Medio Oriente y África.

#### Elastic Stack

Elastic Stack[32], a diferencia de las soluciones descritas anteriormente, no constituye por sí mismo un SIEM como plataforma integral. Sin embargo, es utilizado como columna vertebral de muchos proyectos de software libre, código propietario y soluciones a medida debido a su capacidad de procesamiento de la información.

Elastic Stack se destaca por el soporte de mensajes provenientes de múltiples fuentes y despliegue de distintos tipos de arquitecturas. Como resultado, dispone de capacidades de almacenamiento tales como escalamiento de infraestructuras, alta disponibilidad, recuperación automática de réplicas y balanceo de carga, etc. En cuanto a la visualización cuenta con una GUI web que permite definir consultas e interactuar con los datos recibidos, mostrando los resultados de una manera intuitiva y visible.

#### Security Onion

Se trata de una distribución Linux basada en Ubuntu que cuenta con un suite de diversas herramientas dedicadas a la ciberseguridad. Utiliza la pila completa de Elastic Stack para la recopilación, procesamiento y visualización de la información. Admite distintos tipos de arquitectura de despliegue, permitiendo adaptar los recursos disponibles según las necesidades de una organización. Actualmente se encuentra en desarrollo y cuenta con el soporte activo de una comunidad. Es la solución elegida para el desarrollo de este proyecto integrador.

## Corolario

El uso masivo de las tecnologías de la información, así como la convergencia e interconexión de redes y sistemas, ha generado nuevos tipos de riesgos y amenazas en las organizaciones. Los ataques han evolucionado en complejidad, sigilo y focalización de los objetivos, implicando de esta manera mayores esfuerzos para su detección y resolución.

Las organizaciones, públicas y privadas, deben esforzarse en preservar la seguridad de sus activos de información y responder a los nuevos riesgos e incidentes. Esto lleva a la necesidad de desplegar soluciones del tipo CSIRT.

La Universidad Nacional de Córdoba, por sus características como organización y teniendo en cuenta el panorama anteriormente descrito, no se encuentra exceptuada de posibles ciberataques contra sus activos de información. Por lo tanto, se encuentra justificada la creación de un CSIRT que atienda a sus propias necesidades.

# Descripción de Requerimientos

Con el objetivo de desplegar un sistema SIEM capaz de soportar los requerimientos funcionales y cumplimentar las condiciones que dictan los requerimientos no funcionales, es necesario definir el entorno en el que operará la plataforma.

Para esto se requiere, en primer lugar describir la topología de la red de la organización junto a lo que ello implica: realizar un relevamiento de las conexiones de la infraestructura de red interna de la unidad central, las de sus dependencias y la red entre las unidades geográficamente distribuidas si las hubiera. Deben incluirse la topología de las conexiones de salida a Internet.

En segundo lugar será necesario inventariar los activos de la organización: se requerirá una investigación y un relevamiento de los activos con los que cuentan las infraestructuras de red y de datos a fin de clasificarlos.

Las tareas de relevamiento anteriormente descritas proporcionarán un entendimiento acabado y profundo de la situación en la que se encuentra la infraestructura. Como resultado, será posible identificar puntos críticos a tener en consideración y como consecuencia, elegir la solución que mejor se ajuste a las necesidades de la organización.

## Requerimientos funcionales del SIEM

A continuación se listan los requerimientos funcionales:

1. Recolectar y almacenar datos de incidentes de seguridad en la infraestructura de la red corporativa.
2. Recolectar y almacenar información contextual y asociada a los activos vinculados al incidente.
3. Visualizar las alertas en un tablero de mando.
4. Implementar un sistema de envío de alertas de seguridad que notifique a los responsables de activos de información afectados.
5. Definir un criterio para priorizar alertas.
6. Implementar un sistema de correlación de alertas de seguridad.

## Requerimientos no funcionales

A continuación se listan los requerimientos no funcionales:

1. La solución propuesta debe utilizar software libre
2. La solución debe ser compatible con sistemas operativos abiertos y de tipo Unix.
3. La arquitectura de la solución debe soportar la escalabilidad horizontal a demanda de la organización.
4. Se requiere un despliegue automatizado de la solución.

## Análisis de riesgos

A partir de la definición de los requerimientos, se procedió a realizar un análisis de riesgo en el que se evaluaron las distintas soluciones SIEM mencionadas en el marco teórico. El objetivo de esta comparación fue encontrar la solución que mejor cumpliese los requerimientos formulados.

En primer lugar, se codificaron los requerimientos funcionales y no funcionales Cuadro [6], así como las soluciones disponibles Cuadro [7]. Esta codificación fue utilizada para comparar las soluciones contra los requerimientos del proyecto, para obtener una apreciación general de la viabilidad de cada solución, en términos de su potencial elección para el desarrollo del proyecto. Los Cuadros [6] y [7] se presentan a continuación:

|  |  |
| --- | --- |
| Siglas | Requerimientos |
| RF1 | Recolectar y almacenar datos de incidentes de seguridad en la infraestructura de la red corporativa. |
| RF2 | Recolectar y almacenar información contextual y asociada a los activos vinculados al incidente. |
| RF3 | Visualizar las alertas en un tablero de mando. |
| RF4 | Implementar un sistema de envío de alertas de seguridad que notifique a los responsables de activos de información afectados. |
| RF5 | Definir un criterio para priorizar alertas. |
| RF6 | Implementar un sistema de correlación de alertas de seguridad. |
| RNF1 | La solución propuesta debe utilizar software libre. |
| RNF2 | La solución debe ser compatible con sistemas operativos abiertos y de tipo Unix. |
| RNF3 | La arquitectura de la solución debe ser escalable a demanda de la organización. |
| RNF4 | Se requiere un despliegue automatizado de la solución. |
| Cuadro 6 | |

|  |  |
| --- | --- |
| Sigla | Solución |
| S1 | Graylog |
| S2 | Elastic (ELK) Stack |
| S3 | AlienVault OSSIM |
| S4 | Security Onion |
| S5 | Splunk |
| S6 | McAfee Enterprise Security Manager |
| S7 | AlienVault USM |
| S8 | QRadar SIEM |
| S9 | ManageEngine ADAudit Plus |
| S10 | LogRhythm NextGen SIEM Platform |
| S11 | LogPoint - SIEM |
| S12 | ArcSight Enterprise Security Manager (ESM) |
| S13 | InsightIDR |
| Cuadro 7 | |

Posteriormente, se clasificaron los requerimientos funcionales y no funcionales en base a su prioridad para el proyecto. Se eligieron tres categorías: Alta, Media y Baja. A continuación se procedió a describir detalladamente estos niveles.

Los requerimientos funcionales definidos como de “Alta” prioridad son los que se consideraron que no pueden faltar en una solución SIEM, ya que son inherentes a la naturaleza del mismo. Por otro lado los requerimientos no funcionales de la misma categoría plantean una restricción que no se puede evitar. En caso de incumplimiento de por lo menos uno de estos requerimientos, llevó a descartar la solución.

El nivel de prioridad “Media” para los requerimientos implica que el cumplimiento de estos es deseable para hacer mejor uso de la solución, pero no impactan significativamente en el desempeño general si no son implementados.

Por último, un nivel de prioridad “Baja” implica que los requerimientos de este tipo están orientados a optimizar la solución una vez que esta ya se encuentre funcionando. Su incumplimiento no es crítico.

La información resultante se muestra en el Cuadro [8].

|  |  |  |
| --- | --- | --- |
| Siglas | Requerimientos | Prioridad |
| RF1 | Recolectar y almacenar datos de incidentes de seguridad en la infraestructura de la red corporativa. | Alta |
| RF2 | Recolectar y almacenar información contextual y asociada a los activos vinculados al incidente. | Media |
| RF3 | Visualizar las alertas en un tablero de mando. | Alta |
| RF4 | Implementar un sistema de envío de alertas de seguridad que notifique a los responsables de activos de información afectados. | Media |
| RF5 | Definir un criterio para priorizar alertas. | Baja |
| RF6 | Implementar un sistema de correlación de alertas de seguridad. | Media |
| RNF1 | La solución propuesta debe utilizar software libre. | Alta |
| RNF2 | La solución debe ser compatible con sistemas operativos abiertos y de tipo Unix. | Alta |
| RNF3 | La arquitectura de la solución debe ser escalable a demanda de la organización. | Media |
| RNF4 | Se requiere un despliegue automatizado de la solución. | Baja |
| Cuadro 8 | | |

En el Cuadro [9] se muestran todas las soluciones disponibles y cuales son los requerimientos que estas cumplen. Este cuadro fue elaborado con las codificaciones extraídas de los Cuadros [6] y [7]. El resultado permitió valorar la viabilidad de cada solución en términos de cumplimiento de requerimientos y contribuyo a una preselección inicial de soluciones.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Solución | RF1 | RF2 | RF3 | RF4 | RF5 | RF6 | RNF 1 | RNF2 | RNF 3 | RNF4 |
| S1 | Si | Si | Si | Si | Si | Si | Si | Si | Si | Si |
| S2 | Si | Si | Si | Si | No | Si | Si | Si | Si | Si |
| S3 | SI | No | Si | No | No | Si | Si | Si | No | Si |
| S4 | Si | Si | Si | Si | Si | Si | Si | Si | Si | Si |
| S5 | Si | Si | Si | Si | Si | Si | No | Si | Si | Si |
| S6 | Si | Si | Si | Si | Si | Si | No | Si | Si | Si |
| S7 | Si | Si | Si | Si | Si | Si | No | Si | Si | Si |
| S8 | Si | Si | Si | Si | Si | Si | No | Si | Si | Si |
| S9 | Si | Si | Si | Si | Si | Si | No | Si | Si | Si |
| S10 | Si | Si | Si | Si | Si | Si | No | Si | Si | Si |
| S11 | Si | Si | Si | Si | Si | Si | No | Si | Si | Si |
| S12 | Si | Si | Si | Si | Si | Si | No | Si | Si | Si |
| S13 | Si | Si | Si | Si | Si | Si | No | Si | Si | Si |
| Cuadro 9 | | | | | | | | | | |

Considerando al RNF 1 como un requerimiento de alta prioridad y fundamental para el desarrollo de este proyecto, se procedió a descartar las soluciones comerciales por más que cumplieran con los demás requerimientos de alta prioridad. Esto llevó a continuar el análisis de riesgos teniendo en cuenta solamente las soluciones gratuitas y de código abierto.

Definidos los niveles de requerimi entos de las soluciones, se plantea un sistema de puntuación, donde se asignan valores según la prioridad. El resultado se observa en el Cuadro [10].

|  |  |
| --- | --- |
| Tipo de prioridad | Valoración |
| Alta | 3 |
| Media | 2 |
| Baja | 1 |
| Cuadro 10 | |

Se utilizó un sistema de valoración para contrastar las soluciones preseleccionadas y elegir la más adecuada para este proyecto.

El proceso consistió en comparar los datos sobre el cumplimiento o no de los requerimientos, que fueron extraídos del Cuadro [9]. Posteriormente se asignó un puntaje mediante un sistema de valoración. Este último consistió en multiplicar el valor de cada prioridad por uno (1) si el requerimiento era satisfecho por la solución, en caso contrario el valor asignado fue cero (0). Finalmente los valores fueron sumados y se expuso el resultado final en el Cuadro [11].

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Solución | RF1 | RF2 | RF3 | RF4 | RF5 | RF6 | RNF1 | RNF2 | RNF3 | RNF4 | Total |
| S1 | 1x3 | 1x2 | 1x3 | 1x2 | 1x1 | 1x2 | 1x3 | 1x3 | 1x2 | 1x1 | 22 |
| S2 | 1x3 | 1x2 | 1x3 | 1x2 | 0x1 | 1x2 | 1x3 | 1x3 | 1x2 | 1x1 | 21 |
| S3 | 1x3 | 0x2 | 1x3 | 0x2 | 0x1 | 1x2 | 1x3 | 1x3 | 0x2 | 1x1 | 15 |
| S4 | 1x3 | 1x2 | 1x3 | 1x2 | 1x1 | 1x2 | 1x3 | 1x3 | 1x2 | 1x1 | 22 |
| Cuadro 11 | | | | | | | | | | | |

Se pudo observar que las soluciones mejor puntuadas fueron S1, S2 y S4, que corresponden a Greylog, Elastic (ELK) Stack y Security Onion respectivamente.

Como Graylog y Security Onion utilizan parte o todos los componentes de Elastic dentro de un entorno ya configurado, se procedió a descartar como posible solución a la pila Elastic (S2).

La selección final se realizó mediante una comparativa entre Security Onion y Graylog. Si bien en el Cuadro [11] se pudo apreciar que ambos lograron la misma valoración total, el elegido para llevar a cabo el proyecto fue Security Onion. Esto se debió a las ventajas que presenta Security Onion por sobre todas las soluciones participantes, de la que se destaca el polimorfismo a nivel de plataforma. Esta última característica permite que Security Onion asuma funciones propias de un SIEM como de un sistema IDS.

Dado que este proyecto fue realizado en conjunto con otras dos tesis de grado, donde una de ellas implicó el despliegue y configuración de sensores IDS, disponer de una solución que satisficiera los dos proyectos y garantizará una compatibilidad total, fue determinante para su elección.

## Síntesis

En primer lugar, se seleccionó Security Onion como plataforma para este proyecto. Posteriormente se eligió un sistema operativo libre, tipo Unix, compatible con la solución escogida. En segundo lugar, se adaptó la solución para recolectar y almacenar los datos pertinentes a los incidentes de seguridad que ocurren en la red corporativa, así como la información de contexto de los activos de información que se ven afectados. Se tuvo en cuenta el desarrollo de una arquitectura de despliegue que contemplase la escalabilidad horizontal de la solución para adaptarse a las necesidades de la organización.

En tercer lugar, luego de haber recibido y almacenado los datos, se configuró la solución para visualizar la información disponible. Esto permitió implementar un sistema de envío de alertas para notificar a los responsables de los activos de información que se viesen comprometidos. En consecuencia, se dispuso de un sistema que permitió correlacionar y filtrar alertas, en base a políticas que en su definición incluyen las categorías de eventos y sus prioridades asociadas.

Finalmente se procedió a automatizar el proceso completo que comprende la implementación, despliegue y configuración de la solución, junto a sus componentes asociados, mediante el uso de herramientas de automatización de tecnologías de la información.

# 

# 

# Iteración 1: “Elección de la solución”

## Security Onion como sistema de gestión de eventos

|  |
| --- |
|  |
| Figura 15: logo de Security Onion[38] |

La elección de Security Onion como plataforma se justificó en su naturaleza de código abierto y por sus características destacables respecto de otras soluciones libres, como el soporte de una activa comunidad, el desarrollo continuo de mejoras, actualizaciones y correcciones, su capacidad polimórfica y funcional de actuar como IDS, plataforma SIEM o cluster de almacenamiento. Esto permitió desarrollar distintas arquitecturas de una manera fácil y asistida para el despliegue y la consiguiente optimización de los recursos de hardware y de red.

Otras de las propiedades destacables es la capacidad de integración directa con un conjunto casi universal de los sistemas IDS disponibles, tanto libres como propietarios. Security Onion también incluye un paquete de configuraciones iniciales predefinidas para la infraestructura inicial del sistema, tales como el almacenamiento, normalización y gestión de logs (pila Elastic), los sistemas IDS y de gestión de usuarios, entre otros.

## Arquitectura del sistema de gestión de eventos

### Arquitectura de alto nivel

|  |
| --- |
|  |
| Figura 16: Security Onion: Arquitectura de alto nivel |

En la Figura 16 se observa la distribución de Security Onion y el flujo de datos entre sus componentes principales (la pila Elastic) y secundarios (Curator[43], ElastAlert[44], freqServer[45] y domainStats[46]). Se puede apreciar la conexión con los sistemas de detección IDS como Bro[47], Snort[48], Suricata[49], Syslog, etc. Se distinguen también los enlaces con los puntos de administración de los analistas del CSIRT y con los servicios web externos para el envío y recepción de alertas, notificaciones, análisis de tráfico, entre otros. Un punto a destacar es que la pila Elastic se encuentra desplegada en contenedores Docker[50].

### Arquitectura de despliegue

Security Onion sigue un modelo cliente - servidor y admite múltiples arquitecturas de despliegue, recomendando tres modelos generales: monolítico, densamente distribuido y distribuido. El servidor central es denominado ”nodo master” y los clientes son llamados “nodos Forward” o “sensores”, si se trata de nodos con una versión de Security Onion especialmente configurada para cumplir el rol de un sistema IDS. A continuación se detallan los tipos de nodos para posteriormente describir las distintas arquitecturas que los implementan.

#### Tipo de Nodos

* Nodo Master: este nodo ejecuta su propia copia de la base de datos Elasticsearch, con la que gestiona las búsquedas a través del cluster y estructura a otros nodos en el momento de su despliegue. Lo anterior implica que puede realizar las configuraciones necesarias para los nodos de los tipos “densos” y los de almacenamiento, pero no los de sensores o Forward, por carecer estos últimos de una pila Elastic. Este nodo permite a un analista conectarse mediante un enlace de supervisión para realizar consultas de los datos.
  + Este nodo contiene los siguientes componentes:
    - Elasticsearch[32]
    - Logstash[32]
    - Kibana[32]
    - Curator[43]
    - ElastAlert[44]
    - Redis[51]
    - Wazuh[52] / OSSEC[53]
    - Sguild[54]

Elasticsearch[32], Kibana[32] y Logstash[32] son componentes de la pila Elastic, que se tratarán en la siguiente sección junto a ElastAlert[44]. El objetivo de Curator[43] y Redis[51] es administrar y optimizar las bases de datos de los nodos de almacenamiento; Wazuh[52] es un IDS y Security Onion lo utiliza para el monitoreo de sí mismo, configurando un sistema HIDS ad hoc, aunque es posible desplegarlo en otros nodos o puntos de interés. Sguild[54] permite consultar eventos de una base de datos MySQL desde dentro de Security Onion y muestra los resultados en una GUI. Además, actúa como intermediario de otros componentes secundarios como Squert[55], del que detallaremos sus funciones y comportamiento en una sección posterior.

* Nodos Forward: este nodo cumple la función de procesar el tráfico y reenviar los resultados al nodo master. Los logs generados por Snort / Suricata y Bro son enviados mediante syslog a Logstash en el nodo master, utilizando un túnel ssh, donde finalmente son guardados en la base de datos Elasticsearch,

donde pueden ser reenviados a los nodos de almacenamiento.  
Los logs pueden ser consultados a través de una búsqueda en el cluster.

* + Los componentes de un nodo Forward son:
    - Zeek[47] (sucesor de Bro)
    - Snort[48] / Suricata[49]
    - Netsniff-ng[56]
    - Wazuh[52] / OSSEC[53]
    - Syslog-ng[57]

Zeek, Snort / Suricata y Netsniff-ng son procesadores de tráfico (IDS), donde Snort y Suricata serán tratados en una sección posterior. Syslog-ng es utilizado para recolectar logs de los IDS y enviarlos al Logstash del master, donde serán procesados y tratados antes de ser escritos en Elasticsearch.

* Nodos Pesados: Es un nodo híbrido entre el nodo Forward y el nodo Master, que incluye todos los componentes del nodo Forward, además de una instancia completa de la pila Elastic. Los nodos pesados envían los resultados de las consultas de su instancia local de Elasticsearch a las solicitudes realizadas por el nodo master mediante un túnel de autossh.
  + Los componentes de este nodo son:
    - Elasticsearch
    - Logstash
    - Curator
    - Zeek
    - Snort / Suricata
    - Netsniff-ng
    - Wazuh / OSSEC
    - Syslog-ng (envía los logs a la instancia local de Logstash)

* Nodos de almacenamiento: su objetivo es extender las capacidades de almacenamiento y procesamiento del nodo master. Estos nodos despliegan una instancia local de la pila Elastic. De manera análoga a los nodos pesados, cuando se realiza una consulta por parte de la instancia Elasticsearch del nodo master, esta es procesada por la instancia local del nodo de almacenamiento y devuelta por un túnel autossh.  
  + Los componentes del nodo de almacenamiento son:
    - Elasticsearch
    - Logstash
    - Curator
    - Wazuh / OSSEC

#### Tipos de Arquitectura

La versatilidad de disponer de múltiples arquitecturas permite adaptar la plataforma a las necesidades de la organización en la que se implante. A continuación, se describen cada una de las opciones posibles:

* Arquitectura monolítica: Consiste en un único servidor que ejecuta simultáneamente los componentes centrales o propios de un nodo master y los de un nodo sensor. Es un modo híbrido y concentrado que no se recomienda para enlaces de red de alto rendimiento por los elevados requerimientos de hardware necesarios.   
  Este tipo de arquitectura se recomienda para propósitos de pruebas en laboratorio y en entornos de baja demanda de tráfico de red.

|  |
| --- |
|  |
| Figura 17: Arquitectura monolítica de Security Onion |

* Arquitectura densamente distribuida: consiste en uno o más nodos pesados conectados a un nodo master. Solo se recomienda en el caso de que no sea posible desplegar una arquitectura distribuida, ya que tiene las mismas deficiencias de rendimiento de la arquitectura monolítica y no es apropiado para entornos de producción y/o enlaces de red de alta velocidad.

|  |
| --- |
|  |
| Figura 18: Arquitectura densamente distribuida de Security Onion |

* Arquitectura Distribuida: consiste en un servidor master, uno o más nodos Forward y uno o más nodos de almacenamiento. Es el tipo de despliegue recomendado en términos de eficiencia de requerimientos de hardware, balance de la carga y almacenamiento de datos y optimización general de los recursos disponibles en la organización.

|  |
| --- |
|  |
| Figura 19: arquitectura distribuida de Security Onion |

### Recibiendo, procesando y visualizando eventos: La pila Elastic

En la Figura 9 se observa, que luego de recolectar los datos provenientes de múltiples fuentes, es necesario normalizarlos y agregarlos a la base de datos; estas tareas son llevadas a cabo por los componentes de la pila Elastic, en este caso Logstash y Elasticsearch, respectivamente.

|  |
| --- |
|  |
| Figura 18: Conexión de componentes Elastic |

#### Logstash

Logstash es una tubería (en adelante “pipeline”) de procesamiento de datos gratuito y abierto del lado del servidor que ingesta datos de una multitud de fuentes, los transforma y luego los envía a su destino. Las fuentes de entrada admitidas por logstash son extremadamente amplias, como por ejemplo: syslog, STDIN, TCP, UDP, SNMP, IMAP, entre otras. Posteriormente, Logstash toma los datos sin estructura y los normaliza para crear conjunto ordenado mediante la identificación y conversión de la información a un formato común. Para realizar la tarea anterior, dispone de una gran variedad de filtros que facilitan el procesamiento general, independientemente de la fuente de datos. En este proyecto se utilizó a grok[62] como filtro de las fuentes de información. Con los datos ya normalizados, es posible darles un formato específico para un destino en particular, ya que Logstash admite múltiples destinos para la etapa final del pipeline; desde una base de datos, archivos finales o servicios web. Security Onion, por defecto, almacena estos datos normalizados en un formato JSON en la misma pila Elastic, es decir la base de datos Elasticsearch.

#### Elasticsearch

Elasticsearch es una base de datos del tipo NoSQL distribuida y orientada al almacenamiento de documentos. Los datos normalizados provenientes de Logstash son documentos almacenados en índices en Elasticsearch. Cada índice está compuesto por uno o más shards (fragmento), por lo tanto un shard es un subconjunto de documentos, siendo el elemento básico de Elasticsearch y el que permite la escalabilidad del mismo. Un shard es también una instancia de un “índice de Lucene[63]'', que indexa y almacena un documento en un segmento. Lucene es una librería desarrollada en Java para hacer búsquedas en una base de datos, constituyéndose en un motor de búsqueda que indexa y administra consultas en un conjunto de segmentos. La Figura 21 muestra la arquitectura de alto nivel del almacenamiento en Elasticsearch.

|  |
| --- |
|  |
| Figura 21: Arquitectura de almacenamiento en Elasticsearch |

El índice y segmento correspondiente a un documento al momento de ser almacenado, corresponde a un identificador basado en el hash del contenido del propio documento a guardar. Elasticsearch dispone de una REST API con los métodos para administrar los documentos utilizando el identificador de estos. Otro aspecto a considerar en términos de rendimiento de Elasticsearch sobre el hardware de su host, es el tamaño de los shards. Estos se pueden definir en las configuraciones de Elasticsearch y su importancia radica en que un tamaño demasiado pequeño de los shards provocará un uso ineficiente del hardware, ya que estos deberán ser sometidos a un “merge” con una frecuencia mucho mayor a la habitual. Por el contrario, un tamaño demasiado grande de los shards demandarán tiempos demasiado largos de recuperación del cluster de Elasticsearch en caso de algún inconveniente.

#### Kibana

Todos los datos almacenados en Elasticsearch pueden ser visualizados por Kibana, una interfaz gráfica perteneciente a la pila Elastic. Kibana permite visualizar los datos en gráficos circulares, de barras, histogramas, etc e interactuar con ellos; también es posible realizar análisis de ubicación cuando se disponen de los metadatos correspondientes mediante el complemento Elastic Maps, realizar análisis de series temporales de una manera rápida y sencilla, dispone de herramientas de inteligencia artificial, que mediante aprendizaje no supervisado permite detectar anomalías y patrones mediante las proyecciones sobre los datos.

Otra de sus características es la posibilidad de realizar gráficos de correlación y entrecruzamiento, seleccionando campos de interés y filtros lógicos creados por el usuario. Es de destacar que para algunas de estas características es necesario la instalación de plugins complementarios y aunque en su inmensa mayoría son gratuitos, algunos pueden ser pagos ya que utilizan servicios web de la nube de los desarrolladores.

|  |
| --- |
|  |
| Figura 22: Captura de pantalla de Kibana |

### Elastic, ElastAlert, TheHive y Cortex

Security Onion incluye la pila Elastic, cuyos componentes son la base de datos Elasticsearch, Logstash quien se encarga de recibir, procesar, normalizar y agregar los datos resultantes a la base de datos; estos datos son visibles mediante Kibana. El proceso comienza cuando Logstash recibe los datos sin procesar provenientes de múltiples fuentes, son normalizados por este componente y enviados a Elasticsearch para su almacenamiento. Kibana permite consultar la base de datos mediante una interfaz gráfica de usuario y utilizar esa información para propósitos de análisis de amenazas.

ElastAlert es un framework que permite identificar y alertar sobre eventos anómalos o patrones de interés sobre los datos de Elasticsearch. También provee múltiples mecanismos para enviar alertas mediante distintas plataformas externas, tales como Slack[58], correo electrónico, JIRA[59], Telegram[60] y muchos más. Tanto ElastAlert como los componentes de la pila Elastic están desplegados sobre contenedores Docker[50].

Es destacable que, aunque Security Onion cubre gran parte de los requerimientos de un SIEM, no posee los elementos que permiten completar un sistema de manejo y respuesta a incidentes; por esta razón y luego de una investigación sobre las alternativas posibles, se incluyó a TheHive[61] y Cortex[61] como complemento de Security Onion. TheHive permite la gestión de incidentes de manera detallada y la colaboración con otros CSIRT mediante el uso compartido de información sobre incidentes en tiempo real; mientras que Cortex hace posible la automatización de las respuestas y operaciones ante incidentes utilizando los datos enviados por TheHive.

### Analizando y clasificando eventos: ElastAlert

A pesar de que Kibana permite consultar los datos almacenados en Elasticsearch y presentarlos de diversas maneras que resultan en una gran utilidad, carece de la capacidad de generar alertas cuando los datos coinciden con algun patron, especialmente cuando estos datos son escritos y consultados en tiempo real en la base de datos. Con este objetivo, la plataforma integra a ElastAlert, siendo un componente confiable, modular y simple de configurar. Su funcionamiento se basa en dos componentes principales: reglas y alertas; las primeras son utilizadas para comparar con los datos resultantes de las consultas que se hacen en forma constante a Elasticsearch, esta comparación consiste en hallar patrones o firmas definidas en las reglas dentro de los datos obtenidos de la consulta; si el resultado de la búsqueda es positivo, una alerta es disparada para notificar el evento.

Las alertas consisten en mensajes que permiten notificar a otro sistema con el objetivo de que este último realice una acción sobre las causas del evento que detectó la regla o bien informar a los analistas y/o responsables definidos. En cualquiera de los dos casos, las alertas pueden incluir toda la información recabada en un formato definido, tales como plantillas o cualquier arreglo configurado a tal fin.

Según la naturaleza de los eventos a clasificar, las reglas cuentan con un conjunto común de paradigmas de monitoreo, estos permiten identificar y generar alertas aprovechando las características de las anomalías al mismo tiempo que optimizan los recursos del resto del CSIRT en términos de hardware y atención de los analistas. Algunos de estos paradigmas se basan en el comportamiento, tales como la frecuencia que consiste en generar una alerta cuando se detectan N cantidad de eventos en un intervalo definido, el cambio de tasas de ocurrencia por arriba o abajo de un límite establecido como normal para un determinado tipo de eventos, cuando en los datos se encuentran presente campos que han sido previamente establecidos como parte de una lista blanca, negra u algún campo cuyo valor coincida con otros tipos de filtros, entre otros. Es posible definir y configurar tantas reglas como alertas sean necesarias.

### El panel de control general: TheHive y Cortex

Como se mencionó en las secciones anteriores, Security Onion requiere de otros elementos capaces de realizar la gestión integral de incidentes y sus respuestas, elementos que sean capaces de condensar y presentar información a los analistas del CSIRT encargados de monitorear y responder a las anomalías e incidentes detectados. TheHive es la herramienta que se eligió para esta tarea ya que es una plataforma de respuesta a incidentes de seguridad gratuita y de código abierto, cumpliendo así con uno de los requerimientos no funcionales del proyecto, referido al tipo de licencia y accesibilidad al código. Otra de las razones para la elección de esta plataforma en particular ha sido su escalabilidad y su integración con MISP, lo que permite compartir información sobre las amenazas detectadas con otros CSIRT de organizaciones aliadas. Las tres capacidades centrales son la elaboración de casos, la respuesta a estos y la anteriormente mencionada colaboración con otros CSIRT.

|  |
| --- |
|  |
| Figura 23: Alertas recibidas en el panel de TheHive |

En cuanto a la elaboración de casos y tareas asociadas, estas se crean en base a las alertas recibidas (Figura 21), donde el primer paso consiste en la creación de un caso para luego asociar este a una o varias de las alertas presentes utilizando la plantilla disponible (Figura 22), posteriormente es posible agregar tareas asociadas al caso, las cuales se pueden asignar a distintos analistas; a continuación es posible sumar métricas y campos personalizados, reducir el tiempo de búsqueda y recopilación de datos así como automatizar algunas tareas de recopilación de antecedentes en el manejo de incidentes mediante el uso del tablero (dashboard) dinámico, tal como se observa en la Figura 23. En el proceso de creación del caso, thehive permite agregar cualquier otra información que se considere relevante, como etiquetas, archivos sospechosos de contener malware, etc a modo de evidencias.

|  |
| --- |
|  |
| Figura 24: Plantilla para la creación de nuevos casos |

|  |
| --- |
|  |
| Figura 25: Dashboard dinámico de TheHive |

Luego de la creación de un caso, es posible sumarle todos los “observables” que sean necesarios, donde los observables son todos aquellos campos que se pueden agregar de forma manual y que constituyen fuentes de información para analizar cada caso. Una vez configurado un caso, estos son examinados por scripts llamados “analyzers” que correlacionan y filtran los datos del caso contra los provistos por otras instancias MIPS u otras fuentes de información como la propia base de datos local, servicios de resolución DNS, plataformas como Shodan[64], VirusTotal[65], Google Cloud Visión[66], entre muchas otras. Los observables también se pueden obtener por datos de las alertas recibidas, los cuales son previamente configurados en ElastAlert. Como se mencionó anteriormente ElastAlert realiza consultas a Elasticsearch y con los resultados busca patrones de interés para realizar una notificación, obtenida esta última extrae datos que se consideran de interés para ser enviados a The Hive. Un ejemplo de esto puede ser un número de IP, tipo de protocolo, fecha que se generó el log, puerto de origen y/o destino. La alerta que llega a The Hive contiene todos estos datos, considerados observables.

Luego de que el caso fue creado o sobre la misma alerta, el analista puede dar curso a una respuesta mediante “responders” que son scripts en los cuales se encuentra la respuesta del CSIRT a la amenaza. Tanto los responders como los analyzers se encuentran bajo la responsabilidad de Cortex, el subsistema encargado de procesar los casos de TheHive. Al final de esta sección, se presentan los diagramas de casos de uso correspondientes a TheHive y Cortex.

|  |
| --- |
|  |
| Figura 26: Algunos de los analyzers disponibles en Cortex |

|  |
| --- |
|  |
| Figura 27: Ejemplos de responders utilizables en Cortex |

|  |
| --- |
|  |
| Figura 28: Casos de uso de gestión de usuario y configuraciones de TheHive |

|  |
| --- |
|  |
| Figura 29: Diagrama de casos de uso de alertas y casos |

|  |
| --- |
|  |
| Figura 30: Casos de uso de gestión de usuario y configuraciones de Cortex |

|  |
| --- |
|  |
| Figura 31: Casos de uso del tablero de Cortex |

### 

### Automatizando acciones: TheHiveHooks

Debido a que ciertos eventos tienen una elevada frecuencia de repetición y además su respuesta está perfectamente definida, sería de gran utilidad en estos tipos de ocasiones relevar a los analistas de la tarea que comprende tomar una alerta, elaborar un caso, ejecutar los analyzers y luego activar la respuesta correspondiente mediante los responders. Con este objetivo se implementó una automatización del proceso utilizando webhooks, que en el caso de TheHive se trata de una aplicación web (TheHiveHooks) que expone una REST API receptora de todos los cambios en TheHive y los envía hacia un punto final HTTP, donde un programa (en este caso Cortex) consume los datos recibidos. De esta manera, TheHive envía cada acción realizada en el (crear caso, actualizar caso, agregar tarea) vía los webhooks hacia el punto donde Cortex estará escuchando para actuar en consecuencia. Para esto fue necesario declarar la API mencionada anteriormente en el archivo de configuración *application.conf* de TheHive y luego desarrollar los *handlers* de los eventos para activar los responders apropiados.

La importancia de automatizar respuestas a ciertos tipos de incidentes no radica solo en optimizar el tiempo y la capacidad de atención de los analistas humanos en tanto en cuanto un recurso escaso y valioso, sino fundamentalmente en la capacidad de responder con eficacia y eficiencia a determinados arquetipos de ataques potencialmente devastadores como (por ejemplo) un DDoS, donde miles de eventos (incidentes) se producen en simultáneo y la capacidad humana de responder eficazmente a esta avalancha utilizando el protocolo descrito con anterioridad, es prácticamente nula. A continuación se presenta un diagrama simplificado de secuencia que involucra TheHiveHooks con TheHive y Cortex en la respuesta a un evento particular y la creación de un caso:

|  |
| --- |
|  |
| Figura 32: Diagrama simplificado de secuencia del proceso de respuesta con TheHiveHooks y promover alertas a casos |

## Integración con los sistemas de detección

En secciones anteriores se mencionó que Security Onion cuenta con componentes para realizar tanto monitoreo de red (NIDS) como monitoreo de puntos finales (HIDS). Durante la configuración inicial del sistema se pueden especificar los NIDS a utilizar, para una configuración rápida de los sensores. Esto permite realizar una primera integración con el hardware disponible

### Suricata, Snort y Ossec

Suricata y Snort son motores de detección de amenazas en el tráfico de red. Ambos NIDS se basan en firmas o reglas para realizar la detección de amenazas, estas firmas son actualizadas constantemente conforme a la aparición de nuevos tipos de ataques, exploits y malware. Si bien estos NIDS son gratuitos y de código abierto Snort ofrece la versión paga, la cual cuenta con soporte para descargar las firmas actualizadas a la fecha. Por defecto Snort cuenta con las reglas básicas para la detección de amenazas bien conocidas.   
 Suricata, por otro lado, es desarrollado y mantenido por los colaboradores de la OISF, los cuales también dan soporte a las firmas ya que se actualizan las existentes y se agregan nuevas en forma permanente. Estas actualizaciones en las reglas son descargadas periódicamente mediante PulledPork, una utilidad que también es usada por Snort cuyo fin es descargar reglas y firmas desde distintos centros de investigación reconocidos en todo el mundo, como el SANS institute, Emerging Threats, entre otros.

A continuación se muestran las diferencias entre Snort y Suricata en una tabla comparativa:

|  |  |  |
| --- | --- | --- |
|  | Snort | Suricata |
| Desarrollador | CISCO | Open Information Security Foundation (OISF) |
| Lanzamiento | 1998 | 2009 |
| Lenguaje del código | C | C |
| Sistema operativo | Linux, Windows y Mac OS X | Linux, Windows y Mac OS X |
| Hilos | Monohilo | Soporte múltiples hilos |
| Soporte IPv6 | Si | Si |
| Reglas de Snort | Si | Si |
| Reglas de Emerging Threats | Si | Si |
| Formato de logs | unified2 | unified2 |
| Compatible con Aanval | Si | Si |
| Tabla 4: Comparación entre Snort y Suricata | | |

Por otro lado está Ossec (actualmente llamado Wazuh), que es un IDS orientado a hosts (HIDS). Al igual que los NIDS anteriores está basado en firmas para la detección de amenazas, además también es gratuito y de código abierto. Las reglas pueden descargarse del repositorio disponible en github.

## Arquitectura del despliegue

|  |
| --- |
|  |
| Figura 33: Arquitectura de Despliegue |

En la imagen superior (Figura 33) se muestra la arquitectura de despliegue del proyecto. La descripción, de izquierda a derecha, es: el proveedor ISP de conexión a internet y por consiguiente al exterior de la organización, el switch de capa 3 al que están conectadas las dependencias cuyos enlaces fueron seleccionados para ser monitoreados para este proyecto, los nodos Forward de Security Onion y un switch de la red interna del CSIRT. Se observa que los enlaces “Dependencia 1 - switch capa 3” y el de “switch capa 3 - nodo Forward de Security Onion Dependencia 1” tienen el mismo color; esto se debe a motivos de representar el hecho de que el switch capa 3 fue configurado para reenviar el tráfico entre el enlace de este y la dependencia 1 hacia el nodo Forward mencionado. Una situación análoga ocurre entre la Dependencia 2 y el nodo Security Onion Forward Dependencia 2.

El último eslabón de la conexión, el switch de capa 2, es el encargado de la red interna del CSIRT. A él se encuentran conectados las computadoras de los analistas y el nodo Master de Security Onion, los nodos Forward anteriormente mencionados y el servidor que aloja a TheHive y Cortex. Finalmente, los analistas pueden consultar y administrar los servidores correspondientes a los nodos Master y Forward de Security Onion así como al servidor que contiene a TheHive y Cortex.

# Iteración 2: “Configuración y despliegue en un ambiente de prueba”

En este proyecto se desarrolló sobre un ambiente de prueba primero y de producción después, sobre un servidor central y un sistema operativo de virtualización sobre el que se crearon un conjunto de máquinas virtuales, cada una alojando un servidor con nodos Forward, Master y el correspondiente a TheHive - Cortex. Se utilizó de guia los componentes, software y arquitectura de conexión entre ellos, mencionados en las iteraciones precedentes.

## Configuración del ambiente de prueba

El primer paso consistió en examinar los requisitos de hardware mínimos y recomendados por cada uno de los fabricantes de los sistemas y subsistemas elegidos, al mismo tiempo que se analizaron, por un lado, las demandas de tráfico de red en el ambiente de prueba y por el otro los requerimientos sobre los datos y capacidades que se esperan obtener del proyecto. Se procedió a realizar un diagrama topológico en la infraestructura objetivo, con esta información y los datos anteriormente mencionados, se realizó una estimación del hardware necesario para el servidor central que albergó las correspondientes máquinas virtuales de este proyecto.

Según el fabricante, los requerimientos de hardware para un entorno de producción con enlaces de 1 Gbps son los que figuran en el Cuadro [5].

|  |  |  |  |
| --- | --- | --- | --- |
| Requerimiento | Nodo Master | Nodo Forward | TheHive y Cortex |
| Cantidad de CPU - Arquitectura | Mínimo de 8 núcleos de vCPU | Mínimo de 12 núcleos de vCPU | Mínimo de 8 núcleos de vCPU |
| Memoria RAM | 12 a 128 GB | 128 a 256 GB | A partir de 8 GB |
| Almacenamiento necesario | Mínimo de 1 TB | Mínimo de 540 GB por dia (PCAPS) | A partir de 60 GB |
| Tabla 5: requerimientos de hardware recomendados por el fabricante para el monitoreo de un enlace de 1 Gbps | | | |

Debido a las restricciones en la disponibilidad del hardware, se realizó una implementación aproximada, teniendo en cuenta la optimización al máximo de los recursos disponibles.

Los nuevos requerimientos de hardware que se utilizaron se incluyen en el Cuadro [6].

|  |  |  |  |
| --- | --- | --- | --- |
| Requerimiento | Nodo Master | Nodo Forward | TheHive y Cortex |
| Cantidad de CPU - Arquitectura | 8 nucleos vCPU - x86-64 | 10 nucleos vCPU - x86-64 | 8 nucleos vCPU - x86-64 |
| Memoria RAM | 16 GB | 32 GB | 8 GB |
| Almacenamiento necesario | 500 GB | 200 GB | 60 GB |
| Tabla 6: Requerimientos de hardware según el tipo de nodo desplegado. Los requerimientos varían según el tipo de enlace a monitorear | | | |

En cuanto a los núcleos de CPU virtuales, su frecuencia era de 2.4 Ghz, basados en procesadores físicos Intel Xeon E5620. Las memorias RAM pertenecían a la tecnología DDR3 y su frecuencia de refresco de 2400 MHz.

Los discos eran del tipo mecánico con una velocidad de transferencia para lectura de 196 MB/s y para escritura de 154 MB/s, con interfaz SATA III y 7200 rpm de velocidad de rotación.

En este proyecto se monitorearon enlaces de 1 Gbps de ancho de banda, por lo tanto los requerimientos para el nodo forward se incrementaron a 10 núcleos vCPU y 32 GB de memoria RAM.

Siguiendo el diagrama de la arquitectura de despliegue de la sección anterior, se optimizó al máximo el uso de los recursos del servidor disponible para permitir el despliegue de cuatro nodos: dos Forward y un Master de Security Onion, así como un cuarto conteniendo a TheHive y Cortex.

### Configuración del entorno de virtualización

Para el entorno de virtualización se utilizó VMWare[67], concretamente la suite vSphere HyperVisor v6.7.0 u3. Este sistema operativo basado en Unix permite gestionar los recursos de hardware disponibles, almacenar imágenes de distintos sistemas operativos y crear máquinas virtuales con estos últimos. Durante el proceso de creación de una máquina virtual, se selecciona el sistema operativo deseado y es posible asignar distintas cantidades de memoria principal, secundaria, cantidad de vCPU, número y tipo de enlaces de red, entre otros parámetros.

|  |
| --- |
|  |
| Figura 34: Diagrama de una máquina virtual desde el punto de vista de un HyperVisor |

### Definición y configuración de las redes a observar

Se decidió monitorear dos dependencias en base a un análisis del ancho de banda de las dependencias existentes, por lo tanto se seleccionaron las que mayor volumen de tráfico registraban en función de un registro histórico y mediciones propias realizadas a lo largo de una semana. Las dependencias seleccionadas tenían un enlace con ancho de banda de 1 Gbps cada una, con velocidades promedio consideradas como la suma entre entrada y salida, entre 11,95 y 47,24 Mbps respectivamente; con picos poco frecuentes de 300 Mbps de tráfico, que no se contemplaron en los requisitos de hardware, por lo tanto habrá una posible pérdida de paquetes en estos casos. Se realizó un “port mirroring” de los puertos del switch capa 3 a los que están conectados estas dependencias y se los conecto con los respectivos enlaces de monitoreo de sendos nodos Forward de Security Onion.

Los gráficos a continuación muestran el volumen del tráfico medido en dos periodos de tiempo distintos: Durante un día (exceptuando las horas en las que la actividad era mínima) y a lo largo de una semana. Si bien estos registros que se presentan a continuación corresponden a una sola de las dependencias, la restante tenía un comportamiento análogo.

|  |
| --- |
|  |
| Figura 35: Tráfico correspondiente a una dependencia, medido durante un día, obviando las horas donde este es casi nulo |

|  |
| --- |
|  |
| Figura 36: Tráfico medido durante el periodo correspondiente a una semana |

## Configuración inicial del sistema base

Es posible instalar Security Onion en su versión 16.04 de dos maneras, sea mediante una ISO provista por los desarrolladores o bien mediante una serie de paquetes en una distribución Ubuntu[68]. En este último caso será necesario contar con la distribución Ubuntu en su versión 16.04, ya que las distribuciones de Security Onion siguen a las distribuciones respectivas de Ubuntu; esto fue cierto hasta el año 2020 cuando se lanzaron nuevas versiones de Security Onion con soporte a otras distribuciones Linux: CentOS 7 y Ubuntu 18.04 y 20.04 aunque en el futuro se podrá desplegar en otros tipos de sistema Linux ya que desde la versión 2.x en adelante, el sistema se despliega en contenedores.

### Instalación y configuración de Security Onion

Como se mencionó en la sección anterior, existen dos maneras de instalar Security Onion: a partir de una imagen ISO o mediante paquetes / contenedores. Se eligió para este proyecto la segunda opción, el despliegue mediante paquetes de la distribución 16.04 de Security Onion ya que al momento del desarrollo de este trabajo integrador era la versión estable del sistema. Por consiguiente, se dispuso de un sistema operativo Ubuntu Server 16.04 con la particularidad de tener dos discos montados: el principal para el sistema operativo y el secundario para los datos recolectados en un directorio /nsm: índices en el caso de un servidor Master y capturas de paquetes o logs en el caso de un nodo Forward. Luego de finalizada la instalación de Security Onion, es necesario elegir el rol (Master o Forward) del nodo mediante el asistente y posteriormente realizar la configuración del mismo. Para esto último, se cuenta con la guia del asistente integrado que permite elegir y configurar las interfaces disponibles (observación o administración); en el caso de un nodo Forward, elegir el motor IDS (Snort o Suricata). El último paso consiste en elegir entre dos tipos de modo de funcionamiento: Producción o Evaluación.

|  |
| --- |
|  |
| Figura 37: Asistente de instalación de Security Onion 16.04 |

|  |
| --- |
|  |
| Figura 38: El asistente de instalación permite elegir el modo de despliegue |

Con el objetivo de cumplir uno de los requerimientos no funcionales del proyecto, que implica la automatización del despliegue (instalación y configuración) del sistema, se utilizó una herramienta de administración automatizada de servidores llamada Ansible en su versión 2.8.4 para la cual se desarrollaron scripts YAML conteniendo la secuencia de instalación de los paquetes, configuraciones, rol del nodo (Forward o Master) y librerías requeridas para el apropiado funcionamiento del sistema.

### Instalación y configuración de TheHive - Cortex

Para la instalación del gestor de incidentes, que tiene como componentes a TheHive y Cortex, se utilizó el sistema operativo Debian 10. En primer lugar se instaló TheHive, para ello fue necesario realizar la instalación previa de los componentes necesarios como las librerías de Java, Python y Elasticsearch; este último requirió una configuración en su archivo elasticsearch.yaml:

|  |
| --- |
|  |
| Figura 39: Configuración añadida a elasticsearch.yaml para la instalación de TheHive |

Finalmente, los últimos pasos para la instalación de TheHive consisten en habilitar e iniciar el servicio de elasticsearch, agregar el repositorio que contiene los paquetes de TheHive, instalarlo y luego habilitar el servicio para poder iniciarlo.

En cuanto a Cortex, el proceso es similar al anteriormente descrito para TheHive, donde una vez descargados e instalados los paquetes de Cortex con su correspondiente secuencia de habilitación e inicio; se procedió a descargar del repositorio los responders y analyzers respectivos. Por último, se modifica el archivo de configuración de Cortex para indicar la ubicación del directorio que contiene los responders y analyzers mencionados anteriormente.

Posteriormente se actualizó la base de datos elasticsearch mediante la GUI web de Cortex, se creó un superusuario y luego las organizaciones donde se administrarán usuarios comunes y analyzers; es necesario crear un usuario con el rol de administrador de organizaciones. Las organizaciones tendrán habilitados y configurados determinados responders y analyzers según sea necesario.

El último paso del proceso consiste en comunicar TheHive y Cortex entre sí. Para ello se genera una API key en Cortex que será usada como parte de las modificaciones necesarias al archivo application.conf de TheHive. Las modificaciones completas que se realizaron al mencionado archivo se pueden apreciar en la siguiente imagen:

|  |
| --- |
|  |
| Figura 40: Modificación al archivo application.conf de TheHive para la comunicación con Cortex |

## Configuración de acciones automáticas

El mecanismo elegido para automatizar las acciones fue mediante webhooks, para ello fue necesario implementar un entorno de virtualización en el mismo servidor donde se encuentran alojados TheHive y Cortex. Para ello se optó por utilizar el módulo de python “venv”, lo que requirió la instalación de Python 3.6 como primer paso. En segundo lugar se modificó el archivo application.conf de TheHive que se mencionó en la sección anterior para permitir la comunicación con el puerto del entorno de virtualización. En tercer lugar se verificó que en el nodo Master de Security Onion las reglas de ElastAlert tengan los campos necesarios configurados como observables ya que estos serán necesarios posteriormente dado que en TheHive están creados los observables que esperan esta información (source\_ip, destination\_ip, source\_port, destination\_port, alert, classification, category, etc).

Satisfechos los pasos anteriores, la instalación siguió los siguientes pasos:

1. Se creo una carpeta con el nombre webhooksenv
2. Un entorno virtual fue creado y activado en la mencionada carpeta
3. Se procedió a instalar las librerias necesarias: Flask, Gunicorn, Wheel, Request y Netaddr.
4. Se desactivo el entorno virtual y se habilitó en el firewall el puerto 5000
5. Se agregaron y modificaron valores al archivo de parámetros que utiliza webhooks.
6. Se inició y comprobó el estado del servicio.

Finalmente, se modificaron los archivos de configuración de TheHive y Cortex para actualizar la información necesaria referida a los webhooks.

# Iteración 3: “Reportes de incidentes y acciones automáticas”

En Security Onion y otros sistemas, el elemento descriptor que identifica y procesa a cada definición de incidente en particular es la regla. Las reglas comprenden una serie de campos que describen con precisión la naturaleza de un incidente dado y por lo tanto, existen tantas reglas como amenazas en circulación.

Cuando un nuevo malware es descubierto por el equipo de algún CSIRT con la capacidad de investigación suficiente o reportado a un laboratorio apropiado para este fin, es posible realizar un estudio de sus características y una vez identificadas estas últimas, proceder a crear una regla y agregarla al repositorio correspondiente para que otros CSIRT actualicen sus IDS con esta nueva definición y así contar con un filtro (la regla) que permita detectar este malware. Las reglas tienen un conjunto de campos donde se detallan características del paquete y su contexto, tales como el puerto de origen y destino, protocolo empleado, dirección IP, etc y unos campos dedicados a la naturaleza del incidente (clasificación, mensaje, prioridad, etc). Algunos de estos campos son comunes a todas las reglas y permiten agruparlas para administrar eficientemente las alertas generadas cuando una regla coincide con la descripción de un incidente. Dado que estos campos también se pueden considerar observables, es posible utilizarlos por TheHive y Cortex para automatizar respuestas.

## Análisis de prioridades de los incidentes

Como se indicó anteriormente, la estructura de las reglas consisten en dos partes bien definidas: un encabezado (header) que es obligatorio y un conjunto de campos opcionales. Dentro del header encontramos la acción (alerta, notificación, etc), el protocolo (tcp, udp), puertos de origen y destino, el sentido del evento (entrante o bidireccional) y las direcciones IP de origen y destino.

La segunda parte de las reglas incluye dos tipos de campos: los que describen la naturaleza del evento y aquellos que contienen información del paquete de datos. Dentro del primer grupo encontramos aquellos tales como msg (descripción del evento), sid (id de la firma), classtype (clasificación de reglas o alertas), priority (prioridad de la firma y/o alerta), target (especifica de qué lado está el objetivo, es decir puerto de origen y puerto de destino), entre otros. El segundo grupo contiene datos extraídos que provienen desde de la capa de red hasta la de aplicación de la pila OSI. Se pueden mencionar a los campos “GeoIP” (localización geográfica de la IP), “Fragbits” (presencia del bit de fragmentación), “ACK” (presencia del campo ACK en paquete TCP), “itype” (número del tipo de mensaje ICMP), “http.method” (tipo de método HTTP usado), entre otros.

|  |
| --- |
|  |
| Figura 41: Estructura general de una regla |

Como se describió en los párrafos precedentes, como los campos están presentes en todas las reglas, es posible hacer uso de algunos de ellos para agrupar reglas que describen amenazas pertenecientes a un mismo grupo o categoría de malware, intentos de intrusión, reconocimiento, escalado de privilegios, etc y por lo tanto son útiles para gestionar los incidentes.

Es posible configurar esta gestión a través de un archivo que relaciona campos como categorías de eventos con prioridades de la alerta generada. Este archivo llamado “classification.config” se encuentra bajo el directorio que almacena las reglas descargadas desde diversas fuentes; en particular relaciona los campos “classtype” con “priority”, de manera tal que cualquier regla cuyo campo classtype contenga a los descritos en este archivo, generará una alerta con prioridad definida también en este. De esta manera, es posible administrar un enorme número de reglas agrupadas en un reducido grupo de categorías y modificar el nivel de prioridad que tendrá en el sistema las alertas que generan.

El objetivo de asignar distintos niveles de prioridad a las alertas generadas por los eventos que sucedan radica en la naturaleza de los eventos, su importancia y la gestión de la atención de los analistas del CSIRT. Esto se debe a las necesidades de optimizar el uso de los recursos técnicos y humanos del centro de respuesta a incidentes para cumplir de la manera más eficiente posible con los objetivos y políticas de la organización a la cual pertenece. De esta manera, la naturaleza de los incidentes determina su elegibilidad para una respuesta automatizada al tener en cuenta por un lado su estructura bien conocida y por el otro su alta tasa de repetición en un periodo determinado. En estos casos, sería inutil destinar valiosos recursos como la atención de un analista ya que se conoce perfectamente la estructura del incidente y por lo tanto la respuesta apropiada o en aquellos casos en los que aún conocida su estructura, el incidente proviene en simultáneo de múltiples fuentes en muy poco tiempo, de manera que la capacidad humana de responder de a uno a la vez estaria tan sobrepasada que no sería efectiva. Estos son los casos de ataques de reconocimiento y los de denegación distribuida de servicio, entre otros.

De aproximadamente cuarenta y siete (47) categorías de incidentes disponibles por defecto, consideramos para el máximo nivel de prioridad a siete clasificaciones dado su nivel de ocurrencia y nivel de impacto para la organización.

* Web-application-attack: esta categoría engloba a un conjunto enorme de malware y ataques a nivel de capa de aplicación. Gusanos, ransomware, ataques de reconocimiento entre otras amenazas comparten esta categoría. Sobre el caso particular de los ataques de reconocimiento, se aplicaron filtros para separarlos de los demás ya mencionados.
* Unsuccessful User: intentos repetidos de ganar acceso en ciertos activos e infraestructura de la organización.
* Attempted-dos: intentos de ataque de denegación de servicio y su variante distribuida
* Known client side exploit attempt: intento de ejecución de exploits en el lado del cliente.
* Exploit Kit Activity Detected: detección de actividad de un kit de exploits
* A suspicious filename was detected: detección de nombres de archivos sospechosos
* Network Trojan: detección de un virus troyano de red.

## Automatización de acciones

La automatización de acciones puede dividirse en dos categorías: acciones activas ya sea sobre el incidente, las causas que lo provocan (vulnerabilidades, puertos, direcciones del atacante, etc) o bien mediante notificaciones de la detección a los analistas del CSIRT y/o los responsables designados (analistas de un NOC, administradores de redes, encargados o usuarios de los activos afectados, entre otros). Se eligió esta última categoría por considerar que era la más apropiada para notificar y generar el mayor impacto al cubrir gran parte de los medios de comunicación disponibles en la organización.

Cuando un incidente es reportado por ElastAlert a TheHive tal como se describió al final de la sección anterior, se inicia una serie de pasos en la cual intervienen webhooks y Cortex como se puede apreciar en la Figura 40:

|  |
| --- |
|  |
| Figura 42: Secuencia de respuesta automatizada |

# Iteración 4: “Testing del sistema”

## Test de detección de ataques varios

## Test de reportes de incidentes

## Test de acciones automáticas

# Conclusión

El empleo y consumo masivo de las tecnologías de la información, así como la convergencia e interconexión de redes y sistemas, ha generado nuevos tipos de riesgos y amenazas para las organizaciones. Los ataques han evolucionado en complejidad, sigilo y especialización de los objetivos, requiriendo mayores esfuerzos para la prevención, detección y mitigación de los incidentes. Esto produce que las organizaciones tengan necesidad de desplegar soluciones del tipo CSIRT.

El SIEM, como núcleo de las operaciones de un CSIRT, tiene una gran importancia en cuanto a la recolección de datos, su análisis y las decisiones tomadas en consecuencia.

La solución elegida, Security Onion, representa una excelente alternativa a las soluciones comerciales ya que sus capacidades permiten cumplir los objetivos de cualquier organización. Por su naturaleza de código abierto, puede ser adaptado a configuraciones muy específicas y su desarrollo continuo, lo que nos permitio configurar fácilmente el despliegue inicial, una integración sencilla con los sensores, una presentación intuitiva de las alertas e información contextual. Por otro lado, es remarcable su grado de integración con soluciones complementarias como TheHive y Cortex. Se destaca también la documentación, que por su grado de detalle, facilitó el desarrollo de nuestro proyecto.

Una de las características más sobresalientes de Security Onion es que se trata de un sistema operativo en sí mismo, con distintos grados de modularización que hacen posible desarrollar distintos tipos de arquitecturas según la situación requerida. Esto lo diferencia de otras plataformas y soluciones evaluadas que consisten en un software que necesita un sistema operativo base sobre el cual desplegarse, lo que condiciona su flexibilidad para abordar distintos requerimientos o introduce limitaciones al proyecto.

Finalmente nos agrado saber que existen herramientas libres y gratuitas que, si bien requieren un esfuerzo para implementarlas, están a la altura de las soluciones propietarias. Si bien a nivel global la mayoría de las organizaciones optan por soluciones comerciales, existen alternativas libres que permiten desarrollar centros de monitoreo a la medida de las necesidades de una organización sin depender de terceros.

# Futuros trabajos

* Estabilidad del sistema. Cuales son sus causas
* Responders y analyzers en Cortex
* Implementación de nodos de almacenamiento (nodos storage)
* Desarrollar acciones automáticas con thehive webhooks
* Exportar lo desarrollado en esta tesis a las nuevas versiones de security onion

# 

# 

# 

# Bibliografía

[1] Shenk, J., 2014 SANS Institute, Information Security Reading Room, Ninth Log Management Survey Report.

[2] Cichonski, P., Millar, T., Grance, T. and Scarfone, K., 2012. Computer Security Incident Handling Guide : Recommendations of the National Institute of Standards and Technology. pp.21-44.

[3] It-isac.org. (2020). Recuperado 15 de Junio de 2020, desde https://www.it-isac.org/.

[4] ENISA. (2009). Baseline Capabilities For National / Governmental Certs.

[4b] “Markets & Markets”. Mercado global de soluciones SIEM. https://www.marketsandmarkets.com/Market-Reports/security-information-event-management-market-183343191.html

[5] Dirección Nacional de Ciberseguridad. Argentina.gob.ar. (2020). Recuperado 15 de Junio de 2020, desde https://www.argentina.gob.ar/jefatura/innovacion-publica/direccion-nacional-ciberseguridad.

[6]Jefatura de Gabinete de Ministros. (1998). Decreto 856/98 Creación de la Subsecretaría de Tecnologías Informáticas.

[7] Subsecretaría de Tecnologías Informáticas. (1999). Reglamentación de las funciones del ArCERT.

[8] Jefatura de Gabinete de Ministros. (2011). Créase El Programa Nacional De Infraestructuras Críticas De Información Y Ciberseguridad. Objetivos.

[9] Jefatura de Gabinete de Ministros, Oficina Nacional de Tecnologías de la Información. (2013). Disposición Nº 2/2013: Creación del ICIC - CERT.

[10] Universidad Nacional de Córdoba, Rectorado. (2014). Resolución 1221/2014: Adhesión al Programa Nacional de Infraestructuras Críticas de Información y Ciberseguridad.

[11] Estado Mayor Conjunto de las Fuerzas Armadas, Comando Conjunto de Ciberdefensa. Fuerzas-armadas.mil.ar. (2020). Recuperado el 4 de agosto de 2020, desde http://www.fuerzas-armadas.mil.ar/ComandoConjuntoDeCiberdefensa/Default.aspx.

[12] *CSIRT | Ministerio de Seguridad de la Nación Argentina*. Csirt.minseg.gob.ar. (2020). Recuperado el 3 de agosto de 2020, desde https://csirt.minseg.gob.ar/.

[13] BA-CSIRT | Centro de ciberseguridad ciudadana. Ba-csirt.gob.ar. (2020). Recuperado 3 de agosto de 2020, desde https://www.ba-csirt.gob.ar/.

[14] *CSIRT – NQN*. Csirt-nqn.neuquen.gov.ar. (2020). Recuperado el 3 de agosto de 2020, desde https://csirt-nqn.neuquen.gov.ar/.

[15] *CERT UNLP*. Cespi.unlp.edu.ar. (2020). Recuperado el 3 de agosto de 2020, desde https://www.cespi.unlp.edu.ar/cert.

[16] *1- ¿NIC Argentina tiene un CSIRT (Equipo de Respuesta ante Incidencias de Seguridad) para realizar denuncias por delitos informáticos? ​*. NIC Argentina. Recuperado el 5 de agosto de 2020, desde https://nic.ar/es/node/293.

[17] *Red Link*. Redlink.com.ar. (2020). Recuperado el 5 de Agosto de 2020, desde https://www.redlink.com.ar/servicio\_csirt.html.

[18] *Ciberseguridad en la Industria Financiera*. Itu.int. (2020). Recuperado el 5 de Agosto de 2020, desde https://www.itu.int/en/ITU-D/Regional-Presence/Americas/Documents/EVENTS/2018/20577/Discussion%20Finance%20Medios%20de%20Pago.pdf.

[19] iProUP. (2019). Bancos y servicios financieros, en alerta: Argentina recibió 1.590 millones de ciberataques en 2019. Recuperado el 8 de agosto de 2020, desde https://www.iproup.com/innovacion/11963-seguridad-argentina-recibio-1-590-millones-de-ciberataques-en-2019.

[20] Resumen Ejecutivo de Amenazas, período Abril - Junio de 2020. (2020). Recuperado el 6 de agosto de 2020, desde https://www.fortinetthreatinsiderlat.com/es/Q2-2020/AR/html/trends#trends\_position.

[21] *El panorama de la ciberseguridad en números*. Argentina.gob.ar. (2020). Recuperado el 5 de Agosto de 2020, desde https://www.argentina.gob.ar/sites/default/files/cofemod\_comisionciberseguridad\_el\_panorama\_de\_la\_ciberseguridad\_en\_numeros\_12-08-16.pdf.

[22] Infobae. (2020). En 2016, el PBI cayó en pesos pero alcanzó un récord en dólares. Recuperado 5 de Agosto de 2020, desde https://www.infobae.com/economia/2017/03/25/en-2016-el-pbi-cayo-en-pesos-pero-alcanzo-un-record-en-dolares/.

[23] Chuvakin, A. (2017). Let's Define "SIEM"!. *Gartner*. Recuperado el 6 de agosto de 2020, from https://blogs.gartner.com/anton-chuvakin/2017/08/14/lets-define-siem/.

[24] *Security Information and Event Management*. Gartner. (2020). Recuperado el 7 de Agosto de 2020, desde https://www.gartner.com/reviews/market/security-information-event-management.

[25] *IBM QRadar SIEM - Visión general*. Ibm.com. (2020). Recuperado el 7 de Agosto de 2020, desde https://www.ibm.com/ar-es/products/qradar-siem.

[26] *Solución web para la generación de informes, cumplimiento de TI y auditoría de cambios en tiempo real de Active Directory - ADAudit Plus*. Manageengine.com. (2020). Recuperado el 7 de Agosto de 2020, desde https://www.manageengine.com/latam/active-directory-audit/.

[27] *NextGen SIEM Platform | UEBA, SOAR, & NDR | LogRhythm*. LogRhythm. (2020). Recuperado el 7 de Agosto de 2020, desde https://logrhythm.com/products/nextgen-siem-platform/.

[28] *LogPoint as a SIEM solution*. Logpoint.com. (2020). Recuperado el 7 de Agosto de 2020, desde from https://www.logpoint.com/en/product/logpoint-as-a-siem-tool/.

[29] *Soluciones SIEM | Enterprise Security Manager | McAfee ES*. Mcafee.com. (2020). Recuperado el 7 de Agosto de 2020, desde https://www.mcafee.com/enterprise/es-es/products/enterprise-security-manager.html.

[30] *Software ArcSight de gestión de información y eventos de seguridad (SIEM) | Micro Focus*. Microfocus.com. (2020). Recuperado el 7 de Agosto de 2020, desde https://www.microfocus.com/es-es/products/siem-security-information-event-management/overview.

[31] *InsightIDR: Your Cloud SIEM Tool For Threat Detection & Response*. Rapid7. (2020). Recuperado el 7 de Agosto de 2020, desde https://www.rapid7.com/products/insightidr/.

[32] *ELK Stack: Elasticsearch, Logstash, Kibana | Elastic*. Elastic.co. (2020). Recuperado el 7 de Agosto de 2020, desde https://www.elastic.co/es/elk-stack.

[33] *Machine Data Management & Analytics | Splunk Enterprise*. Splunk. (2020). Recuperado el 7 de Agosto de 2020 desde https://www.splunk.com/en\_us/software/splunk-enterprise.html.

[34] *The Smarter SIEM Solution | Exabeam*. Exabeam. (2020). Recuperado el 7 de Agosto de 2020, desde https://www.exabeam.com/product/.

[35] *Unified Security Management (USM) Anywhere*. Cybersecurity.att.com. (2020).

USM Anywhere es el nuevo nombre de AlienVault USM. Recuperado el 7 de Agosto de 2020, desde https://cybersecurity.att.com/products/usm-anywhere.

[36] *Open Source Log Management for All | Graylog*. Graylog.org. (2020). Recuperado el 8 de Agosto de 2020, desde https://www.graylog.org/products/open-source.

[37] *OSSIM: The Open Source SIEM | AlienVault*. Cybersecurity.att.com. (2020). Recuperado el 8 de Agosto de 2020, desde https://cybersecurity.att.com/products/ossim.

[38] *Security Onion Solutions*. Securityonionsolutions.com. (2020). Recuperado el 8 de Agosto de 2020, desde <https://securityonionsolutions.com/software>.

[39] *TravisFSmith/SweetSecurity*. GitHub. (2020). Recuperado el 8 de Agosto de 2020, desde <https://github.com/TravisFSmith/SweetSecurity>.

[40] *FortiSIEM | Sistema de administración de eventos & información de seguridad (SIEM)*. Fortinet. (2020). Recuperado el 8 de Agosto de 2020, desde https://www.fortinet.com/lat/products/siem/fortisiem.

[41] Caldentey, D. (2019). Por qué la ciberseguridad en la Industria 4.0 ya es tan necesaria y estratégica. *Universidad UNIR, Noticias De Ingeniería Y Tecnología*. Recuperado el 9 de Agosto de 2020, desde https://www.unir.net/ingenieria/revista/por-que-la-ciberseguridad-en-la-industria-4-0-ya-es-tan-necesaria-y-estrategica/.

[42] CCN-CERT. (2018). *Ciberamenazas Sector Salud*. CCN-CERT.

[43] *About | Curator Reference [5.8] | Elastic*. Elastic.co. (2020). Recuperado el 9 de Agosto de 2020, desde https://www.elastic.co/guide/en/elasticsearch/client/curator/current/about.html.

[44] *ElastAlert - Easy & Flexible Alerting With Elasticsearch — ElastAlert 0.0.1 documentation*. Elastalert.readthedocs.io. (2020). Recuperado el 9 de Agosto de 2020, desde https://elastalert.readthedocs.io/en/latest/elastalert.html.

[45] *FreqServer — Security Onion 16.04.7.1 documentation*. Docs.securityonion.net. (2020). Recuperado el 9 de Agosto de 2020, desde https://docs.securityonion.net/en/16.04/freqserver.html.

[46] *MarkBaggett/domain\_stats*. GitHub. (2020). Recuperado el 9 de Agosto de 2020, desde https://github.com/MarkBaggett/domain\_stats.

[47] *The Zeek Network Security Monitor*. Zeek. (2020). Recuperado el 9 de Agosto de 2020, desde https://zeek.org/.

[48] *Snort - Network Intrusion Detection & Prevention System*. Snort.org. (2020). Recuperado el 9 de Agosto de 2020, desde https://snort.org/.

[49] *Suricata*. Suricata. (2020). Recuperado el 9 de Agosto de 2020, desde https://suricata-ids.org/.

[50] *Why Docker? | Docker*. Docker. (2020). Recuperado el 9 de Agosto de 2020, desde https://www.docker.com/why-docker.

[51] *Redis*. Redis.io. (2020). Recuperado el 9 de Agosto de 2020, desde https://redis.io/.

[52] *Wazuh · The Open Source Security Platform*. Wazuh. (2020). Recuperado el 9 de Agosto de 2020, desde https://wazuh.com/.

[53] *OSSEC - Open Source HIDS - FIM, Rootkit Detection, Malware Detection*. OSSEC. (2020). Retrieved 4 November 2020, from https://www.ossec.net/about/.

[54] *Sguil - Open Source Network Security Monitoring*. Bammv.github.io. (2020). Recuperado el 9 de Agosto de 2020, desde http://bammv.github.io/sguil/index.html.

[55] *The squertproject*. Squertproject.org. (2020). Recuperado el 9 de Agosto de 2020, desde http://www.squertproject.org/.

[56] *Netsniff-ng toolkit*. Netsniff-ng.org. (2020). Recuperado el 9 de Agosto de 2020, desde http://netsniff-ng.org/.

[57] *Open Source Log Management Tool*. Syslog-ng.com. (2020). Recuperado el 9 de Agosto de 2020, desde https://www.syslog-ng.com/products/open-source-log-management/.

[58] *El motor de tu trabajo*. Slack. (2020). Recuperado el 9 de Agosto de 2020, desde https://slack.com/intl/es-ar/.

[59] *Jira | Software de seguimiento de proyectos e incidencias*. Atlassian. (2020). Recuperado el 9 de Agosto de 2020, desde https://www.atlassian.com/es/software/jira.

[60] *Telegram – a new era of messaging*. Telegram. (2020). Recuperado el 9 de Agosto de 2020, desde https://telegram.org/.

[61] *TheHive Project*. Thehive-project.org. (2020). Retrieved 4 November 2020, from https://thehive-project.org/.

[62] *Grok filter plugin | Logstash Reference [7.9] | Elastic*. Elastic.co. (2020). Recuperado el 9 de Agosto de 2020, desde https://www.elastic.co/guide/en/logstash/current/plugins-filters-grok.html.

[63] *Apache Lucene - Welcome to Apache Lucene*. Lucene.apache.org. (2020). Recuperado el 9 de Agosto de 2020, desde https://lucene.apache.org/.

[64] *Shodan*. Shodan.io. (2020). Recuperado el 18 de Agosto de 2020, desde https://www.shodan.io/.

[65] *VirusTotal*. Virustotal.com. (2020). Recuperado el 18 de Agosto de 2020, desde https://www.virustotal.com/gui/intelligence-overview.

[66] *Vision AI | Obtén información valiosa de las imágenes a través del AA*. Google Cloud. (2020). Recuperado el 18 de Agosto de 2020, desde https://cloud.google.com/vision?hl=es-419.

[67] *VMware vSphere Hypervisor gratuito, virtualización gratuita (ESXi) | AR*. VMware. (2020). Recuperado el 18 de Agosto de 2020, desde <https://www.vmware.com/ar/products/vsphere-hypervisor.html>.

[68] *Download Ubuntu Desktop | Download | Ubuntu*. Ubuntu. (2020). Recuperado el 18 de Agosto de 2020, desde from https://ubuntu.com/download/desktop.

# Anexos