OFICINA DE POSTGRADOS

Tema:

IMPLEMENTACIÓN DE UNA SOLUCIÓN “SECURITY INFORMATION AND EVENT MANGEMENT” ESCALABLE Y ACCESIBLE BASADA EN OPEN SOURCE

Proyecto de investigación previo a la obtención del título de Magíster en Ciberseguridad

Línea de Investigación:

Protección de datos y comunicaciones

Autor:

Miguel Alexander Mejía Broncano

Director:

Mg. Paul Fernando Bernal Barzallo

Ambato – Ecuador

SEPTIEMBRE 2021

PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR

SEDE AMBATO

HOJA DE APROBACIÓN

Tema:

IMPLEMENTACIÓN DE UNA SOLUCIÓN "SECURITY INFORMATION AND EVENT MANGEMENT" ESCALABLE Y ACCESIBLE BASADA EN OPEN SOURCE

Línea de Investigación:

Protección de datos y comunicaciones

Autor:

Miguel Alexander Mejía Broncano

Paul Fernando Bernal Barzallo**,** Mg. f._____

Firmado electrónicamente por: PAUL FERNANDO BERNAL BARZALLO

CALIFICADOR

Diego Fernando Ávila Pesantez, Mg. f._____

Firmado electrónicamente por: DIEGO FERNANDO AVILA PESANTEZ

CALIFICADOR

Jaime Gabriel Llumiquinga Veintimilla, Mg. f._____

JAIME GABRIEL LLUMIQUINGA VEINTIMILLA

Firmado digitalmente por JAIME GABRIEL LLUMIQUINGA VEINTIMILLA
Fecha: 2021.09.20 17:02:58 -05'00'

CALIFICADOR

Juan Carlos Acosta Teneda. Ing. Mg. f._____

COORDINADOR DE LA OFICINA DE POSGRADOS

Hugo Rogelio Altamirano Villaroel, Dr. f._____

SECRETARIO GENERAL PUCESA

Ambato – Ecuador

SEPTIEMBRE 2021

DECLARACIÓN DE AUTENCIDAD Y RESPONSABILIDAD

Yo: **MIGUEL ALEXANDER MEJIA BRONCANO**, con CC. **0604499798** autor del trabajo de graduación intitulado: **"IMPLEMENTACIÓN DE UNA SOLUCIÓN "SECURITY INFORMATION AND EVENT MANGEMENT" ESCALABLE Y ACCESIBLE BASADA EN OPEN SOURCE"**, previa a la obtención del título profesional de MAGÍSTER EN CIBERSEGURIDAD, en la OFICINA DE POSGRADOS.

1.- Declaro tener pleno conocimiento de la obligación que tiene la Pontificia Universidad Católica del Ecuador, de conformidad con el artículo 144 de la Ley Orgánica de Educación Superior, de entregar a la SENESCYT en formato digital una copia del referido trabajo de graduación para que sea integrado al Sistema Nacional de Información de la Educación Superior del Ecuador para su difusión pública respetando los derechos de autor.

2.- Autorizo a la Pontificia Universidad Católica del Ecuador a difundir a través del sitio web de la Biblioteca de la PUCE Ambato, el referido trabajo de graduación, respetando las políticas de propiedad intelectual de Universidad.

Ambato, septiembre 2021

MIGUEL ALEXANDER MEJIA BRONCANO

CC. 0604499798

AGRADECIMIENTO

A mi esposa por ser mi principal apoyo y motivación.

A mis padres por su ejemplo de trabajo y responsabilidad

A mi tutor, el Mg Paul Bernal por su experiencia, su tiempo y sus conocimientos aportados al presente trabajo

Miguel

DEDICATORIA

A mi esposa por su apoyo incondicional, su ejemplo de responsabilidad, su paciencia y en especial por su amor que me motiva a seguir adelante día a día.

Miguel

RESUMEN

Según datos de la revista Ecos del 2017, el 95% de las unidades productivas en el Ecuador son generados por pequeñas y medianas empresas, mismas que por su tamaño y condiciones, normalmente no poseen el presupuesto necesario para contratar y mantener una solución SIEM (*Security Information and Event Management*), el presente trabajo de investigación tiene como objetivo general implementar una herramienta SIEM basada en open source para el acceso a la gestión de incidentes de seguridad en las pequeñas y medianas empresas. Como metodología para el desarrollo se utilizó el marco de trabajo Scrum y se estableció la metodología de investigación experimental por los procesos comparativos requeridos en entorno con y sin la solución planteada. Durante el desarrollo de la solución se utilizaron tecnologías como docker, docker compose, java, spring boot, elasticsearch, logstash, kibana, miro, gitlab entre otras. Como resultado se obtiene una herramienta open source, más asequible que las existentes en el mercado actual, capaz de detectar y notificar en menos de 60 segundos a partir del incidente el 100% de las anomalías en el comportamiento de los servidores, así como identificar también posibles incidentes de seguridad basados en los comportamientos correlacionados de los diferentes orígenes de datos.

PALABRAS CLAVES: SIEM, Gestión de eventos de seguridad, Análisis de logs, Big Data

ABSTRACT

According to data from the Ecos magazine of 2017, 95% of the production units in Ecuador are generated by small and medium-sized companies, which due to their size and conditions, normally do not have the necessary budget to contract and maintain a SIEM (Security Information and Event Management). The present research work has the objective of implementing a SIEM tool based on open source for access to security incident management in small and medium-sized companies. The methodology for the development of this research applies the Scrum framework. Likewise, the experimental research methodology is used for the comparative processes required according to the environment, with and without the proposed solution. During the proposal development, technologies such as docker, docker compose, java, spring boot, elasticsearch, logstash, kibana, miro, gitlab, among others, are applied. As a result, an open source tool is obtained, without any direct installation or maintenance cost, capable of detecting and reporting 100% of the anomalies in the behavior of the servers in less than 60 seconds from the incident. As well as also identify possible security incidents based on the correlated behaviors of the different data sources.

KEYWORDS: SIEM, SecurIty Event Management, Log analysis, Big Data

ÍNDICE GENERAL

ÍNDICE DE TABLAS

ÍNDICE DE ILUSTRACIONES

INTRODUCCIÓN

La gran acogida del internet provocó un crecimiento exponencial en cuanto a la cantidad de información que se transmite actualmente, lo cual ha ocasionado que las infraestructuras se vuelvan cada vez más complejas e integren mayor cantidad de herramientas tecnológicas y sistemas, en relación con este contexto, Miloslavskaya, (2018) puntualiza que, mientras más herramientas ingresan a un ecosistema tecnológico, crece exponencialmente la cantidad de vulnerabilidades a las que están sujetos, por lo que termina su estudio sobre el uso de sistemas *Security Information and Event Management* (SIEM) en Operaciones de Seguridad al mencionar que, el 60% de las ocasiones la información sensible de una empresa se encuentra simplemente a minutos de trabajo de un hacker.

Específicamente, Bussa et al. (2010) en su estudio *Critical Capabilities for Security Information and Event Management***,** emiten una cifra alarmante, en la que afirman que el 30% de las empresas relevantes globales han visto directa o indirectamente comprometida su información, dato que concuerda con lo expresado por Chopra & Mahapatra (2019) los cuales mencionan un 23% de robos de información en empresas medianas europeas, este estudio menciona, además, la complejidad de la interpretación de la información y los elevados costos que implica contratar un servicio que lo realice.

Relacionado al tema del monitoreo, en el estudio *Why SIEM is Irreplaceable in a Secure IT Environment,* Podzins & Romanovs (2019) realizan un análisis del estado del arte actual de las empresas y la cantidad de sistemas y logs que maneja cada uno según su tamaño, presenta también el concepto de *Total Cost of Ownership* (TCO) mismo que servirá como medida de cuantía dado que en su cálculo incluye eventualidades de seguridad y *Security Operations Centre* (SOC) que es el departamento encargado de las operaciones de seguridad de la empresa, el estudio termina con el análisis entre su TCO y el elevado costo de la implementación de la solución SIEM de IBM, para evidenciar que es rentable en relación al gran tamaño de la empresa.

En relación a las alternativas existentes sin contar con la mencionada anteriormente propiedad de IBM, el análisis comparativo realizado por Safarzadeh et al. (2019)

deja en evidencia que todas las herramientas relevantes actualmente en el ámbito de las SIEM están principalmente a cargo de empresas privadas, lo cual limita el acceso a las mismas por parte de las empresas que no poseen los suficientes recursos económicos para su adquisición y mantenimiento.

Finalmente, en el entorno local, en la Revista Ecos Ron Amores & Sacoto Castillo (2017) manifiestan que, en el Ecuador 7 de cada 10 plazas de trabajo son generadas por las pequeñas y medianas empresas, en adelante PYME, así mismo como el 95% de las unidades productivas, lo cual, al contrastarlo con el presupuesto de operaciones por nivel, denota que el sector de interés para el estudio son las PYME, dado que, a diferencia de las grandes empresas, no suelen contar con el elevado presupuesto requerido para la implementación y mantenimiento de una herramienta SIEM.

Por lo cual, el problema de investigación se define como: el acceso limitado de las pequeñas y medianas empresas a herramientas SIEM, por sus elevados costos.

Es así, que para esta investigación se ha definido la siguiente hipótesis de trabajo: la implementación de una herramienta SIEM basada en open source, mejorará el acceso a la gestión de incidentes de seguridad en pequeñas y medianas empresas.

Para lograr alcanzar este resultado se ha definido el siguiente objetivo general: Implementar una herramienta SIEM basada en open source para el acceso a la gestión de incidentes de seguridad en las pequeñas y medianas empresas.

Mismo que se subdivide en los siguientes objetivos específicos.

1. Analizar teóricamente los componentes de las herramientas SIEM propietarias existentes para la definición de las funcionalidades mínimas requeridas.
2. Definir un listado de funcionalidades requeridas para su implementación utilizando el marco de trabajo ágil scrum.
3. Aplicar métodos para la recolección y análisis de resultados
4. Diseñar un procedimiento para la replicabilidad de la herramienta por parte de la comunidad.

Una vez definido los elementos de la solución, para el desarrollo de los mismos se analiza la metodología que se adapte mejor a la propuesta, para lo cual previo análisis se determina el proyecto como ágil e iterativo, además, dado que durante el desarrollo pueden surgir cambios repentinos o nuevas funcionalidades a implantar se establece que la metodología Scrum cumple con lo que se requiere en la implementación de la propuesta, por otra parte, al ser un proyecto con un enfoque experimental y se requiere establecer comparaciones del estado del ambiente antes y después de la implementación de la solución, se considera como metodología de investigación a la experimental.

En base a todo lo mencionado anteriormente y a la situación social de los países en vías de desarrollo, quienes prefieren invertir en operatividad antes que en la seguridad de la información, se hace visible la necesidad que justifica la presente investigación, pues, al entregar a las PYME una herramienta de software libre gratuita, su uso podrá masificarse y así mejorar la visibilidad de incidentes de seguridad sin una fuerte inversión económica, es importante también aclarar que es un sistema de monitoreo y análisis de incidentes, no orientado a la corrección automática de los mismos, por cuanto debería usarse en ambientes previamente asegurados para poder conservarlos en ese estado.

Para el presente estudio, según lo especifica Ron Amores & Sacoto Castillo (2017) en la revista espacios, se considera que una PYME es aquella con menos de 200 empleados y que tiene como finalidad la creación de productos o servicios para cubrir las necesidades de su mercado, con estas condiciones, se consigue que la Corporación Ecuatoriana para el Desarrollo de la Investigación y la Academia, en adelante CEDIA, participe como institución de pruebas y provea del primer entorno para el funcionamiento de la herramienta resultante, por cuanto, la presente solución se desarrolló con su colaboración, dado que, tiene entre 51 y 200 trabajadores según su página oficial de LinkedIn CEDIA (2021).

Para mantener la privacidad de la información, que es un tema muy demandado actualmente, se recomienda que el servidor de monitoreo se encuentre desplegado de manera local en la infraestructura informática a proteger, pero, al seguir la filosofía del software libre, se espera que alguna institución se convierta en hospedera oficial de la solución y la comparta como servicio a las PYME

interesadas del sector, como ejemplo CEDIA sería un primer hospedero en el sector educativo para las universidades miembros de su red.

CAPÍTULO I. ESTADO DEL ARTE Y LA PRÁCTICA

1.1. Security Information and Event Management

Según Pico Barrera (2016) el término SIEM es utilizado para señalar a las herramientas relacionadas a la información de seguridad y administración de eventos dentro de un contexto informático a nivel de toda su infraestructura, tanto hardware como software y es el resultado de tres herramientas antecesoras: *Security Information Management* (SIM), *Security Event Management* (SEM) y *Security Event and Information Management* (SEIM).

Básicamente la función principal de un SIEM se basa en el monitoreo en tiempo real de los eventos suscitados dentro de la infraestructura tecnológica institucional, mismos que dan origen a varias alertas, logran definir causas o advertencias de aspectos sospechosos para el normal desempeño tanto del hardware y software institucional, las amenazas pueden proyectarse desde el internet o desde el interior de la misma organización.

1.1.1. Características

Tanto para Villafuerte Quiroz & Bravo Bravo (2015), como para, Balarezo Chávez & Poveda Pilatasig (2015) una herramienta SIEM moderna debería cumplir los siguientes aspectos fundamentales.

- Control de direccionamiento IP
- Cumplimiento normativo de TI.
- Acceso centralizado y administración de logs.
- Correlación de eventos.
- Respuesta activa del servidor (seguridad y monitoreo local)
- Seguridad de *endpoint* y escaneo de equipos

De los cuales, se menciona que el punto central de origen para el análisis completo es la centralización de logs, convirtiéndose la fase de recolección en fundamental para el éxito de la herramienta.

1.1.2. Alcance

Como método de publicidad, empresas de tecnología como Nsit, (2020) mencionan que una herramienta SIEM debe permitir una cobertura absoluta de la

infraestructura informática tanto hardware como software, al igual que la solución QRadar de IBM, (2020).

Con un punto de vista técnico y no de marketing, en base a los resultados de Villafuerte Quiroz & Bravo Bravo (2015) y de Balarezo Chávez & Poveda Pilatasig (2015), se define que el alcance de una herramienta SIEM se ve delimitado por los orígenes de datos configurados, es decir, los sistemas o equipos que producen logs hacia el SIEM y la capacidad de este para integrarlos a los datos ya existentes, se hace énfasis en que la palabra “absoluta” utilizada para publicidad se limita a “todo aquello que esté en la capacidad de entregar logs que puedan ser normalizados a un formato específico”.

1.1.3. Opciones en el mercado

En el análisis comparativo de Vazão et al. (2019) tenemos un listado de alternativas open source que cumplen la función de SIEM o que conjuntamente con otras herramientas también open source pueden cumplirlo, de las que destaca el stack de elasticsearch como el más completo y viable para su necesidad, esta herramienta se analizará más adelante.

Del lado privativo existe un mayor rango de posibilidades como QRadar de IBM, Nsit SIEM, entre otros de gigantes tecnológicos, los cuales coinciden en un modelo de negocio basado precio por cantidad de sistemas monitoreados o precio por cantidad de logs procesados, lo que limita el crecimiento del área de cobertura del SIEM, así lo define el estudio de Chopra & Mahapatra (2019).

Finalmente, Fernández et al. (2017) realiza un análisis comparativo de los SIEM analizados, la siguiente tabla resumen, contiene el listado de las funcionalidades más recurrentes en los mismos.

Ilustración 1: Comparativa de funcionalidades

Descripción	*Splunk*	*Alient Vault OSSIM*	*Elastic Mozdef*	*Prelude*
Basado en software libre	Si	Si	Si	Si
Escalabilidad horizontal	No	No	Si	No
Capacidad de almacenamiento y procesamiento de *logs*.	Limitada	Limitada	De acuerdo a hardware	De acuerdo a Hardware
Posibilidad de transferencia segura de datos.	Si	Si	Si	Si
Visualizaciones de estado diario, semanal, anual.	Si	Si	Si	Si
Gestor de Incidentes	No	No	Si	Si
Normalización de eventos.	Si	Si	Si	Si
Cluster o balanceo de carga	Si	Si	Si	No
Dashboard personalizables	Si	Si	Si	Si
Experiencia				
Casos de uso, acercamiento clientes conocidos	Si	Si	Si	No
Experiencia en el mercado	12 años	9 años	7 años	10 años
Soporte y documentación				
Cuenta con sitio web de foros y comunidad para soporte	Si	Si	Si	Si
Calificación	100	100	120	90

Fuente: adaptado a partir de Fernández et al. (2017)

1.2. Monitoreo de Red

Según la tesis universitaria de Lima Torrico (2019), el monitoreo se define como el proceso de supervisar y dar seguimiento a cada una de las actividades o eventos que suceden dentro del ambiente de la empresa, con el fin de identificar tanto comportamientos como posibles eventualidades.

Dado la gran cantidad de información que circula dentro de una empresa y se debería monitorear, este proceso se ha dividido por áreas, para este trabajo se utilizará únicamente la informática, por cuanto el ambiente a dar seguimiento se encuentra compuesto por la red y la información que circula por la misma.

Debido a la naturaleza de esta información y lo variado de sus orígenes, suele ser procesos bastante costosos tanto en tiempo como en recursos, por lo cual se apoyan sobre herramientas para poder optimizar el esfuerzo, se dividen principalmente en 3 tipos.

- **Herramientas de observación**: Son las encargadas de recoger las eventualidades cronológicamente y almacenarlas para su posterior análisis.
- **Herramientas analíticas**: Una vez almacenados los datos, estas herramientas se encargan de obtener información importante de estos y presentarlos en un formato comprensible.
- **Herramientas de *engagement***: Este último apartado, tiene como objetivo facilitar la toma de decisiones en base a la información obtenida y analizada.

1.3. Logs

El término log es utilizado en actividades de *logging*, análisis de logs y administración de logs, debido a que posee varios significados. Básicamente son archivos contenedores de texto que pueden mostrar información ocurrida después de un evento o diagnosticar problemas para lograr una solución, se caracterizan por no poder ser modificados o alterados automáticamente durante el normal uso de algún sistema específico. Los logs usualmente tienen referencias de tiempo en cada grabación, almacenan una secuencia cronológica de eventos, que no solo muestra lo que sucedió, sino cuando sucedió y en qué orden.

Un sistema SIEM posee la cualidad de centralizar todos los procesos que se encuentran disponibles para lograr obtener su cometido, adicional la administración de logs empieza con la configuración de nodos dentro de un sistema de tecnologías de la información, particularmente los nodos más importantes o críticos, para enviar la información relevante y eventos de aplicaciones (logs) a una base de datos centralizada y administrada por la aplicación SIEM, su base de datos analiza y normaliza los datos enviados por los numerosos y muy rápidos nodos existentes dentro de un sistema de tecnologías de la información.

1.4. Correlación

De acuerdo a lo mencionado por Pico Barrera (2016), la correlación de eventos permite aportar al sistema SIEM con un nivel de inteligencia mayor, debido a que no muestra únicamente los eventos sino también la posibilidad de reaccionar o no a dicha acción, fundamentándose en varias condiciones, una vez ejecutadas las alarmas, el motor de correlación de los sistemas SIEM está diseñado para investigar y considerar eventos existentes dentro de una base de datos dispuesta

para este fin, existen varias opciones en el mercado, de las que actualmente destaca elasticsearch dentro de los proyectos open source.

1.5. Metodologías ágiles

Aparecieron formalmente con la definición del Manifiesto Ágil y su variante especializada en el desarrollo de software, según Beck et al. (2001), nacieron en oposición a los métodos tradicionales que eran considerados pesados de cumplir y mantener.

Enfocados en reducir la excesiva documentación, los métodos ágiles a diferencia de los tradicionales prefieren entregar productos funcionales en períodos cortos de tiempo con el uso de estándares, se logra así, que el cliente tenga un conocimiento más extenso del producto que simplemente al leer sobre él, con este enfoque, se pueden mencionar 2 diferencias importantes entre las 2 tendencias.

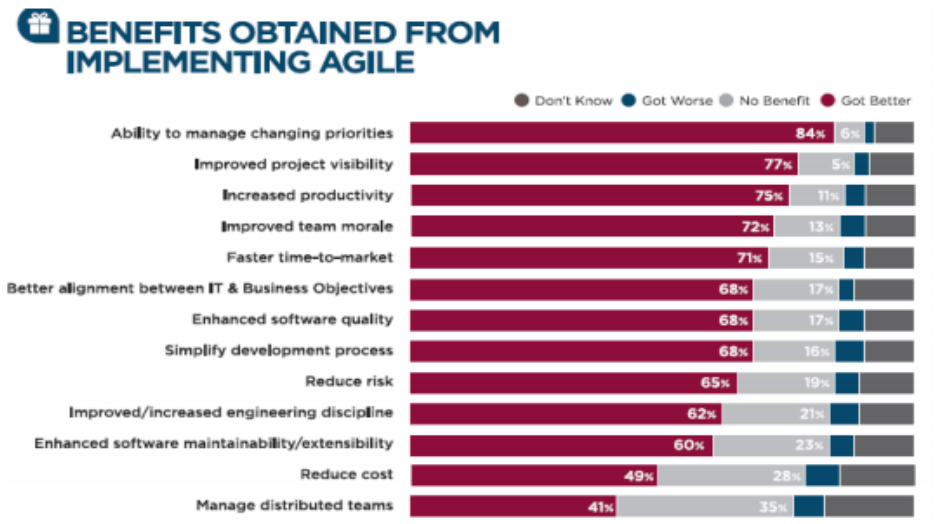
Primero, los métodos tradicionales normalmente se basan en estrictas planificaciones que pretenden adivinar todos los eventos posibles a suceder durante el desarrollo, proceso que las hace poco flexibles en caso de cambios inesperados, mientras que las metodologías ágiles consideran al cambio como normal y se basan en adaptarse pronto a este para obtener soluciones.

Segundo, para lograr reaccionar rápidamente a los cambios o inconvenientes, los métodos ágiles se enfocan en las personas, antes que, en los procesos, lo que permite que las decisiones sean más oportunas y no necesariamente burocráticas, con la aceptación de la naturaleza humana en lugar de intentar controlar la misma.

En consideración a lo anteriormente mencionado, Digital.ai (anteriormente CollabNetVersionOne) (2020) en su reporte *14th Annual State of Agile Report*, menciona como ventajas del uso de metodologías ágiles un 71% de mejora en su indicador *Faster time-to-market,* que hace referencia a tiempo de puesta en

producción, mientras que muestra también una mejora de un 68% en el enfoque a los objetivos de la organización.

Ilustración 2: Beneficios de implementar metodologías agiles



Fuente: adaptado a partir de VisionOne (2020)

1.6. Asequibilidad

La Real Academia Española (2021) define como asequible aquello que se puede conseguir o adquirir, basados en esta definición formal, se tienen 2 condiciones para que una herramienta sea más asequible o menos asequible:

Se puede conseguir, en referencia a la existencia de al menos una unidad que pueda suplir la necesidad requerida.

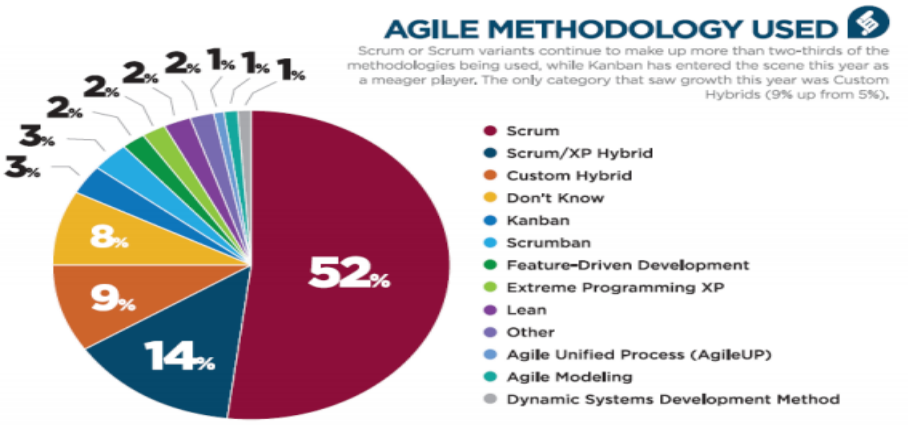
Se puede adquirir, en referencia a la existencia del presupuesto necesario para realizar la compra o alquiler del bien.

1.7. Scrum

Nacido como un marco de trabajo para procesos de producción, fue adoptado como metodología de desarrollo a principios de los 90 con la publicación del Manifiesto Ágil, basado en el empirismo, trata de producir el conocimiento en base a la experiencia y a la toma de decisiones sobre las mismas Schwaber & Sutherland (2020), está optimizado a través de Lean, lo que permite dejar de lado las practicas obsoletas y poco eficientes para enfocarse en el trabajo.

Una vez decididos los métodos ágiles, basados en el informe de Digital.ai, mismo que muestra en sus resultados que Scrum y sus variantes son actualmente las más utilizadas con un alto margen de éxito en proyectos iterativos y evolutivos.

Ilustración 3: Metodologías ágiles



Fuente: adaptado a partir de VisionOne (2020)

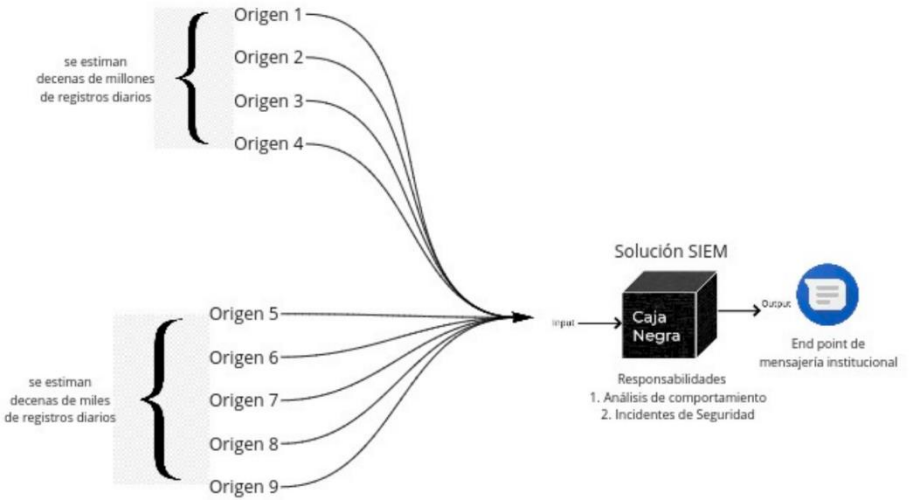
La versión más actual de la guía de scrum fue liberada en noviembre de 2020 y se puede encontrar en el siguiente enlace: https://scrumguides.org/docs/scrumguide/v2020/2020-Scrum-Guide-Spanish-European.pdf

CAPÍTULO II. DISEÑO METODOLÓGICO

2.1. Security Information and Event Management

Mediante una entrevista no estructurada mantenida el día 8 de octubre de 2020 con los Coordinadores del Área Técnica de CEDIA se define el proceso de análisis y correlación requerido, así como las tecnologías y herramientas necesarias para la implementación, la ilustración 3 representa el diagrama de proceso efectuado para el requerimiento.

Ilustración 4: Diagrama de procesos



Fuente: elaboración propia

Como se puede apreciar en el gráfico, los orígenes de datos son muy variados y se comunican a través de protocolos distintos, las tendencias actuales de diseño basado en dominios, a partir de ahora DDD por sus siglas en inglés *(Domain Driven Design),* sugiere para arquitecturas con estas características construir una capa anticorrupción que sea la responsable directamente de la ingesta de información útil, que deseche cualquier dato incorrecto o inconsistente, al tratarse de un proyecto basado en software libre, se plantea que este componente se desarrolle en modelo de *companion (P2P)* para que la creación de más traductores para más orígenes pueda ser realizado independientemente del *core* principal del SIEM.

Otro punto para considerar es la cantidad de información manejada (logs), como se puede apreciar en la ilustración 3**,** se estima recibir decenas de millones de registros diarios desde los orígenes 1, 2, 3 y 4, mismos que son un interés de la organización pues corresponden a las instituciones miembros, por cuanto la escalabilidad y la alta disponibilidad serán factores importantes en la implementación.

Se considera que la información a analizar es producida continuamente y que el moverla para un análisis aislado es poco práctico dado su extensión y su constante varianza, se requiere que el desarrollo sea incremental y que inicie por un agente capaz de recibir los logs en tiempo real para su posterior utilización, por cuanto un marco de trabajo basado en iteraciones será muy beneficioso en esta implementación.

Adicionalmente, dado que el desarrollador posee una certificación como Scrum *developer*, por facilidad y conocimiento previo, así como para aplanar la curva de aprendizaje requerida en caso de la definición de algún *framework* de trabajo desconocido, se plantea este marco de trabajo como solución metodológica.

2.2. SCRUM

2.2.1. El Equipo Scrum (*Scrum Team*)

Como lo menciona la metodología, en dependencia a lo que cada usuario deberá aportar en el proyecto, se le asigna un rol, para la presente implementación se definieron 3 roles, mismos que se detallan en la Tabla 1.

Tabla 1: Scrum Team

Rol	Persona	Funciones
Product Owner	Ing. Paul Bernal Barzallo	Propietario del *Product Backlog,* experto en el funcionamiento del negocio con una amplia visión sobre la solución a largo plazo requerida con la implementación actual.
Development Team	Ing. Miguel Mejía	Encargado del desarrollo de las Historias de Usuario del *Backlog*, experto en desarrollo de software y prácticas ágiles
Stakeholders	Coordinadores del Área Técnica de CEDIA	Consultores y Asesores internos expertos en el funcionamiento del negocio y sus áreas involucradas.

Fuente: elaboración propia

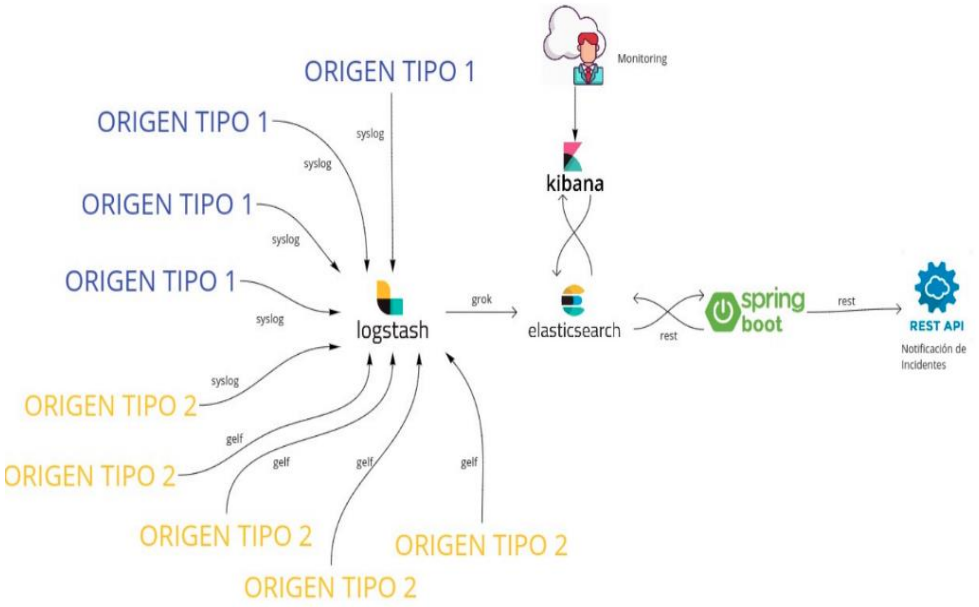
2.2.2. Estudio Preliminar

Previo al inicio del desarrollo y la inversión de esfuerzo y de recursos en el mismo, tanto por parte del equipo de desarrollo como del cliente, se procede a realizar un estudio de factibilidad entre las 2 partes involucradas, mismo que se declara factible bajo las condiciones expuestas y genera el convenio de acuerdo de desarrollo entre las 2 partes, cuya última página se encuentra en el **Anexo 1.**

Arquitectura Propuesta

Por todas las interacciones necesarias se plantea que la arquitectura sea basada en capas con la finalidad de obtener escalabilidad y basarse en el principio de responsabilidad única, el software al ser un producto evolutivo es propenso a cambiar de arquitectura rápidamente, por cuanto este diagrama se encuentra disponible en el siguiente link de Miro (https://miro.com/app/board/o9J_l_VBkpY=/)

Ilustración 5: Arquitectura de la Solución



Fuente: elaboración propia

Enfoque de Escalabilidad

En consideración a lo expuesto en la ilustración 3 y su análisis, se requiere que la solución sea capaz de mantener volúmenes considerablemente extensos de información en tiempos aceptables, lo cual relacionado a la arquitectura especificada en la ilustración 4**,** denota la necesidad de escalabilidad tanto en los colectores de información, así como en los nodos procesadores.

Según la documentación de elasticsearch mencionada en el **Capítulo 1**, su redundancia y sus múltiples nodos son su principal ventaja en cuanto a rendimiento y mencionan el uso de la tecnología de contenedores para su gestión, por cuanto los colectores y procesadores a desarrollarse también se encontrarán bajo esta arquitectura de desarrollo, lo que permite dar soporte al tamaño requerido de manera horizontal.

2.2.3. Product Backlog

Dada la naturaleza del negocio y ya definidos los roles, se procede al levantamiento de los requisitos funcionales, se utiliza como punto de partida el listado expuesto en el punto 1.1.3 y se complementa con las necesidades particulares de la institución, el listado resultante se plasma en el *Product Backlog* con la herramienta de gestión de tareas de gitlab (https://about.gitlab.com/), como lo menciona Scrum, las historias se organizan según su importancia para el negocio y se agregarán al sprint en ese orden para que sean desarrolladas.

Como resultado se obtiene el siguiente listado de funcionalidades asignadas al sprint en el cual fueron tratadas y su correspondiente estado**.**

Tabla 2: Product Backlog

N°	HISTORIA DE USUARIO	ESTADO	SPRINT
1	Definición de la arquitectura de la solución.	Terminado	1
2	Preparación del ambiente de desarrollo.	Terminado	1
3	Despliegue de la infraestructura para pruebas	Terminado	2
4	Implementación de un nodo de pruebas elasticsearch	Terminado	2
5	Implementación de un stack elk	Terminado	3
6	Implementación de un colector de logs para syslog	Terminado	3
7	Optimización de los servicios para evitar desbordamiento de recursos	Terminado	3
8	Implementación de un intérprete para logs dns	Terminado	4
9	Implementación de un colector de logs para gelf	Terminado	5
10	Implementación de un intérprete para logs gelf	Terminado	5
11	Definición de una línea base de funcionamiento	Terminado	6
12	Servicio de consolidación de logs	Terminado	6
13	Servicio de análisis de tendencias	Terminado	7
14	Servicio de detección de anormalidades	Terminado	7
15	Servicio de correlación de orígenes	Terminado	8
16	Servicio de reporte de incidentes	Terminado	8

Fuente: elaboración propia

2.2.4. Sprint planning meeting

Como lo menciona la guía oficial, se trata de una reunión corta para determinar las funcionalidades requeridas en el siguiente sprint, la definición de una meta o incremento deseado por parte del *product owner* se considera importante, dado que este será el punto de enfoque del equipo durante la iteración y cualquier esfuerzo o decisión debe enfocarse a cumplir esta meta aún sobre el desarrollo de las historias, en la siguiente tabla se detalla las fechas en que se realizaron cada uno de los 8 *planning* para los 8 *sprint* trabajados.

Tabla 3: Sprint planning

N°	Historia de Usuario	Fecha de inicio	Fecha de Fin	*Sprint Goal*
1	1,2	14/09/2020	27/09/2020	Tener el entorno preparado para iniciar la Implementación.
2	3,4	28/09/2020	11/10/2020	Tener un servidor local con características al solicitado en funcionamiento
3	5, 6, 7	08/02/2021	21/02/2021	Iniciar la ingesta de logs dns en el servidor autorizado
4	8	22/02/2021	07/03/2021	Iniciar el procesamiento de logs dns
5	9, 10	05/04/2021	18/04/2021	Iniciar la ingesta y procesamiento de logs gelf.
6	11, 12	19/04/2021	02/05/2021	Definir una línea base de comportamiento de los servidores
7	13, 14	03/05/2021	16/05/2021	Identificar comportamientos anormales en los servidores analizados
8	15, 16	17/05/2021	30/05/2021	Identificar posibles incidentes de seguridad basados en la correlación de orígenes y notificarlos al api de mensajería institucional

Fuente: elaboración propia

2.2.5. Sprint Backlog

Como parte fundamental del desarrollo de la solución, se tiene el sprint, que según menciona la documentación no debería ser mayor a 4 semanas, por conveniencia de horarios dentro del equipo, se define el mismo para 2 semanas de trabajo, al ser un equipo maduro en scrum, se acoge la sugerencia de no tener estimaciones de tiempo sino entregas constantes de software utilizable.

El artefacto resultante del Sprint *Planning*, es decir el listado de funcionalidades deseadas representadas por las historias de usuario correspondientes constituyen formalmente el Sprint *Backlog*, mismo que será tratado en la duración del Sprint historia por historia, a continuación, se detalla como ejemplo la Historia de Usuario 5, las demás Historias se encuentran en el **Anexo 2.**

Tabla 4: Historia de usuario 5

Nombre de la Historia: Implementación de un stack elk	
Estado: Terminado	**Responsable:** Miguel Mejía
Fecha de Inicio: 08/02/2021	**Fecha Fin:** 11/02/2021
Contexto Dado que, ya se encuentra aprobado el servidor a utilizarse para este proyecto en la institución, es necesario desplegar la infraestructura requerida para iniciar con la recolección y organización de información. **Como** administrador de la herramienta **necesito** que la información generada para analizar pueda ser procesada sin interferir en el funcionamiento de las operaciones y la misma se mantenga disponible en el tiempo. **Criterios de aceptación** 1. Al reiniciar algún servicio, las configuraciones deben ser persistentes 2. En caso de destruir y volver a construir los contenedores, las configuraciones deben ser persistentes	

3. El *stack* debe poder replicarse en otro servidor sin errores mientras tengan instalada una versión actual de docker y docker-compose

Fuente: elaboración propia

De la Historia de usuario a medida que se realizó el desarrollo se generaron las siguientes tareas.

1. Análisis *bind mount vs volume*
2. Implementación de elasticsearch
3. Implementación de logstash
4. Implementación de kibana
5. Configuraciones iniciales
6. Definición de volúmenes

2.2.6. Incremento

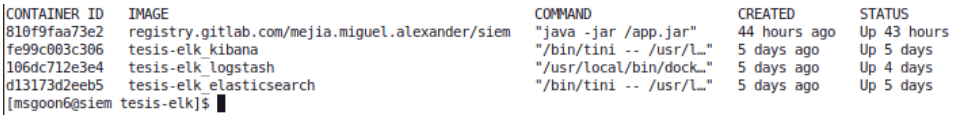
De acuerdo a la filosofía del agilismo, cada Historia terminada debería generar un producto potencialmente desplegable en producción que brinde valor al negocio, a la suma de todos los resultados de las historias del sprint se lo conoce como Incremento. Para este proyecto, se generaron un total de 8 incrementos, mismos que fueron colocados en producción con cada *sprint review*, los mismos se detallan a continuación.

Incremento 1: En cumplimiento con los criterios de aceptación establecidos en la Historia de usuario 1, se obtuvo el diagrama de la arquitectura propuesta para el sistema, mismo que ha evolucionado con los nuevos requerimientos, la Ilustración 4 muestra la arquitectura, además, el siguiente enlace mantiene una versión actualizada https://miro.com/app/board/o9J_l_VBkpY=/, se obtuvo también una segunda parte, un ambiente de desarrollo compuesto de repositorios y herramientas requeridas para desarrollar la solución definida en la arquitectura.

Incremento 2: Se obtiene un servidor local con un stack elasticsearch, logstash y kibana, mismo que puede ser replicado en cualquier servidor con docker y docker-compose, el siguiente archivo resultante cumple las funcionalidades requeridas para el ambiente de pruebas a utilizar

https://gitlab.com/mejia.miguel.alexander/tesis-elk/-/blob/50cd8cee8ef7d234b547a20c4796a3d864e118bd/docker-compose.yml

Ilustración 6: Entorno de pruebas ELK



Fuente: elaboración propia

Incremento 3: Ya con el servidor autorizado y disponible, se despliega el resultado del Incremento 2 en esta nueva infraestructura, mismo que funciona normalmente gracias a la portabilidad que ofrece docker, para iniciar con la ingesta de datos se establecen los puertos requeridos, así como los filtros necesarios en el archivo pipelines/logstash.conf.

Ilustración 7: Puertos logstash para syslog

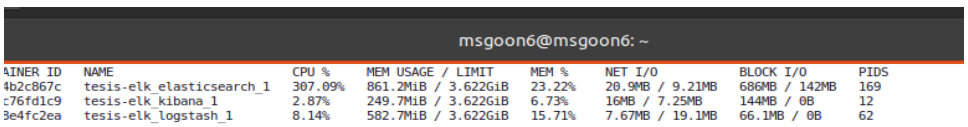
```
[msgoon6@siem tesis-elk]$ cat logstash/pipeline/logstash.conf
input {
        beats {
                port => 5044
        }

        tcp {
                port => 5000
                type => syslog
        }
        udp {
                port => 5000
                type => syslog
        }
}
```

Fuente: elaboración propia

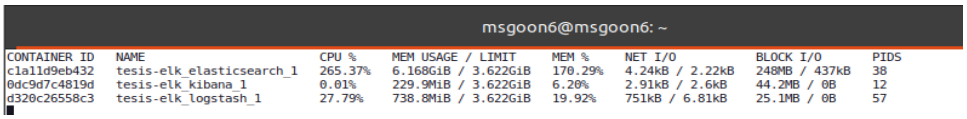
Dado que, cualquier puerto inferior al 1024 requiere permisos de administrador, y que, syslog usa por defecto el puerto 514, se decide utilizar el puerto 5000 local para evitar el permiso elevado requerido y mapearlo a través de docker al 514 requerido en el host, al iniciar la ingesta de datos, el servidor inmediatamente se queda sin recursos, intermitentemente se agota la ram o el cpu, por lo que se solicita más hardware a la institución.

Ilustración 8: CPU desbordada primera ingesta de datos syslog



Fuente: elaboración propia

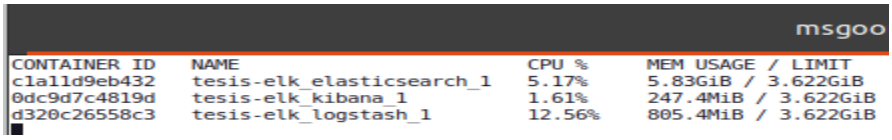
Ilustración 9: RAM desbordada primera ingesta de datos syslog



Fuente: elaboración propia

Una vez aprobado el nuevo hardware, se modifica el despliegue con las siguientes configuraciones https://gitlab.com/mejia.miguel.alexander/tesis-elk/-/blob/718d2968768a4c1196fec5f1151f0e1f0ef06114/docker-compose.yml, mismas que entregan un servidor estable que recibe datos de 1 nodo syslog.

Ilustración 10: Uso estable de recursos

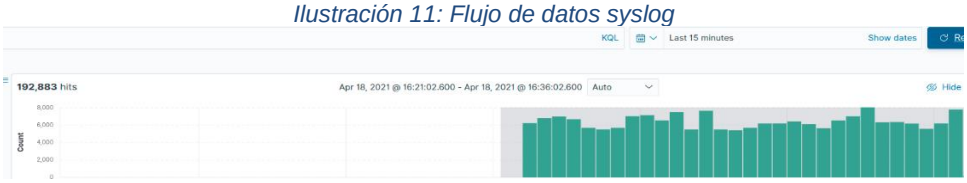


Fuente: elaboración propia

Al usar docker-compose fuera de un *swarm*, la etiqueta *deploy* es ignorada como lo menciona el hilo oficial del *issue* en github https://github.com/docker/compose/issues/4513, por cuanto se agrega el modificador --compatibility al comando docker-compose up -d para que este transforme internamente la sintaxis a la versión 2 de docker-compose y reconozca la etiqueta *deploy,* como un *walkaround* del problema de la versión 3 de docker-compose, cambio que será tratado como una deuda técnica para un sprint siguiente.

Incremento 4: Ya con un flujo de logs funcional, se siguen las configuraciones iniciales por defecto y se verifica en el menú de kibana la cantidad de logs que llegan desde el servidor monitoreado, 214 eventos por segundo aproximadamente.

Ilustración 11: Flujo de datos syslog



Fuente: elaboración propia

Se toma como muestra el primer log obtenido y mediante el lenguaje grok se define como se interpretará para almacenarse en elasticsearch.

Ilustración 12: Sentencia grok para el mensaje syslog del servidor dns unbound

```
[msgoon6@siem tesis-elk]$ cat logstash/pipeline/logstash.conf
input {
        beats {
                port => 5044
        }

        tcp {
                port => 5000
                type => syslog
        }
        udp {
                port => 5000
                type => syslog
        }
}

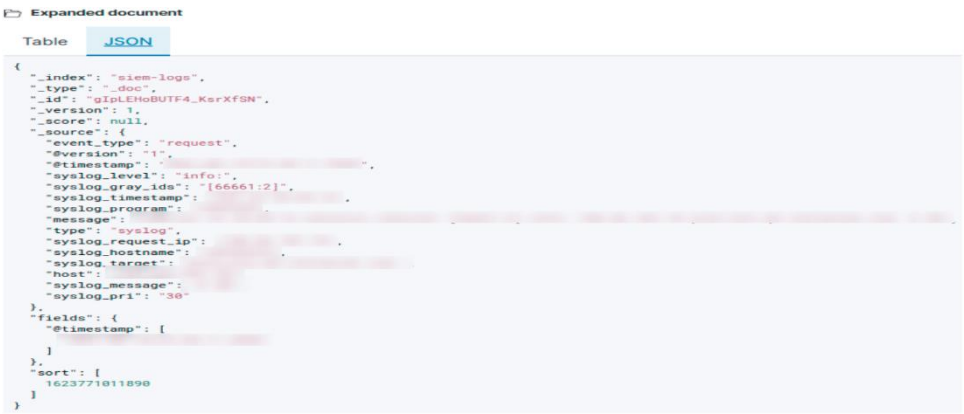
## Add your filters / logstash plugins configuration here

filter {
        if [type] == "syslog" {
                grok {
                        match => { "message" => "<%{POSINT:syslog_pri}>%{SYSLOGTIMESTAMP:syslog_timestamp} %{SYSLOGHOST:syslog_hostname} %{DATA:syslog_program}(?:\[%{POSINT:syslog_pid}\])?: %{GREEDYDATA:syslog_message}" }
                }
        }
}
```

Fuente: elaboración propia

Lo cual produce un flujo de datos formateados a elasticsearch, mientras que los errores se almacenarán en un archivo de logs, mismos que serán analizados para poder agregar más filtros y mejorar la cobertura, el siguiente filtro recibe correctamente los logs necesarios para el análisis https://gitlab.com/mejia.miguel.alexander/tesis-elk/-/blob/d201458c8ac5d3c75d2093e99cebfa1e3ea8feca/logstash/pipeline/logstash.conf, mismo que ya contiene los campos requeridos.

Ilustración 13: Campos mensaje dns procesado



Fuente: elaboración propia

Con el funcionamiento correcto de este primer servidor dns, se agregan los demás servidores con alto tráfico, se alcanzan aproximadamente 1500 eventos por segundo con las siguientes configuraciones https://gitlab.com/mejia.miguel.alexander/tesis-elk/-

/blob/7895f780dbb71dcb27abb8b6427f4e5a891cda0d/logstash/pipeline/logstash.conf.

Incremento 5: Ya con el monitoreo de los servidores dns en correcto funcionamiento, la institución requiere monitorear una serie de *honeypots* que mantiene, mismos que están centralizados en un origen particular capaz de reenviarlos únicamente en el protocolo gelf, por cuanto se genera un *listener gelf* en *logstash.*

Por recomendación de elastic, cada origen se establece en un archivo de *pipeline* independiente, por cuanto se tiene uno para syslog y uno para gelf https://gitlab.com/mejia.miguel.alexander/tesis-elk/-/tree/master/logstash/pipeline.

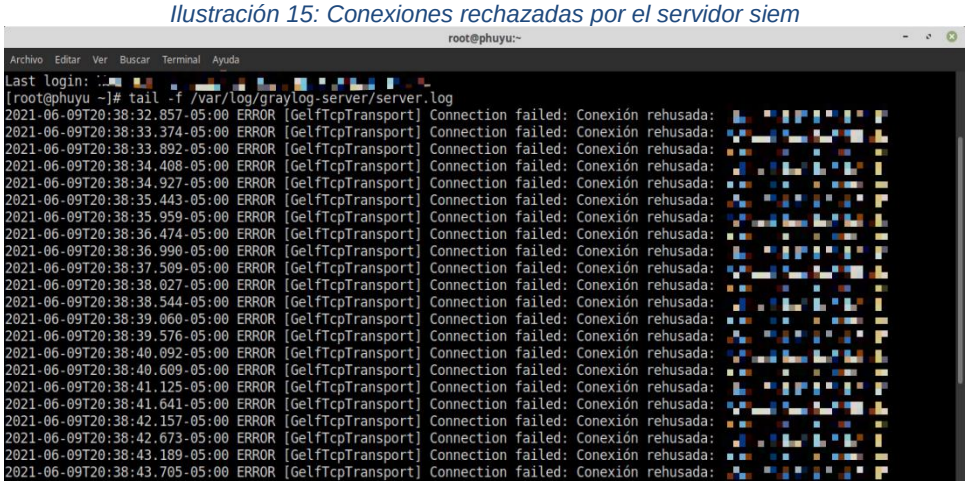
La historia de usuario 9, especificaba que los logs se recibirían por UDP, por cuanto se estableció la configuración requerida

Ilustración 14: Logstash con el protocolo UDP

Fuente: elaboración propia

Pero al iniciar la ingesta de datos el servidor rechazaba las conexiones

Ilustración 15: Conexiones rechazadas por el servidor siem



Fuente: elaboración propia

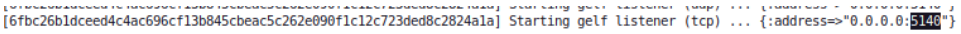
Se prueban diferentes configuraciones para lograr establecer una conexión, pero los logs seguían sin ser accesibles.

Ilustración 16: Conexión exitosa UDP

Fuente: elaboración propia

El problema se solventó con el cambio del protocolo de comunicación a tcp.

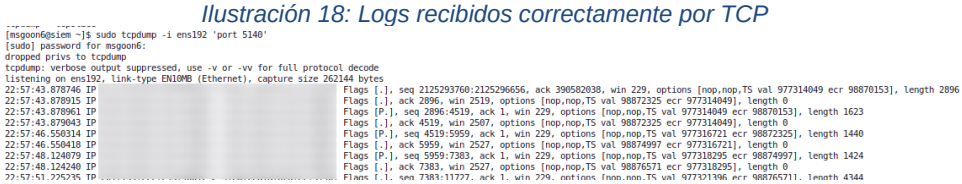
Ilustración 17: Logstash en TCP



Fuente: elaboración propia

Y así los logs comenzaron a interpretarse correctamente, ya con el servidor funcional, se reciben 18 000 reportes diarios aproximadamente.

Ilustración 18: Logs recibidos correctamente por TCP



Fuente: elaboración propia

Incremento 6: Al ser datos que contienen información sensible de la institución como de sus miembros, se requiere agregar seguridad a la comunicación, por cuanto se habilita la clave compartida de encriptación para kibana https://gitlab.com/mejia.miguel.alexander/tesis-elk/-/blob/7895f780dbb71dcb27abb8b6427f4e5a891cda0d/kibana/config/kibana.yml.

Para definir la línea base de funcionamiento, se analizan las diferentes opciones disponibles, el primer punto a tratar fue el nuevo módulo de SIEM disponible en elasticsearch https://www.elastic.co/es/siem, mismo que al probarlo no satisface la necesidad del proyecto, probablemente por su poco tiempo en producción, se prueba también las soluciones individuales de elastic para la definición y seguimiento de líneas de tendencia con los siguientes resultados.

Elastic Machine Learning: La licencia *comunity* solo permite la visualización de datos y no ninguna clase de procesamiento de los mismos.

Ilustración 19: Limitación Elastic Machine Learning

Fuente: elaboración propia

Elastic *Dataflow*: No permite el análisis de índices existentes en elasticsearch, sino que únicamente puede conectarse con agentes, por cuanto tampoco es viable.

Ilustración 20: Orígenes de datos Elastic Dataflow

Fuente: elaboración propia

Por esta razón y para aprovechar las capacidades del equipo de investigación, se decide con la institución que el componente requerido será desarrollado con las siguientes responsabilidades.

1. Definir el comportamiento normal de los diferentes servidores
2. Analizar el comportamiento permanentemente en busca de anormalidades
3. Reportar las anormalidades con un detalle de los eventos

El desarrollo genera un api alojado en el siguiente repositorio, https://gitlab.com/mejia.miguel.alexander/SIEM, mismo que, al momento consolida los eventos por tipo y por origen de cada minuto lo que vuelve más manejable la

información, que semanalmente alcanza los 150gb de logs (siem-logs), mientras que consolidada aproximadamente 300mb (syslog-stats).

Ilustración 21: Estado de los índices de elasticsearch

Name	Health	Status	Primaries	Replicas	Docs count	Storage size	Data stream
dns-request-address	yellow	open	1	1	1429	43.6mb	
siem-logs	yellow	open	1	1	576631573	138gb	
sense-detail	yellow	open	1	1	10	333.8kb	
error-events	yellow	open	1	1	399	55.3kb	
metrics-endpoint.metadata_current_default	green	open	1	0	0	208b	
syslog-stats	yellow	open	1	1	4908463	297.8mb	
gelf-logs	yellow	open	1	1	82236	70.6mb	

Fuente: elaboración propia

Se genera una imagen docker para poder ser agregada al docker-compose de producción y se extraen las configuraciones al mismo, con el siguiente resultado https://gitlab.com/mejia.miguel.alexander/tesis-elk/-/blob/888c4e75a23b2c054b04004b0c7d2605b1236fca/docker-compose.yml, se procesa toda la información obtenida desde el mes de febrero para poder liberar el espacio disponible en el servidor.

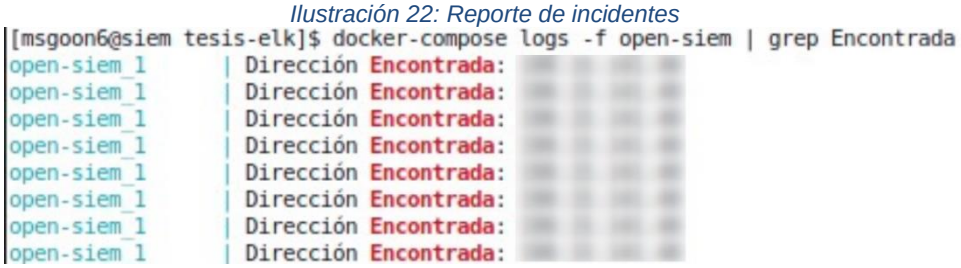
Incremento 7: Para no interferir en la ingesta de datos por su amplio volumen, se utiliza la funcionalidad de asíncronos implementada en el jdk11, con lo que en primer plano se ejecutará la consolidación de logs mientras que en segundo plano se dejará en ejecución un hilo independiente que realice el análisis de tendencia en base a la desviación standard de los datos considerados en la muestra, se establecen 2 modos de funcionamiento, estricto y abierto, mismos que pueden ser cambiados en cualquier momento desde las variables del archivo docker-compose.yml.

La detección de los incidentes a este punto se imprime en consola para poder verificar su correcto funcionamiento, deuda técnica que se resolverá en el sprint siguiente, según la definición de un SIEM *(Security Information) and (Event Management)*, el desarrollo hasta este punto cubre con la gestión de eventos requerida, por lo que, la última funcionalidad, información de seguridad, será tratada en el último sprint.

Incremento 8: Por las características de la institución (CEDIA) y los servicios disponibles para correlación, se decide que los incidentes valiosos de seguridad a detectar son aquellos generados por alguna institución miembro que utiliza los servidores dns y que produzca tráfico hacia algún *honeypot* monitoreado.

Como resultado del incremento anterior, existe un api de análisis, para este análisis adicional se utiliza el mismo con otro hilo independiente https://gitlab.com/mejia.miguel.alexander/SIEM/-/blob/master/src/main/java/com/msgoon6/siem/service/impl/SiemServiceImpl.java, el api se pone en funcionamiento y se realiza una prueba del mismo con una petición manual que es reportada correctamente por la herramienta.

Ilustración 22: Reporte de incidentes



Fuente: elaboración propia

Además, para pagar la deuda técnica dejada en el sprint anterior se genera un servicio responsable de emitir las notificaciones a un *end point* definido por la institución, mismo que también será configurable desde el archivo docker-compose.yml https://gitlab.com/mejia.miguel.alexander/SIEM/-/blob/master/src/main/java/com/msgoon6/siem/service/impl/ReportServiceImpl.java.

Para la correlación de eventos en consideración al requerimiento de la institución, se sigue el siguiente algoritmo de detección:

1. Ubicados dentro de *minuteSchedule* y después de haber disparado los métodos asíncronos de resumen y análisis de anormalidades.
2. Se obtienen todos los registros generados por los *honeypot,* agrupados por dirección ip de origen.
3. Se toma el conjunto de eventos *syslog* correspondiente al lapso de análisis.

4. Por cada registro *syslog* de los clientes DNS (universidades miembros que usan el servicio DNS) se buscan los errores registrados en los *honeypot*.
5. Se detecta si alguna universidad miembro genera tráfico hacia los *honeypot.*
6. En el caso de encontrar una incidencia de seguridad, esta se almacena en la colección correspondiente.
7. Se verifica si la misma no fue reportada en el transcurso del día, para evitar así saturación del canal de comunicaciones durante el tiempo que tarde el equipo en resolver el problema notificado.
8. Si no ha sido reportado, se envía al *end point* configurado.

Finalmente se genera una colección en elasticsearch para almacenar los detalles de las incidencias detectadas para su posterior consulta, mismos que revelan la detección de un incidente real reportado a la institución el día 13 de junio de 2021 a las 10 y 18 de la mañana.

Ilustración 23: Primer incidente real detectado

Fuente: elaboración propia

2.2.7. Sprint Review

Es una reunión a la que asiste el equipo scrum, así como cualquier otro invitado que el *product owner* considere importante, y sirve para socializar las nuevas funcionalidades obtenidas durante el sprint, la guía oficial recomienda no manejarla como un *check list* de cumplimiento, con sprints de 1 mes de duración se recomiendan reuniones de 2 horas, en este caso al ser tiempos más cortos de trabajo, la reunión se estableció de 30 minutos.

Durante estas reuniones al ser un producto incremental, se entregaron a la institución los productos funcionales detallados en el apartado de incrementos, de los cuales aún sin haber finalizado el desarrollo ya se tuvo información valiosa para resolver incidentes, por cuanto la cantidad de los mismos fue significativamente menor durante las futuras mediciones.

2.2.8. Sprint Retrospective

Inmediatamente finalizado el *review* y con el fin de iterar, no solo en funcionalidades sino también en el comportamiento del equipo se discuten principalmente 3 ejes representados en 3 frases.

1. Seguir Haciendo *(Keep Doing),* en donde se colocan las prácticas y actividades que fortalecen al equipo y que han mostrado buenos resultados.
2. Dejar de Hacer (*Stop Doing*), en donde se colocan los problemas suscitados durante la iteración para solventarlos y reducir su impacto.
3. Comenzar a hacer (*Start Doing*), en donde irán las propuestas del equipo de cómo se podría mejorar el proceso.

Una vez atendidos estos puntos se definen elementos de acción, los cuales serán procedimientos puntuales a seguir para resolver los 3 puntos tratados, scrum establece un tiempo limitado a máximo tres horas para un sprint de un mes, por las características del presente proyecto el evento se lo estableció en máximo 30 minutos de duración. A continuación, se detalla el extracto de estas reuniones.

Tabla 5: Sprint retrospective

Sprint	**Fecha terminación del Sprint**	***Retrospectiva***
1	27/09/2020	Seguir Haciendo Selección de herramientas web colaborativas para fomentar el trabajo remoto. Seleccionar proyectos en actualización constante con licencias de uso claras, para evitar perder el soporte a corto plazo. Dejar de hacer Priorizar herramientas conocidas por el equipo. Comenzar a hacer Analizar el comportamiento de recursos hardware de cualquier componente que se implemente en la arquitectura propuesta.

2	11/10/2020	Seguir Haciendo Utilización de docker y docker-compose Dejar de hacer Probar repositorios que soliciten permisos elevados de ejecución como requisitos. Comenzar a hacer Definir los volúmenes necesarios para preservar la información.
3	19/02/2021	Seguir Haciendo Monitoreo de recursos en el servidor. Comunicación inmediata para cambiar un requerimiento funcional por uno técnico. Dejar de hacer Pruebas en horarios laborales, desarrollos fuera del horario acordado. Comenzar a hacer Agregar al monitoreo del uso de recursos el comportamiento de los índices de elasticsearch para el alto volumen de información.
4	05/03/2021	Seguir Haciendo Uso de lenguajes estándar como grok en lugar de soluciones alternativas. Considerar también los eventos incomprensibles con la etiqueta *unknown* para poder ser analizados posteriormente. Dejar de hacer Consultas KQL sobre toda la data porque estresa a elasticsearch restándonos rendimiento. Comenzar a hacer Definir una política de limpieza o alternativa de solución para el límite de 50gb por índice.
5	16/04/2021	Seguir Haciendo Separar los archivos de pipeline por funcionalidad. Dejar de hacer Pruebas en producción, dado que los segundos que tarda el servicio en restituirse representan una acumulación de logs que podrían alterar la integridad de los mismos. Comenzar a hacer Monitorear el comportamiento del nuevo índice dedicado a eventos gelf.

6	30/04/2021	Seguir Haciendo Uso del api *stream* de java y uso de buenas prácticas de desarrollo. Dejar de hacer Probar funcionamientos con impresiones en consola. Comenzar a hacer Agregar un *logger* para la emisión de mensajes por consola.
7	14/05/2021	Seguir Haciendo Utilización de asíncronos para no interferir en la ingesta de datos. Seguir el principio de responsabilidad única en servicios. Dejar de hacer Probar funcionamientos con impresiones en consola. Comenzar a hacer Agregar un *logger* para la emisión de mensajes por consola.
8	28/05/2021	Seguir Haciendo Seguir el principio de responsabilidad única en servicios. Dejar de hacer No aplica, fin del proyecto. Comenzar a hacer No aplica, fin del proyecto.

Fuente: elaboración propia

2.3. Metodología de la Investigación

Al considerar que la implementación de la solución SIEM establecerá un antes y un después en el entorno de las PYMES, el impacto de la misma debe ser medido y analizado, por lo que se escoge el enfoque experimental para la investigación.

2.3.1. Planteamiento del problema

El problema científico de la presente investigación se define como: Limitada asequibilidad de las pequeñas y medianas empresas a herramientas *"Security Information and Event Management"*, por sus elevados costos.

2.3.2. Planteamiento de la hipótesis

La Implementación de una herramienta *Security Information and Event Management* basada en open source, mejorará el acceso a la gestión de incidentes de seguridad en pequeñas y medianas empresas.

2.3.3. Definición de variables

Variable Dependiente: Acceso a la gestión de incidentes de seguridad

Variable Independiente: Implementación de la solución SIEM

2.3.4. Operacionalización de variables

Tabla 6: Operacionalización de las variables dependientes

Tipo	Variable	Descripción	Indicador	Descripción	Técnicas	Instrumentos
Dependiente	Acceso a la gestión de incidentes de seguridad	Relacionada a la asequibilidad, mencionada en el punto 1.6, se define como un valor cualitativo que representa la capacidad de conseguir o adquirir un bien o servicio, al existir herramientas diversas en el mercado además de la desarrollada en esta investigación, el apartado de adquisición definido por los costos de implementación y de mantenimiento es el que necesita analizarse.	Costo de Implementación	Valor monetario en dólares americanos que representa el presupuesto requerido para la primera puesta en marcha de la herramienta.	Observación directa	Registros
			Costo de Mantenimiento	Valor monetario en dólares americanos que representa el presupuesto requerido para mantener la herramienta en óptimo funcionamiento a través del tiempo.	Observación directa	Registros

Fuente: elaboración propia

Tabla 7: Operacionalización Variable Independiente

Tipo	Variable	Descripción	Método	Herramientas
Independiente	Implementación de la solución SIEM	Desarrollo de una solución software que permita la gestión y seguimiento de los logs de una organización.	SCRUM	-Entorno integrado de desarrollo -Docker

Fuente: elaboración propia

2.3.5. Procedimiento y recolección de datos

Para la fase inicial de investigación, es decir, el análisis de las soluciones existentes en el mercado y su viabilidad para el caso de pruebas de la investigación, se utilizaron las técnicas de observación y búsqueda de información bibliográfica, enfocada principalmente en sitios web que traten sobre las características de las mismas.

El análisis de costos necesario para tratar las 2 variables dependientes se realizó mediante una investigación bibliográfica, se consideraron como fuente principal de información, los sitios web de ventas de las soluciones analizadas o en su defecto, los sitios oficiales de empresas como G2 que se dedican a realizar comparativas empresariales de costos.

CAPÍTULO III. ANÁLISIS DE LOS RESULTADOS DE LA INVESTIGACIÓN

A continuación, se presentan los resultados más relevantes del trabajo de investigación, detallados de la siguiente manera:

- Resultados del desarrollo, se detalla la arquitectura resultante de la herramienta y las consideraciones técnicas de la misma.
- Resultados operativos, se detalla el funcionamiento en producción de la herramienta y se demuestra su utilidad para la empresa al lograr identificar incidentes de seguridad tanto de pruebas como reales.
- Resultados estadísticos de la hipótesis de la investigación.
- Proceso de replicabilidad de la herramienta.

3.1. Resultados del desarrollo de la herramienta

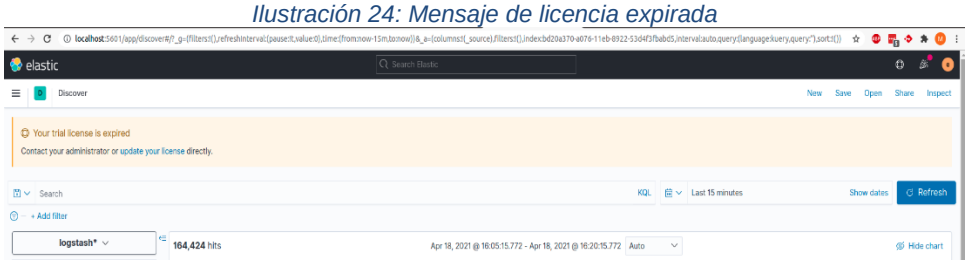
Se obtiene como resultado un SIEM desarrollado con herramientas y lenguajes open source, mismos que se encuentran consolidados en el siguiente repositorio público https://gitlab.com/mejia.miguel.alexander/tesis-elk, mismo que utiliza un servicio creado especialmente para el análisis, la imagen docker a utilizarse se encuentra en el siguiente container registry público registry.gitlab.com/mejia.miguel.alexander/siem y su código fuente en el repositorio correspondiente https://gitlab.com/mejia.miguel.alexander/SIEM, misma que muestra *un up time* del 100% en los períodos entre nuevos lanzamientos.

Como se expuso en apartados anteriores, debido a la elevada cantidad de registros por segundo que procesa la herramienta, el uso de ram y procesador son un factor determinante durante la ingesta de datos, por cuanto, se recomienda seguir la configuración de recursos predeterminada en el archivo docker-compose para obtener resultados similares en condiciones similares https://gitlab.com/mejia.miguel.alexander/tesis-elk/-/blob/master/docker-compose.yml, además, establecer las configuraciones en las variables de ambiente ES_JAVA_OPTS y LS_JAVA_OPTS para elasticsearch y logstash respectivamente en lugar de hacerlo a nivel de docker, por el problema de versiones mencionado en el incremento 3.

Con el uso permanente, la herramienta muestra la necesidad de realizar ciertas configuraciones de manera manual, mismas que se detallan a continuación.

1. Incrementar el límite máximo de campos a consultar en elasticsearch, para lo cual modificaremos la propiedad "index.mapping.total_fields.limit" a un valor de 2000
2. La licencia debe ser configurada en su versión *comunity* en lugar de la prueba gratuita que se establece por defecto, este proceso se encuentra ya implementado en el repositorio, pero por un bug de kibana la notificación aparece eventualmente sin interferir con el funcionamiento de la herramienta, para quitar el mensaje falso positivo basta con seguir las instrucciones en el propio mensaje.

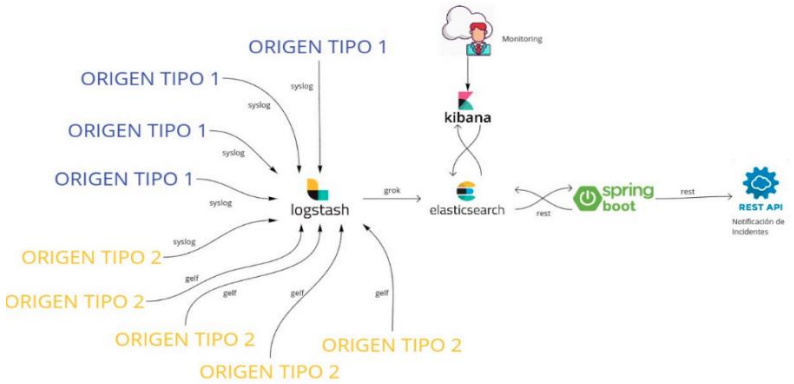
Ilustración 24: Mensaje de licencia expirada



Fuente: elaboración propia

La arquitectura resultante de la aplicación se encuentra disponible y permanentemente actualizada en el siguiente enlace https://miro.com/app/board/o9J_l_VBkpY=/, al momento de escrito este documento es la siguiente.

Ilustración 25: Arquitectura final de la solución



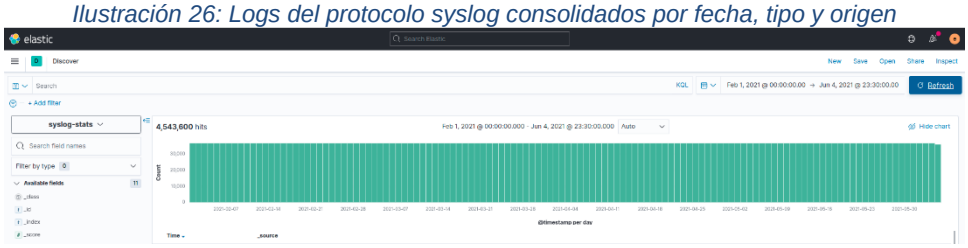
Fuente: elaboración propia

Para el uso de la herramienta y por seguridad de la información, se establece que la manera de acceder a los componentes web de la misma sea mediante túneles ssh, lo que incrementa el nivel de protección de la información sensible almacenada y procesada por la herramienta.

3.2. Resultados Operativos

La herramienta muestra un comportamiento estable desde el día 1 de febrero de 2021 con un flujo constante y homogéneo de logs consolidados como lo muestra la siguiente imagen

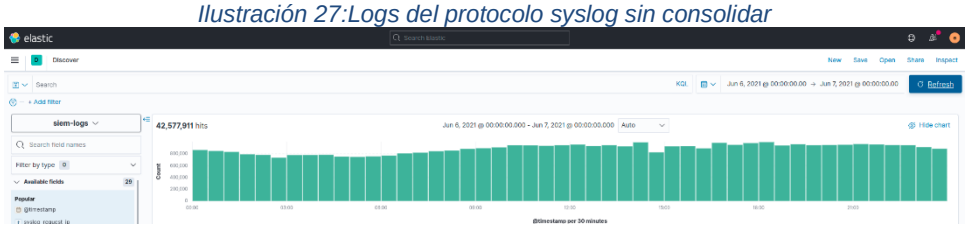
Ilustración 26: Logs del protocolo syslog consolidados por fecha, tipo y origen



Fuente: elaboración propia

Mientras que la información íntegra de logs fluye de manera constante en un aproximado de 40 millones de registros al día en syslog

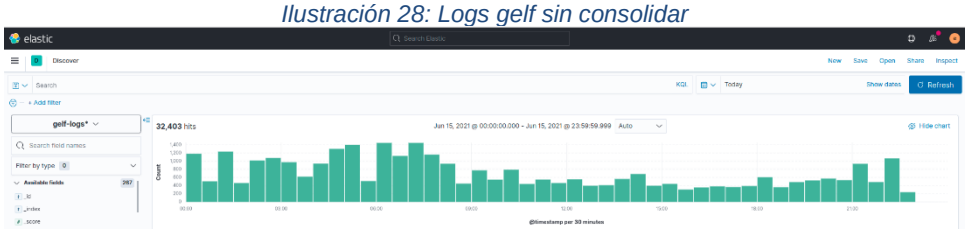
Ilustración 27:Logs del protocolo syslog sin consolidar



Fuente: elaboración propia

Y 30 mil registros incidentes en gelf

Ilustración 28: Logs gelf sin consolidar



Fuente: elaboración propia

En cuanto al reporte de incidencias de seguridad, se tienen 3 reportes en la semana comprendida entre el 8 de junio de 2021 y el 15 de junio de 2021.

Ilustración 29: Incidentes de seguridad detectados

Fuente: elaboración propia

Mientras que en el análisis de comportamiento se tienen 1639 anomalías en el comportamiento del flujo de logs.

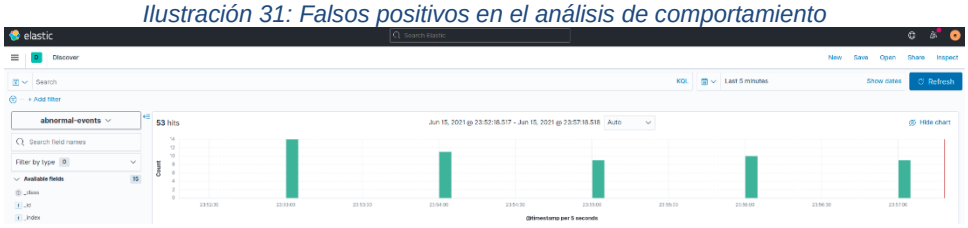
Ilustración 30: Anormalidades en el comportamiento de los servidores

Fuente: elaboración propia

Dado que, una anomalía puede ser permanente y tardar un tiempo en solucionarse, la herramienta enviará una sola notificación al día por cada caso que detecte, para evitar un ataque de denegación de servicios involuntario al receptor de notificaciones, así como evitar la saturación de la red.

Finalmente, con el api en correcto funcionamiento se determina que, por la naturaleza de los datos, la desviación standard es un valor demasiado cerrado para identificar anormalidades, puesto que por minuto detecta hasta 10 incidentes que la institución declara como falsos positivos.

Ilustración 31: Falsos positivos en el análisis de comportamiento



Fuente: elaboración propia

Por lo que se acuerda ampliar este espectro con un modo no estricto, configurable desde el archivo docker-compose, que permitirá que el rango de tolerancia sean 2 veces el valor de la desviación estándar, lo que reduce los falsos positivos y vuelve útil y eficiente el uso de la herramienta para la institución.

3.3. Resultados estadísticos de la hipótesis de la investigación

En base a la definición establecida en el punto 1.6 sobre asequibilidad, la existencia de la herramienta queda cubierta con el resultado del punto anterior, por cuanto el presente apartado se enfocará únicamente en el presupuesto requerido.

HIPOTESIS NULA Ho: La Implementación de una herramienta *Security Information and Event Management* basada en open source, no mejorará el acceso a la gestión de incidentes de seguridad en pequeñas y medianas empresas.

HIPOTESIS ALTERNATIVA Ha: La Implementación de una herramienta *Security Information and Event Management* basada en open source, mejorará el acceso a la gestión de incidentes de seguridad en pequeñas y medianas empresas.

NIVEL DE SIGNIFICANCIA: 0.05.

3.3.1. Comparativa de costos

Dado que, los modelos de negocio actuales estiman necesario un ingreso inicial y un ingreso permanente para obtener una rentabilidad en el transcurso del tiempo se toman como base estos 2 costos para la comparación, expresados como costo de implementación, que será definido como el presupuesto directo necesario para la primera puesta en marcha de la herramienta y el costo de mantenimiento, es este el presupuesto necesario anual para que la herramienta continúe en normal funcionamiento durante el transcurso del tiempo.

Las herramientas a considerar para esta comparativa son aquellas de las que se logró obtener información de sus costos en fuentes confiables y oficiales como las páginas web de cotización de los fabricantes de las herramientas, en consideración a que el enfoque de este estudio son las PYME, se opta por el plan más económico disponible de cada una, además, que al menos cubran en su oferta los 9 orígenes diferentes de

datos que fueron analizados en la institución de pruebas, finalmente en caso de disponer de una versión vendida como servicio, esta no se consideró al ser un modelo de software como servicio diferente a la solución propuesta.

En consideración a que ninguna de las herramientas evaluadas incluye dentro del paquete la configuración del servicio, así como tampoco de los productores de logs, el costo indirecto relacionado a la curva de aprendizaje de las herramientas y la implementación de las mismas queda totalmente del lado de la empresa interesada, tanto en las soluciones existentes como en la desarrollada en esta investigación, por cuanto, según Fernández (2018), con las necesidades claras y la herramienta seleccionada se requieren 44 días para la puesta en marcha de un SIEM y 22 días para sus pruebas y ajustes en promedio, mismos que se traducen a 3 meses de trabajo, mismo que multiplicado por 1500 dólares que es el salario promedio de un experto en sistemas en Ecuador según Glassdoor (2021), resultan en un costo de implementación para cualquiera de las soluciones de aproximadamente 4500 dólares. Para costo de mantenimiento, como se explicó en el Incremento 6, el principal elemento computacional es el almacenamiento en disco, ya que con información volátil (siem-logs) se alcanzan a utilizar 150gb a la semana, mismo que consolidados reducen a 300mb (syslog-stats), dándonos así un uso por origen estimado promedio de 17gb semanales volátiles y 35mb que representan 1.78gb anuales persistentes.

Según la investigación de Tenelema (2020), el almacenamiento en un servidor local se puede estimar en el país durante el año 2021 en 1 dólar por gb, por lo que el presupuesto requerido por sitio sería de 18.78 dólares anuales, 17gb para espacio de trabajo volátil, dado que, se borran semanalmente y 1.78 dólares para almacenamiento persistente.

Tabla 8: Comparativa de costos anuales de mantenimiento

Proveedor	Cantidad máxima de orígenes	Costo de Mantenimiento anual estimado	Costo de Mantenimiento anual por sitio estimado	Referencias
FortiSIEM	300	$ 18406.88	$ 61.36	(itprice, 2021)

SolarWinds	30	$ 2613.00	$ 87.10	(solarwinds, 2021)
ManageEngine EventLog Analyzer	100	$ 2495.00	$ 24.95	(ManageEngine, 2021)
Splunk Enterprice Security	25	$ 1725.00	$ 69.00	(splunk, 2021)
IBM QRadar	450	$ 9600.00	$ 21.33	(g2.com, 2021)
RSA NetWitness	100	$ 10284.00	$ 102.84	(eSecurityPlanet, 2021)
Solución Propuesta	9	$ 169.02	$ 18.78	-
Promedio	167.5	$ 6470.41	$ 55.05	-

Fuente: elaboración propia

Al tener un conjunto de datos pareado, lo primero que se requiere es verificar si se utilizará un método paramétrico o no paramétrico, por cuanto se procede con la prueba de normalidad de Shapiro-Wilk con un nivel de significancia de 0.05.

Ilustración 32: Prueba de normalidad

```
> normalityTest(~costo_mantenimiento, test="shapiro.test", data=costos)

        Shapiro-Wilk normality test

data:  costo_mantenimiento
W = 0.89311, p-value = 0.2913
```

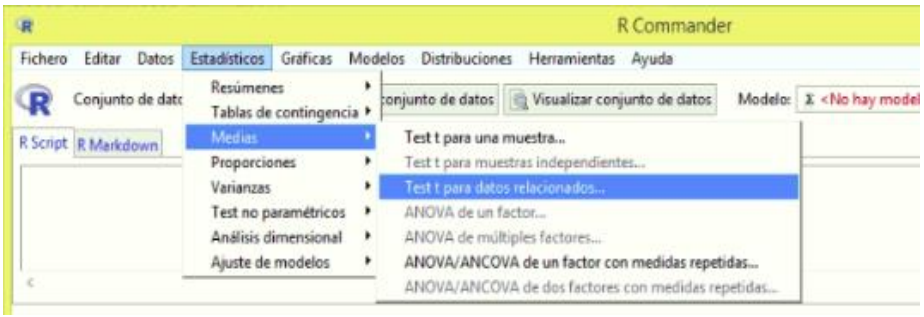
Fuente: elaboración propia

Con un p valor de 0.2913 > 0.05, se deduce que los datos siguen una distribución normal, por tal motivo para su análisis se aplica t de Student, mediante el software R como se demuestra en las siguientes ilustraciones.

Ilustración 33: Costos en el software estadístico R

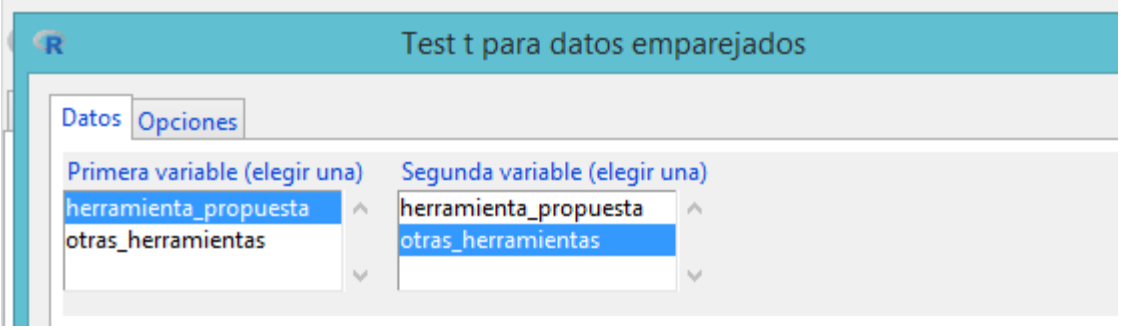
Fuente: elaboración propia

Ilustración 34: Selección de test T en el software R



Fuente: elaboración propia

Ilustración 35: Selección de Variables para el test T



Fuente: elaboración propia

Ilustración 36: Comando para el test T en R

```
```{r}
with(costos, (t.test(herramienta_propuesta, otras_herramientas, alternative='two.sided',
  conf.level=.95, paired=TRUE)))
```
```

Fuente: elaboración propia

Ilustración 37: Prueba de hipótesis - Costos de implementación

```
> with(costos, (t.test(herramienta_propuesta, otras_herramientas, alternative='two.sided',
+   conf.level=.95, paired=TRUE)))

        Paired t-test

data:  herramienta_propuesta and otras_herramientas
t = -3.1631, df = 5, p-value = 0.02501
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -76.70608  -7.92725
sample estimates:
mean of the differences
              -42.31667
```

Fuente: elaboración propia

Análisis: Al aceptar un error de 0.05 equivalente al 95% de acierto, con un p valor igual a 0.02205 menor al nivel de significancia del 0.05, se decide aceptar la hipótesis alternativa: “La Implementación de una herramienta *Security Information and Event Management* basada en open source, mejorará el acceso a la gestión de incidentes de seguridad en pequeñas y medianas empresas”, lo que permite concluir que, la herramienta creada es más asequible que la competencia generando un ahorro de entre 1555.98 dólares americanos y 18237.86 dólares americanos.

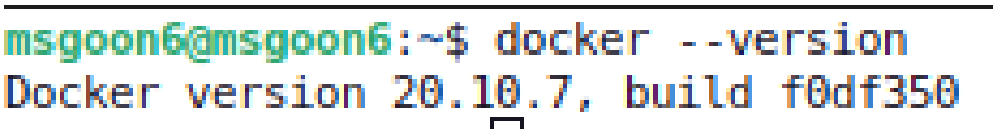
3.4. Proceso de Replicabilidad

Con la herramienta validada por el cliente y estadísticamente demostrado que es más asequible para las PYME, al ser un proyecto open source que pretende ser masificado en la comunidad, es necesario que el mismo sea fácilmente replicable, para lo cual desde el inicio del proyecto se planteó docker-compose como herramienta para la organización de los contenedores y configuraciones requeridas.

La siguiente guía de replicabilidad especifica los pasos a seguir para poner en funcionamiento la herramienta.

1. Instalar docker (versión 17.05 o superior).

Ilustración 38: Versión de docker



Fuente: elaboración propia

2. Instalar docker-compose (versión 1.20.0 o superior).

Ilustración 39: Versión de docker-compose

Fuente: elaboración propia

3. Clonar el repositorio https://gitlab.com/mejia.miguel.alexander/tesis-elk

Ilustración 40: Repositorio clonado localmente

```
msgoon6@msgoon6:~/workspace$ git clone https://gitlab.com/mejia.miguel.alexander/tesis-elk
Clonando en 'tesis-elk'...
warning: redirigiendo a https://gitlab.com/mejia.miguel.alexander/tesis-elk.git/
remote: Enumerating objects: 930, done.
remote: Counting objects: 100% (930/930), done.
remote: Compressing objects: 100% (582/582), done.
remote: Total 930 (delta 287), reused 910 (delta 267), pack-reused 0
Recibiendo objetos: 100% (930/930), 17.30 MiB | 8.66 MiB/s, listo.
Resolviendo deltas: 100% (287/287), listo.
msgoon6@msgoon6:~/workspace$
```

Fuente: elaboración propia

4. Ingresar a la carpeta del repositorio llamada tesis-elk.

Ilustración 41: Ingreso a la carpeta dedl proyecto

```
msgoon6@msgoon6:~/workspace$ cd tesis-elk/
msgoon6@msgoon6:~/workspace/tesis-elk$
```

Fuente: elaboración propia

5. Revisar las configuraciones en el archivo docker-compose.yml.

Ilustración 42: Archivo de configuraciones

```
msgoon6@msgoon6:~/workspace/tesis-elk$ cat docker-compose.yml
version: '3.2'

services:
  elasticsearch:
    build:
      context: elasticsearch/
      args:
        ELK_VERSION: $ELK_VERSION
    volumes:
      - type: bind
        source: ./elasticsearch/config/elasticsearch.yml
        target: /usr/share/elasticsearch/config/elasticsearch.yml
        read_only: true
      - type: bind
        source: ./elasticsearch/data
        target: /usr/share/elasticsearch/data
    ports:
      - "9200:9200"
      - "9300:9300"
    environment:
      ES_JAVA_OPTS: "-Xmx12288m -Xms1288m"
      ELASTIC_PASSWORD: changeme
      # Use single node discovery in order to disable production mode and avoid bootstrap checks.
      # see: https://www.elastic.co/guide/en/elasticsearch/reference/current/bootstrap-checks.html
      discovery.type: single-node
    networks:
      - elk
    restart: unless-stopped
```

Fuente: elaboración propia

6. Revisar los pipelines requeridos en la carpeta logstash/pipeline, quitar o agregar alguno en caso de requerirlo, así como los puertos y protocolos.

Ilustración 43: Archivo de pipeline gelf

```
msgoon6@msgoon6:~/workspace/tesis-elk$ cat logstash/pipeline/gelf.conf
input {
        gelf {
                port => 5140
                type => gelf
                use_tcp => true
        }
}

output {
        if [type] == "gelf" {
                elasticsearch {
                        hosts => "elasticsearch:9200"
                        user => "elastic"
                        password => "changeme"
                        index => "gelf-logs"
                        ecs_compatibility => disabled
```

Fuente: elaboración propia

Ilustración 44: Archivo de pipeline syslog

```
msgoon6@msgoon6:~/workspace/tesis-elk$ cat logstash/pipeline/syslog.conf
input {
        tcp {
                port => 5000
                type => syslog
        }
        udp {
                port => 5000
                type => syslog
        }
}

## Add your filters / logstash plugins configuration here

filter {
        if [type] == "syslog" {
                if "unbound" in [message] {
                        if "sdns1" in [message] or "sdns2gye" in [message] or "sdns2uio" in [message] or "sdns2cue" in [message] {
                                if "TYPE0" in [message] {
                                        grok {
                                                match => { "message" => ["<%{POSINT:syslog_pri}>%{SYSLOGTIMESTAMP:syslog_timestamp} %{S
24SD:syslog_gray_ids} %{DATA:syslog_level} (?:%{DATA:syslog_request_ip}) %{GREEDYDATA:syslog_message}"] }
                                                add_field => {"event_type" => "unknown"}
                                        }
                                } else if "error:" in [message] or "failed" in [message] or "failed:" in [message] or "failure" in [mess
ailure" in [message] {
                                        grok {
                                                match => { "message" => ["<%{POSINT:syslog_pri}>%{SYSLOGTIMESTAMP:syslog_timestamp} %{S
24SD:syslog_gray_ids} %{DATA:syslog_level} %{GREEDYDATA:syslog_error}"] }
                                                add_field => {"event_type" => "error"}
                                                tag_on_failure => []
                                        }
                                } else if [message] =~ ".*\IN$" {
```

Fuente: elaboración propia

7. Levantar todo el ambiente con el comando docker-compose up –d.

Ilustración 45: Stak en funcionamiento en el equipo de pruebas

```
msgoon6@siem:~/workspace/tesis-elk
CONTAINER ID   NAME                      CPU %     MEM USAGE / LIMIT     MEM %     NET I/O           BLOCK I/O         PIDS
02ab55106462   tesis-elk_open-siem_1     0.20%     620.4MiB / 15.43GiB   3.98%     75.4MB / 1.13MB   16.7MB / 0B       68
b4d1af38ef1a   tesis-elk_kibana_1        0.06%     223.3MiB / 15.43GiB   1.41%     3.04MB / 1.93MB   1.29MB / 0B       12
20f0e980132e   tesis-elk_logstash_1      35.00%    1.432GiB / 15.43GiB   9.28%     16.3MB / 51.9MB   0B / 0B           98
374ee5887c2c   tesis-elk_elasticsearch_1 41.46%    2.407GiB / 15.43GiB   15.59%    53.7MB / 79.7MB   89.4MB / 194MB    170
```

Fuente: elaboración propia

Esta información también se encuentra disponible en el archivo README del repositorio para que pueda ser accedida públicamente.

CONCLUSIONES

1. El análisis teórico de los componentes de las herramientas SIEM propietarias, permitió determinar que las soluciones existentes cubren funcionalidades muy distintas entre sí, por cuanto el levantamiento de requerimientos con el cliente fue fundamental para decidir las funcionalidades a implementar.
2. La definición de funcionalidades requeridas mediante el marco de trabajo ágil scrum, demuestra que, el agilismo fue fundamental para este proyecto, dado que, permitió que el equipo reaccione y cambie la prioridad de las funcionalidades para resolver problemas como el uso de recursos y comportamientos no deseados de la solución.
3. La aplicación de métodos para la recolección y análisis de resultados demostró que, la implementación de la solución permitió a la institución, durante el período de levantamiento de información, visibilizar 1639 anomalías en el comportamiento de los servicios monitoreados con fecha de corte 15 de junio de 2021 y 3 alertas de seguridad en la semana del 8 de junio de 2021 al 15 de junio de 2021.
4. Al diseñar el procedimiento para la replicabilidad de la herramienta por parte de la comunidad, se notó que el uso de software libre permite construir soluciones complejas que aprovechan las funcionalidades individuales existentes y que se complementan con procesos particulares desarrollados para conseguir soluciones integrales a cualquier escala con costos accesibles.

RECOMENDACIONES

1. Para futuras investigaciones se recomienda la implementación de pipelines adicionales para servicios populares como wordpress para poder masificar el uso de la herramienta.
2. En relación a la ingesta de data, se recomienda que desde el origen de *logs* ya se encuentre configurado un formato *standard* como syslog o gelf usados en esta implementación, para evitar así un sobre esfuerzo al tratar de acoplar la herramienta un formato personalizado con un *standard* propio.
3. Se recomienda realizar pruebas exhaustivas de rendimiento en diferentes configuraciones de hardware para poder definir requerimientos mínimos de la herramienta.
4. Para casos de uso con un volumen de información bajo, se recomienda considerar la implementación de una fórmula de detección adicional, dado que, en valores pequeños, al ser el conteo un valor entero y la desviación estándar un valor decimal, los redondeos han demostrado penalizar negativamente en la precisión.

BIBLIOGRAFÍA

Balarezo Chávez, A. F., & Poveda Pilatasig, D. X. (2015). *Propuesta de mejoramiento de la herramienta ossim siem (Open Source), para obtener los niveles óptimos de gestión en la administración de la seguridad, en una red implementada en cloud computing*. http://dspace.ups.edu.ec/handle/123456789/10101

Bussa, T., Sadowski, G., & Kavanagh, K. (2010). Critical Capabilities for Security Information and Event Management Technology. *Event (London)*, *May*, 16. https://www.gartner.com/en/documents/3894576/critical-capabilities-for-security-information-and-event

CEDIA. (2021). *CEDIA | LinkedIn*. LinkedIn. https://ec.linkedin.com/company/fundación-cedia

Chopra, M., & Mahapatra, C. (2019). Significance of security information and event management (SIEM) in modern organizations. *International Journal of Innovative Technology and Exploring Engineering*, *8*(7), 432–435.

Digital.ai. (2020). *14th Annual State of Agile Report*. Digital.Ai.

eSecurityPlanet. (2021). *RSA NetWitness Software Review & Analysis | RSA SIEM Tool*. https://www.esecurityplanet.com/products/rsa-netwitness/

Fernández, A. (2018). *IMPLEMENTACIÓN DE UN SISTEMA DE GESTIÓN DE EVENTOS DE SEGURIDAD DE UNA EMPRESA DE TAMAÑO MEDIO* [UNIVERSITAT POLITECNICA DE VALENCIA]. https://riunet.upv.es/bitstream/handle/10251/109765/Ramírez - Implementación de un sistema de gestión de eventos de seguridad en una empresa de tamañ....pdf

Fernández, J., Herrera, J., & Camilo, J. (2018). *IMPLEMENTACIÓN DE UN SECURITY INFORMATION AND EVENT MANAGEMENT –SIEM– EN EL COMANDO DE LA ARMADA NACIONAL. DIRECCIÓN DE TECNOLOGÍAS DE LA INFORMACIÓN Y LAS COMUNICACIONES*. http://polux.unipiloto.edu.co:8080/00003801.pdf

g2.com. (2021). *IBM Security QRadar Pricing 2021 | G2*. https://www.g2.com/products/ibm-security-qradar/pricing?__cf_chl_jschl_tk__=f76544e66ae463188426a29c41294658066e3ef1-1624159585-0-AdKIBcAn9YKH_5voXmddWUUnewLfU-veIybqx4QIJoOOLtJzRI6ji-_XqZDCdI6nnZIpLPWF3sNf9q-kXsXLdCYirc4EGlT1VjWNmAESQDQG1vaNB9JZBb0nakl6k4yFtB2qriTnoZSBe4Xd_J9iKUwQ7Rp-00j0h9rwQTsRCeWdoD4mGoVDxcnL6HXp6E83TvC_r7zDVOVkHXXnvtfHhjMulLpJAW5POUecyL-2aNRr4JFDiVCbazRF2hvoTxdnTnbHM-Ic8_hDETv-7FJ0V4uZrhlMxfYzFxRM75lJJ0OjmfMXycFchdqwivsL02y_XN4tDoCK-teEKCJPTopvdSDTtxikYnw_7_k3fNDPVpNOUDXA0R_xlm9izS-a4YHHeSZoFfu8bE2554HE1ILNyH_g6oIglHCTsiXNRy6rrmD6iKtS1ehlum9I5zjo5jb8SckAiMgx_M1VPo6TzSxBWMchzKnt-N4Nk-yd0QLGc_8qIEr4VkufczaP-IjoCj4K6g

Glassdoor. (2021). *Sueldo: Software Developer en Quito, Ecuador | Glassdoor*. https://www.glassdoor.com.mx/Sueldos/quito-software-developer-sueldo-SRCH_IL.0,5_IM1362_KO6,24.htm

IBM. (2020). *IBM QRadar SIEM - Visión general - España | IBM*. https://www.ibm.com/es-es/products/qradar-siem

itprice. (2021). *Precio Fortinet SIEM - Lista de precios Fortinet 2021*. https://itprice.com/es/fortinet-price-list/siem.html

Lima Torrico, O. E. (2019). *Monitorización de datos en tiempo real usando ELK y desarrollo de una web de visualización y gestión*. http://diposit.ub.edu/dspace/bitstream/2445/145478/2/memoria.pdf

ManageEngine. (2021). *Editions - Event Log Management Software - EventLog Analyzer*. https://www.manageengine.com/products/eventlog/eventloganalyzer-editions.html?index

Miloslavskaya, N. (2018). Analysis of siem systems and their usage in security

operations and security intelligence centers. *Advances in Intelligent Systems and Computing*, *636*, 282–288. https://doi.org/10.1007/978-3-319-63940-6_40

Nsit. (2020). *¿Qué es SIEM y cómo funciona? Alcance e implementación | Nsit*. https://www.nsit.com.co/que-es-siem-en-seguridad-informatica-alcance-e-implementacion/

Pico Barrera, F. M. (2016). *UNIVERSIDAD REGIONAL AUTÓNOMA DE LOS ANDES UNIANDES TESIS PREVIO A LA OBTENCIÓN DEL GRADO ACADÉMICO DE* [UNIVERSIDAD REGIONAL AUTÓNOMA DE LOS ANDES]. http://dspace.uniandes.edu.ec/bitstream/123456789/4691/1/PIUAMIE003-2016.pdf

Podzins, O., & Romanovs, A. (2019, April 1). Why SIEM is Irreplaceable in a Secure IT Environment? *2019 Open Conference of Electrical, Electronic and Information Sciences, EStream 2019 - Proceedings*. https://doi.org/10.1109/eStream.2019.8732173

Real Academia Española. (2021). *Diccionario panhispánico de dudas | RAE*. https://www.rae.es/dpd/asequible,

Ron Amores, R. E., & Sacoto Castillo, V. A. (2017). Las PYMES ecuatorianas: Su impacto en el empleo como contribución del PIB PYMES al PIB total. *Espacios*, *38*(53), 15.

Safarzadeh, M., Gharaee, H., & Panahi, A. H. (2019). A Novel and Comprehensive Evaluation Methodology for SIEM. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *11879 LNCS*, 476–488. https://doi.org/10.1007/978-3-030-34339-2_28

solarwinds. (2021). *Security Event Manager - Vea registros de eventos de forma remota | SolarWinds*. https://www.solarwinds.com/es/security-event-manager

splunk. (2021). *Splunk for Security | Pricing | Splunk*.

https://www.splunk.com/en_us/software/pricing/cyber-security.html

Tenelema, E. (2020). *IMPLEMENTACION DE UN MODELO DE SEGURIDAD PARA MITIGACIÓN DE VULNERABILIDADES EN AMBIENTES DE ALMACENAMIENO EN LA NUBE CON BASE EN LAS NORMAS ISO 27017 Y 24018* [ESCUELA SUPERIOR POLITECNICA DE CHIMBORAZO]. http://dspace.espoch.edu.ec/bitstream/123456789/14346/1/20T01347.pdf

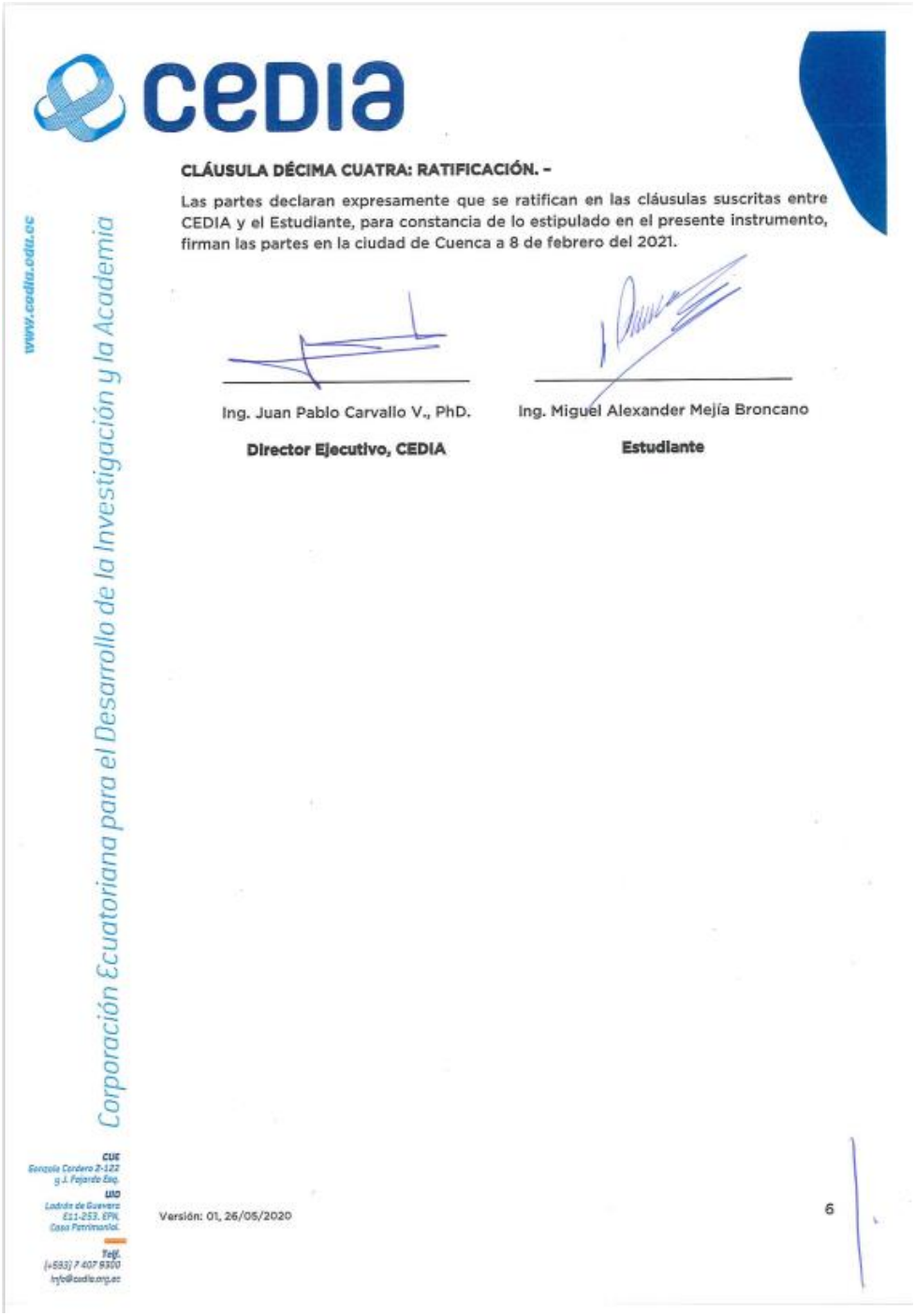
Vazão, A., Santos, L., Piedade, M. B., & Rabadão, C. (2019). SIEM open source solutions: A comparative study. *Iberian Conference on Information Systems and Technologies, CISTI*, *2019-June*. https://doi.org/10.23919/CISTI.2019.8760980

Villafuerte Quiroz, A. L., & Bravo Bravo, A. H. (2015). *Implantación De Una Herramienta Ossim Para El Monitoreo Y Gestión De La Seguridad De La Red Y Plataformas Windows Y Linux Aplicado A Empresas Medianas* [ESPOL]. http://www.dspace.espol.edu.ec/handle/123456789/29939

ANEXOS

Anexo 1: Acuerdo de confidencialidad

Ilustración 46: Acuerdo de Confidencialidad



CEDIA

CLÁUSULA DÉCIMA CUATRA: RATIFICACIÓN. -

Las partes declaran expresamente que se ratifican en las cláusulas suscritas entre CEDIA y el Estudiante, para constancia de lo estipulado en el presente instrumento, firman las partes en la ciudad de Cuenca a 8 de febrero del 2021.

Ing. Juan Pablo Carvallo V., PhD.
Director Ejecutivo, CEDIA

Ing. Miguel Alexander Mejía Broncano
Estudiante

www.cedia.edu.ec
Corporación Ecuatoriana para el Desarrollo de la Investigación y la Academia

CUE
Gonzalo Cordero Z-122 y J. Pajarito Esq.
UIO
Ladrón de Guevara E11-253. EPN. Casa Patrimonial.
Telf.
(+593) 7 407 9300
info@cedia.org.ec

Versión: 01, 26/05/2020

6

Fuente: elaboración propia

Anexo 2: Historias de Usuario

Tabla 9: Historia de Usuario 1

Nombre de la Historia: Definición de la arquitectura de la solución	
Estado: Terminado	**Responsable:** Miguel Mejía
Fecha de Inicio: 14/09/2020	**Fecha Fin:** 20/09/2020
Contexto Dado que se requiere manejar grandes volúmenes de información de manera permanente y que la misma debe ser analizada de igual manera, se requiere una arquitectura que soporte escalabilidad y brinde estabilidad a la solución. **Como** equipo de desarrollo se **necesita** analizar herramientas disponibles para definir una arquitectura candidata con las que se estime adecuadas **para** poder implantarla en la solución **Criterios de aceptación** 1. Debe incluir únicamente software libre 2. Debe alojarse en un repositorio para que pueda ser consultada a medida que sea modificado	

Fuente: elaboración propia

Tabla 10: Historia de Usuario 2

Nombre de la Historia: Preparación del ambiente de desarrollo	
Estado: Terminado	**Responsable:** Miguel Mejía
Fecha de Inicio: 21/09/2020	**Fecha Fin:** 27/09/2020
Contexto Con las tecnologías a utilizar definidas en la arquitectura, se requiere un ambiente de desarrollo funcional para todas ellas, para evitar así demoras durante el desarrollo a causa de instalaciones, incompatibilidades o cualquier otro tipo de problema **Como** equipo de desarrollo **necesito** preparar y probar todas las herramientas necesarias para el desarrollo antes de iniciar el mismo **para** evitar inconvenientes a futuro que retrasen los incrementos **Criterios de aceptación** 1. Todo el software deberá ser legal, ya sean versiones comunitarias gratuitas o versiones debidamente licenciadas 2. Las versiones más actuales disponibles en canales estables deben ser utilizadas	

Fuente: elaboración propia

Tabla 11: Historia de Usuario 3

Nombre de la Historia: Despliegue de la infraestructura para pruebas	
Estado: Terminado	**Responsable:** Miguel Mejía
Fecha de Inicio: 28/09/2020	**Fecha Fin:** 04/10/2020
Contexto. Como paso inicial y mientras se espera la aprobación del hardware en la institución, se requiere definir entornos locales de pruebas de similares características para poder avanzar con las pruebas iniciales de las herramientas **Como** equipo se **necesita** tener un ambiente de pruebas similar a producción **para** poder efectuar cambios y experimentos sin afectar la disponibilidad ni el funcionamiento de la herramienta en producción. **Criterios de aceptación** 1. El entorno se basará en docker 2. El entorno será ejecutado en una plataforma linux	

Fuente: elaboración propia

Tabla 12: Historia de Usuario 4

Nombre de la Historia: Implementación de un nodo de pruebas elasticsearch	
Estado: Terminado	**Responsable:** Miguel Mejía
Fecha de Inicio: 05/10/2020	**Fecha Fin:** 11/10/2020

Contexto. Con el fin de familiarizarse con la gestión de índices y consultas a elasticsearch, se requiere generar un nodo de pruebas con data de pruebas. **Como** equipo de desarrollo se **requiere** un entorno elasticsearch **para** probar configuraciones, familiarizarse con el funcionamiento y verificar el rendimiento con data de pruebas **Criterios de aceptación** 1. El nodo deberá ser configurado en single y no tener ninguna configuración fuera de las mínimas por defecto 2. El nodo deberá ser implementado en docker

Fuente: elaboración propia

Tabla 13: Historia de Usuario 6

Nombre de la Historia: Implementación de un colector de logs para syslog	
Estado: Terminado	**Responsable:** Miguel Mejía
Fecha de Inicio: 11/02/2021	**Fecha Fin:** 18/02/2021
Contexto. Con el servidor y las herramientas listas, se requiere habilitar un puerto al cual los servicios puedan reportar sus eventos a través del protocolo syslog **Como** administrador de la herramienta se **necesita** que los logs emitidos por el protocolo syslog puedan ser recibidos y almacenados por la herramienta **para** su posterior consulta **Criterios de aceptación** 1. Se requiere que se use el puerto syslog por defecto 514 2. La comunicación se realizará por el protocolo TCP pero se debe dejar habilitado también UDP para una futura necesidad	

Fuente: elaboración propia

Tabla 14: Historia de Usuario 7

Nombre de la Historia: Optimización de los servicios para evitar desbordamiento de recursos	
Estado: Terminado	**Responsable:** Miguel Mejía
Fecha de Inicio: 19/02/2021	**Fecha Fin:** 21/02/2021
Contexto. Debido al incidente ocurrido el día 18 de febrero, se requiere solucionar el inconveniente de agotamiento de recursos hardware en el servidor, se solicita el incremento de la memoria ram en el servidor hasta 16 gb, por cuanto este valor se debe tomar en cuenta para las configuraciones **Como** administrador de la herramienta se **necesita** que la misma sea estable y no tenga caídas **para** poder confiar en su funcionamiento **Criterios de aceptación** 1. Se tendrá como máximo 16gb de ram 2. Se requiere un up time superior al 95%	

Fuente: elaboración propia

Tabla 15: Historia de Usuario 8

Nombre de la Historia: Implementación de un intérprete para logs dns	
Estado: Terminado	**Responsable:** Miguel Mejía
Fecha de Inicio: 22/02/2021	**Fecha Fin:** 07/03/2021
Contexto. Con el servidor en funcionamiento, se requiere que los eventos recibidos sean procesado para ser almacenados de una manera interpretable **Como** administrador de la herramienta se **necesita** que los logs recibidos por el protocolo syslog, se almacenen en un formato entendible e interpretable **para** su posterior consulta **Criterios de aceptación** 1. Se requiere extraer de los eventos el tipo de registro dns (consulta o respuesta) para poder relacionar esos valores posteriormente 2. Las etiquetas de los campos deberán estar en inglés	

Fuente: elaboración propia

Tabla 16: Historia de Usuario 9

Nombre de la Historia: Implementación de un colector de logs para gelf	
Estado: Terminado	**Responsable:** Miguel Mejía
Fecha de Inicio: 05/04/2021	**Fecha Fin:** 11/04/2021
Contexto. Con la herramienta en correcto funcionamiento para el protocolo syslog, se requiere agregar los servidores honeypot al ambiente de monitoreo **Como** administrador de la herramienta se **necesita** recibir logs gelf **para** su posterior consulta **Criterios de aceptación** 1. Se podrá utilizar cualquier puerto superior a 1024 2. La comunicación se realizará por UDP	

Fuente: elaboración propia

Tabla 17: Historia de Usuario 10

Nombre de la Historia: Implementación de un intérprete para logs gelf	
Estado: Terminado	**Responsable:** Miguel Mejía
Fecha de Inicio: 12/04/2021	**Fecha Fin:** 18/04/2021
Contexto. Con un flujo estable de logs hacia el servidor se requiere procesar los mismos para almacenarlos en un formato interpretable **Como** administrador de la herramienta se **requiere** que los logs del protocolo gelf recibidos sean almacenados en un formato entendible **para** su posterior consulta **Criterios de aceptación** 1. Los eventos deberán ser persistentes sin importar su antigüedad y solo depurarse manualmente 2. Se deberán almacenar en un índice diferente al de syslog	

Fuente: elaboración propia

Tabla 18: Historia de Usuario 11

Nombre de la Historia: Definición de una línea base de funcionamiento	
Estado: Terminado	**Responsable:** Miguel Mejía
Fecha de Inicio: 19/04/2021	**Fecha Fin:** 25/04/2021
Contexto. Con todos los logs disponibles de manera continua en el servidor, se requiere poder definir el comportamiento normal de los mismos para definir una línea base de comportamiento **Como** administrador de la herramienta se **necesita** analizar el comportamiento de los servidores en base a los logs que reportados **para** su posterior análisis **Criterios de aceptación** 1. Debe ser una herramienta open source 2. El licenciamiento de la misma no deberá ser una limitante para la cantidad de información ni para su uso	

Fuente: elaboración propia

Tabla 19: Historia de Usuario 12

Nombre de la Historia: Servicio de consolidación de logs	
Estado: Terminado	**Responsable:** Miguel Mejía
Fecha de Inicio: 26/04/2021	**Fecha Fin:** 02/05/2021
Contexto. Como resultado de las pruebas de la historia de usuario 11, se detectó que consultar al origen principal de datos resulta demasiado lento a causa de la gran cantidad de información, por cuanto se requiere un proceso que analice periódicamente la información y la consolide en otro índice para reducir el volumen de elementos. **Como** administrador de la herramienta se **necesita** que sin importan la cantidad de datos existente, la herramienta sea capaz de concluir el análisis antes de que inicie el siguiente ciclo automático **para** evitar concurrencia o caídas del sistema	

Criterios de aceptación 1. El ciclo de análisis deberá permanecer en 1 minuto 2. Deberá mantenerse el *uptime* superior al 95%

Fuente: elaboración propia

Tabla 20: Historia de Usuario 13

Nombre de la Historia: Servicio de análisis de tendencias	
Estado: Terminado	**Responsable:** Miguel Mejía
Fecha de Inicio: 03/05/2021	**Fecha Fin:** 09/05/2021
Contexto. Con la herramienta de consolidación en funcionamiento, se requiere retomar la definición de la línea base de comportamiento para poder definir tendencias de los servidores **Como** administrador de la herramienta se requiere tener una tendencia del comportamiento normal de los servidores **para** poder detectar eventualidades fuera de esta **Criterios de aceptación** 1. Se deberá analizar el origen de datos consolidado 2. Los parámetros de aprendizaje deben poder ser configurables	

Fuente: elaboración propia

Tabla 21: Historia de Usuario 14

Nombre de la Historia: Servicio de detección de anormalidades	
Estado: Terminado	**Responsable:** Miguel Mejía
Fecha de Inicio: 10/05/2021	**Fecha Fin:** 16/05/2021
Contexto. Con una tendencia de funcionamiento definida, se requiere poder determinar eventualidades ocurridas fuera de la misma **Como** administrador de la herramienta se **necesita** poder detectar comportamientos anormales de los servidores basados en la tendencia de comportamiento normal **para** su correspondiente análisis **Criterios de aceptación** 1. La detección de anormalidades debe basarse en un proceso matemático 2. Los parámetros de detección deben poder ser configurables	

Fuente: elaboración propia

Tabla 22: Historia de Usuario 15

Nombre de la Historia: Servicio de correlación de orígenes	
Estado: Terminado	**Responsable:** Miguel Mejía
Fecha de Inicio: 17/05/2021	**Fecha Fin:** 23/05/2021
Contexto. En consideración a los orígenes de datos disponibles y las características de la institución, se requiere poder identificar que instituciones miembros que utilizan los servidores dns monitoreados pueden ser víctimas de ataques o infecciones, se toman como coincidentes si tienen actividad de ataques contra los honeypot también monitoreados **Como** administrador de la herramienta se **necesita** poder correlacionar los orígenes de datos en busca de posibles incidentes de seguridad **para** su correspondiente análisis y solución **Criterios de aceptación** 1. Se deberá analizar contra toda la actividad generada hacia los honeypot sin límite de tiempo 2. Se deberá identificar antes de que inicie el siguiente ciclo de análisis 3. Se considerará un posible incidente de seguridad la existencia de la ip de algún miembro que utilizó alguno de los servidores dns durante la unidad de tiempo dentro de los orígenes identificados como sospechosos por los honeypot	

Fuente: elaboración propia

Tabla 23: Historia de Usuario 16

Nombre de la Historia: Servicio de reporte de incidentes	
Estado: Terminado	**Responsable:** Miguel Mejía
Fecha de Inicio: 24/05/2021	**Fecha Fin:** 30/05/2021
Contexto. Con los incidentes identificados tanto de comportamiento como de seguridad, se requiere que los mismos sean notificados inmediatamente al personal correspondiente, por lo cual se requiere la comunicación con el servidor institucional de comunicación para poder manejar y distribuir el mismo **Como** administrador de la herramienta se **necesita** que los incidentes sean reportados vía rest al api de comunicación institucional mediante una petición post **para** su inmediata replicación al personal responsable. **Criterios de aceptación** 4. La petición post deberá ser vía https 5. El cuerpo de la petición deberá contener el detalle en formato json 6. El end point del api deberá poder ser configurable	

Fuente: elaboración propia