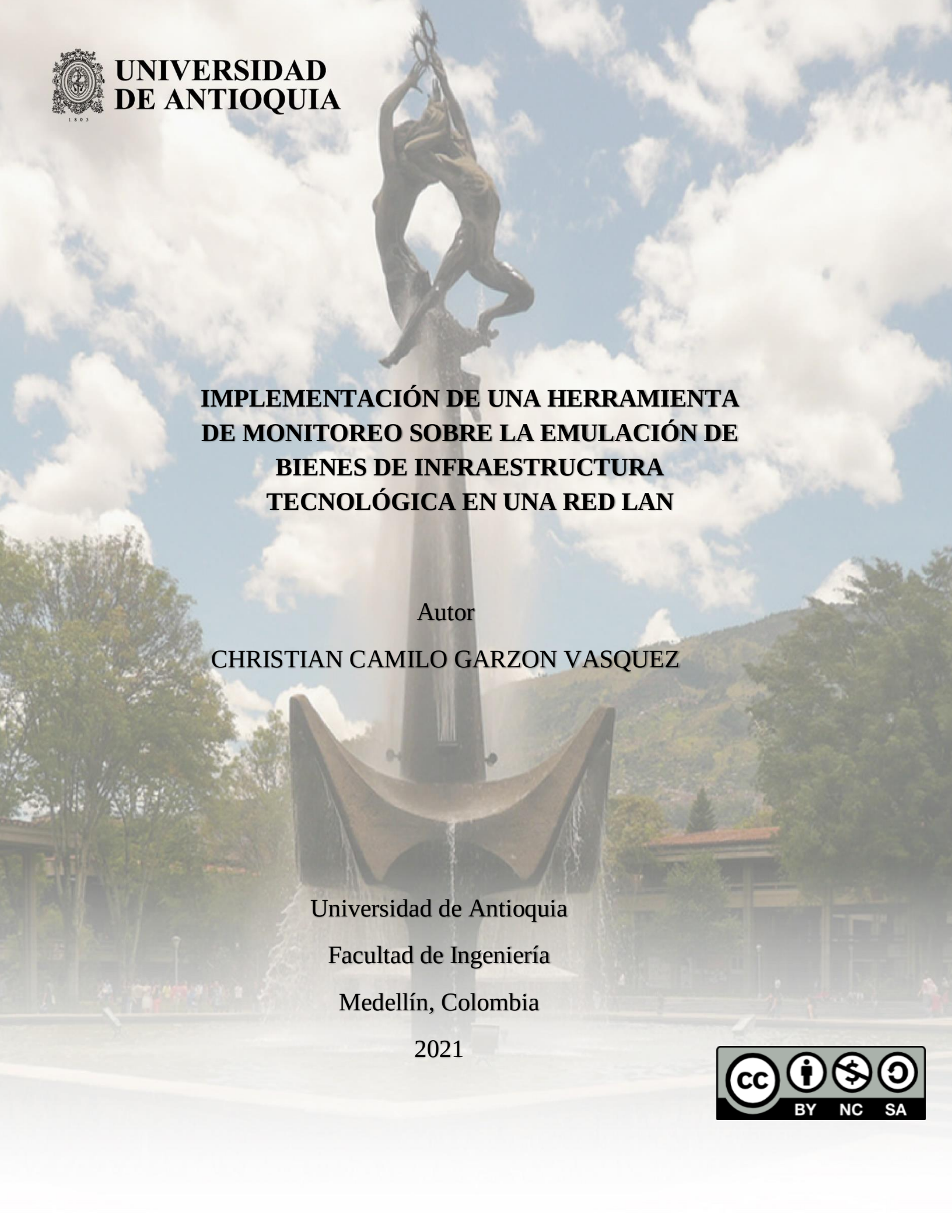


UNIVERSIDAD DE ANTIOQUIA

IMPLEMENTACIÓN DE UNA HERRAMIENTA DE MONITOREO SOBRE LA EMULACIÓN DE BIENES DE INFRAESTRUCTURA TECNOLÓGICA EN UNA RED LAN

Autor

CHRISTIAN CAMILO GARZON VASQUEZ

Universidad de Antioquia

Facultad de Ingeniería

Medellín, Colombia

2021

IMPLEMENTACIÓN DE UNA HERRAMIENTA DE MONITOREO SOBRE LA EMULACIÓN DE BIENES DE INFRAESTRUCTURA TECNOLÓGICA EN UNA RED LAN

Christian Camilo Garzón Vásquez

Informe de práctica académica como requisito para optar al título de:

Ingeniero de Telecomunicaciones

Jaime Alberto Vergara Tejada
Docente
Universidad de Antioquia

Robinson Osorio Ramírez
Analista de Telecomunicaciones
Grupo Éxito

Universidad de Antioquia
Facultad de Ingeniería
Medellín, Colombia
2021

I. Resumen.

El monitoreo se ha convertido en un trabajo de alta necesidad independiente de su enfoque, más aun si de tecnología se trata. Pero, es la herramienta utilizada y el personal que la maneja quienes determinan su rendimiento.

Como punto de partida para un trabajo de mayor alcance este proyecto pretende emular una tienda o sede de Grupo Éxito, la cual será elegida en base a análisis de incumplimientos de acuerdos de nivel de servicio y fallas eléctricas (Procesos paralelos al proyecto), para luego integrar su infraestructura tecnológica a una herramienta de monitoreo y así aprovechar las capacidades analíticas y de visualización que la misma entrega. La elección de la herramienta a usar se basó en necesidades empresariales, experiencias de herramientas en otros usos o aplicaciones, software con o sin licenciamiento, etc.

Tras la integración de la herramienta de monitoreo con la emulación definida mediante virtudes del protocolo SNMP (Simple Network Management Protocol) se recolectaron datos, los cuales pueden ser organizados mediante visualizaciones para dar un mejor análisis gráfico de la red emulada, las características de su infraestructura y su salud en general.

II. Introducción.

El constante monitoreo de las distintas herramientas tecnológicas más que una ostentación empresarial ha pasado a ser una verdadera necesidad, más en la actualidad donde los grandes volúmenes de datos apoyados de algoritmos inteligentes han logrado el desarrollo de las tecnologías involucradas en la denominada industria 4.0. Las distintas herramientas de monitoreo (de uso libre o licenciadas) son de carácter reactivo o proactivo, donde las primeras que procesan la información en el momento de ser entregada y las otras que toman acciones sin un evento que genere dicho procedimiento.

Para el área de la infraestructura tecnológica, y en especial aquellos negocios y organizaciones que dependen de esta para el manejo de sus productos, existe una cantidad considerable y variada de bienes físicos y no físicos necesarios para la entrega y el soporte de servicios de TI (Tecnologías de la información), tal es el caso de centros de datos, servidores, redes, hardware y software de computadoras y almacenamiento. El monitoreo de estos bienes de infraestructura tiene como objetivo recolectar y analizar gran cantidad de datos y almacenarlos en una base de datos (Centralizada o distribuida según software) que puede ser ordenada, consultada y analizada por humanos o máquinas con el fin de mejorar resultados y generar valor en el negocio. [1]

Dentro de Grupo Éxito se cuenta con una gran infraestructura de redes de datos en las cuales hay equipos como routers, switches, access points, balanceadores, servidores, entre otros. Y son estos bienes, junto con los no físicos, los que se encuentran en las distintas tiendas de los diferentes formatos de varias regiones del territorio colombiano, que a su vez son la base para el despliegue de diversas aplicaciones tecnológicas que día a día se encargan de generar valor a esta compañía sudamericana, que es líder en el comercio minorista masivo de bienes y servicios al consumidor final (Retail)[2]. Sin embargo, las actuales herramientas de monitoreo a nivel de redes LAN no entregan una información detallada y exhaustiva de los procesos que se ejecutan en los equipos, limitándose a dar un disminuido informe del estado de los dispositivos, es decir, si está operativa o no. Esta flaqueza disminuye la capacidad para prevenir daños parciales, interrupciones en la red o permanentes en los equipos por distintas factores,

sustituyendo un trabajo proactivo por uno reactivo que toma lugar después de un incidente.

Las herramientas de software especializados en el monitoreo de toda la infraestructura TI logran agregar datos (con ayuda de aplicaciones o equipos especializados) en formas de logs (registros) según el tráfico de la red o la actividad de los usuarios, los cuales contienen información que dependerá de la aplicación de origen tales como la hora, fecha, nombre e IP del equipo, descripciones de eventos, entre otros. De esta manera, es posible detectar posibles problemas operativos, brechas de seguridad, prevención de daños, mejoras, etc. En el área de las redes de datos y su respectiva infraestructura, es esencial un monitoreo de estos con el fin de verificar un apropiado funcionamiento y manejo de los niveles requeridos en el rendimiento de las compañías. [1]

Teniendo en cuenta lo anterior, se busca implementar una herramienta de monitoreo que permita utilizar sus principales virtudes en favor de la infraestructura de red del Grupo Éxito; la elección del software estará sujeta a estudios de pros, contras, costos y disponibilidad entre una serie de herramientas opensource o con licencia que serán ofertadas por los distintos proveedores de la compañía. Debido a la cantidad de sedes, equipos, redes y conexiones en la infraestructura de telecomunicaciones de Grupo Éxito, la implementación de una herramienta de monitoreo se llevará a cabo en un ambiente de pruebas de laboratorio, considerando los requerimientos de procesamiento que sean necesarios, emulando situaciones cercanas a la realidad, pero a una menor escala respecto a lo comentado anteriormente. La elección de sedes de prueba adecuadas se basará en otros procesos desarrollados paralelamente a la implementación del software, los cuales implicarán escogencia según afectaciones a sedes por incumplimientos de SLA del proveedor, fallas eléctricas en sedes y levantamientos de arquitecturas de red a sedes de varios tamaños y formatos.

III. Objetivos.

Objetivo General.

- Implementar una herramienta de monitoreo sobre un ambiente de pruebas que permita la recopilación de registros de los equipos de red, con el fin de generar acciones que puedan ser utilizados en estrategias de mejoramiento del servicio.

Objetivos Específicos.

- Seleccionar una herramienta de monitoreo según necesidades, licenciamiento, disponibilidad, precio y valores requeridos por la compañía.

- Definir la arquitectura de red de los sitios a emular, según reportes mensuales de afectaciones por indisponibilidad, fallas eléctricas e infraestructura existente.

- Implementar la herramienta de monitoreo en la arquitectura previamente definida.

- Evaluar, dentro de la arquitectura definida, el desempeño de la herramienta de monitoreo.

IV. Marco Teórico.

HISTORIA, UTILIDAD Y BENEFICIOS

Desde los primeros años de los sistemas de computación el software de monitoreo ha estado implícito de una forma u otra en una relación de décadas que no espera romperse. La importancia no radica en la historia de una específica herramienta, si no en el conjunto de estas donde el desarrollo y despliegue de las mismas ha aumentado su necesidad de uso. No sería muy adecuado catalogar la época de los primeros minicomputadores como la era del no monitoreo (O al menos algo cercano), pues los sistemas operativos en ese entonces y hoy incluyen servicios internos de monitoreo que podrían producir tanto basura como registros para el manejo de múltiples usuarios, memoria virtual, en fin. En el principio se contaba con sistemas que hacían monitoreo de lotes considerables, pero con poca respuesta en tiempo real para entradas y salidas que estaban bajo el cuidado de algunos técnicos operacionales que debían interpretar y responderlas sobre el plano de control para el monitoreo de funciones básicas y la salud del sistema.

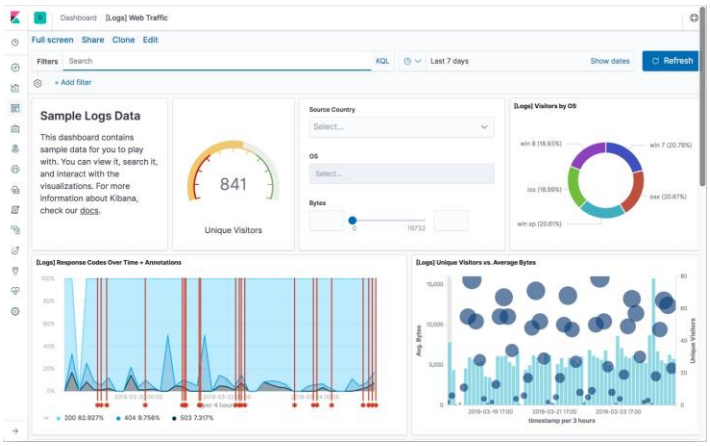


Figura 1: Dashboard Kibana del ELK Stack . Imagen tomada de https://i.pinimg.com/originals/ac/9a/27/ac9a276c522d3b94dacccf6f5cfb5a60.jpg

Con la llegada de Unix esencialmente se pasó del procesamiento por los lotes a la interacción en tiempo real, dando cabida a comandos y herramientas de monitoreo (TOP, vmstat, fuser y syslog) que a principios de los noventas serían estándar de Linux y Unix en único equipos. Posteriormente las herramientas de monitoreo gráficas se incluyeron en distintos S.O de 32 y 64 bits de Windows y

Linux/Unix como nmon, MTRG, y Big Brother. El desarrollo de herramientas de monitoreo de red también tuvieron su avance en los 80s y 90s, mientras las de S.O solo se enfocan en un solo equipo y usuario las de red tienen retos enormes, pues se enfocan en el rendimiento del hardware y software de administración de toda la red, aparte de la actividad de múltiples usuarios. Por tanto, tenemos herramientas que ayudan al desempeño del sistema y que cuentan con enfoques a un solo equipo o a una red que incorpora interfaces físicas de comunicación, hardware de servidores y recursos del sistema y que al mismo tiempo proveen registros de los datos del tráfico en la red para el aprovechamiento del sistema y usuario.

Con la llegada del siglo XXI y la expansión de internet a distintos sitios web y sus servicios, ya no bastaba el monitoreo de equipos o conjuntos de estos en una LAN, por consiguiente herramientas como Cacti, Nagios, y Zabbix se aplicaron e incluían características de soporte a protocolos de internet, multiplataforma, escalables y con interfaz web que resolvían la funcionalidad y el rendimiento en servidores, hardware de comunicación y tópicos relacionados. Actualmente con la gran acogida del e-commerce, procesamiento en nube y masificación constante de la cantidad de datos en la red, el monitoreo de datos ha dejado de ser solo amontonamiento y almacenamiento de registros para convertir en agregación de datos, filtrado, analítica, decisión, acción, etcétera. [3]

EJEMPLOS SOFTWARE DE MONITOREO

Figura 2: Logos de herramientas de monitoreo. Imagen tomada de [4] [5] [6] [7] [8]

ELK STACK: Sigla para tres proyectos open source. Elasticsearch que es un motor de búsqueda y analítica. Logstash como pipeline de procesamiento de datos del servidor que ingesta datos de diversas fuentes, los transforma y luego los envía a

Elasticsearch. Kibana es la herramienta que permite a los usuarios visualizar los datos en cuadros y gráficos con Elasticsearch. [4]

CACTI: Solución de gráficos de red diseñada para aprovechar el poder de RRDtool (Estándar de logging y sistema gráfico). Catci provee un rápido encuestador, plantillas de gráficos avanzados, múltiples métodos de adquisición de datos y mucho más. Todo lo anterior envuelto en una interfaz gráfica fácil de usar en complejas redes LAN con gran cantidad de dispositivos. [5]

NAGIOS: Software que provee monitoreo a todos los componentes de infraestructura crítica incluyendo servicios, sistemas operativos, protocolos de red, métricas del sistema e infraestructura de red. Con su servidor de registros permite encontrar rápidamente los datos, poner alertas de notificación para potenciales amenazas o simplemente consultar datos de registro para rápida auditoría de cualquier sistema. [6]

PROMETHEUS: Herramienta de monitoreo open source que integra gran cantidad de características como dimensionamiento de los datos, potenciamiento de consultas, poderosos y diversos modos de visualización, almacenamiento eficiente, operación simple, alertamiento preciso, muchas librerías de usuarios y distintas integraciones. [7]

ZABBIX: Este potente software brinda monitoreo a redes, servidores, nube, aplicaciones y servicios. Además brinda escalabilidad ilimitada, monitoreo distribuido, alta disponibilidad y fuerte seguridad. Sus soluciones son brindadas a industrias aeroespaciales, retail, energéticas, bancos, educación, entre muchas otras. [8]

V. Metodología.

1. Selección de la herramienta de monitoreo.

En el mercado existen distintas herramientas de monitoreo, las cuales fueron diseñadas para casos más específicos que otros, y aunque su fin puede ser más enfocado a la administración, tienen características útiles al monitoreo. Además, existen software de distintas índoles que tienen un límite en su escalabilidad, lo cual es interesante al momento de tratar con la infraestructura de red existente de Grupo Éxito estando distribuida a lo largo del país, en pequeñas o grandes cantidades de equipos, con alto o bajo tráfico, gran cantidad de redes LAN integradas por una red WAN, etcétera. Es de aclarar que la compañía no está enfocada cien por ciento en el sector tecnológico, pues su área de acción es la de un Retail de productos y servicios que aprovecha la tecnología para su operación, delegando su manejo a un área dentro de sí misma y a terceros.

Durante el proceso de búsqueda de una herramienta de monitoreo que se acople a las necesidades y existencias de la compañía debe tenerse en cuenta software que al hoy y en una ventana a futuro cuenten con soporte por licencia, o en su defecto, comunidades colaborativas para open source. Además de ello, es de destacar que existe software de monitoreo que no aprovecha esa gran cantidad de datos recolectados para nutrir algoritmos de aprendizaje y así poder pensar en tecnologías de cuarta generación.

Indagando distintas y populares herramientas de monitoreo se encuentra en primera instancia a Prometheus la cual es una herramienta open source bastante usada, Cacti que es recomendada para redes LAN complejas, Nagios con una considerable cantidad de características y Zabbix que también tiene monitoreo a servidores y nubes. Sin embargo, estos software (Licenciado o de uso libre) no han sido utilizados en Grupo Éxito y no hay certeza si en alguno de sus proveedores, siendo lo anterior algo crucial al no contar al momento con un soporte incluso para pruebas.

Por parte de software y herramientas utilizadas por proveedores en el tema de monitoreo es posible encontrar casos de uso con SolarWinds y Axiros, que aunque están enfocadas en la administración TI tienen importantes aplicaciones de monitoreo remoto. Sin embargo, ambas herramientas funcionan bajo licenciamiento y debido a su alta experiencia, integridad de productos, soporte y entre muchas otras radica su alto costo. Así mismo, otros terceros usan monitoreo de otros elementos de infraestructura, y de elementos lógicos, el software ELK que no requiere licenciamiento y cuenta con una gran comunidad colaborativa y varios releases de versiones al año

SOLARWINDS: Sólido software de monitoreo de redes la cual permite detectar, diagnosticar y resolver rápidamente problemas de conectividad y rendimiento en la red. En sus características radica el monitoreo de fallas, desempeño y disponibilidad evitando la inactividad de la red. Además es posible realizar alertas inteligentes con posibilidad de personalización reconociendo topologías y dependencias, incluso descubrimiento y monitoreo por protocolo SNMP.

Figura 3: Logo de SolarWinds. Imagen tomada de [9]

SolarWinds permite realizar diagnósticos mediante sus herramientas, las cuales monitorean constantemente dispositivos de red posibilitando realizar un troubleshooting (Solución de incidentes) de redes cuando llega el problema, logrando alertar basado en el rendimiento.

Según la siguiente figura SolarWinds brinda información de ponderados del tiempo de respuesta, pérdida de paquetes, uso de CPU, memoria actual, velocidad y latencia. Además está herramienta permite realizar un troubleshooting analizando paquetes y posteriormente investigar si la causa raíz es en la aplicación o en la red, incluso priorizando aquellos tipos de redes que son más críticos. [9]

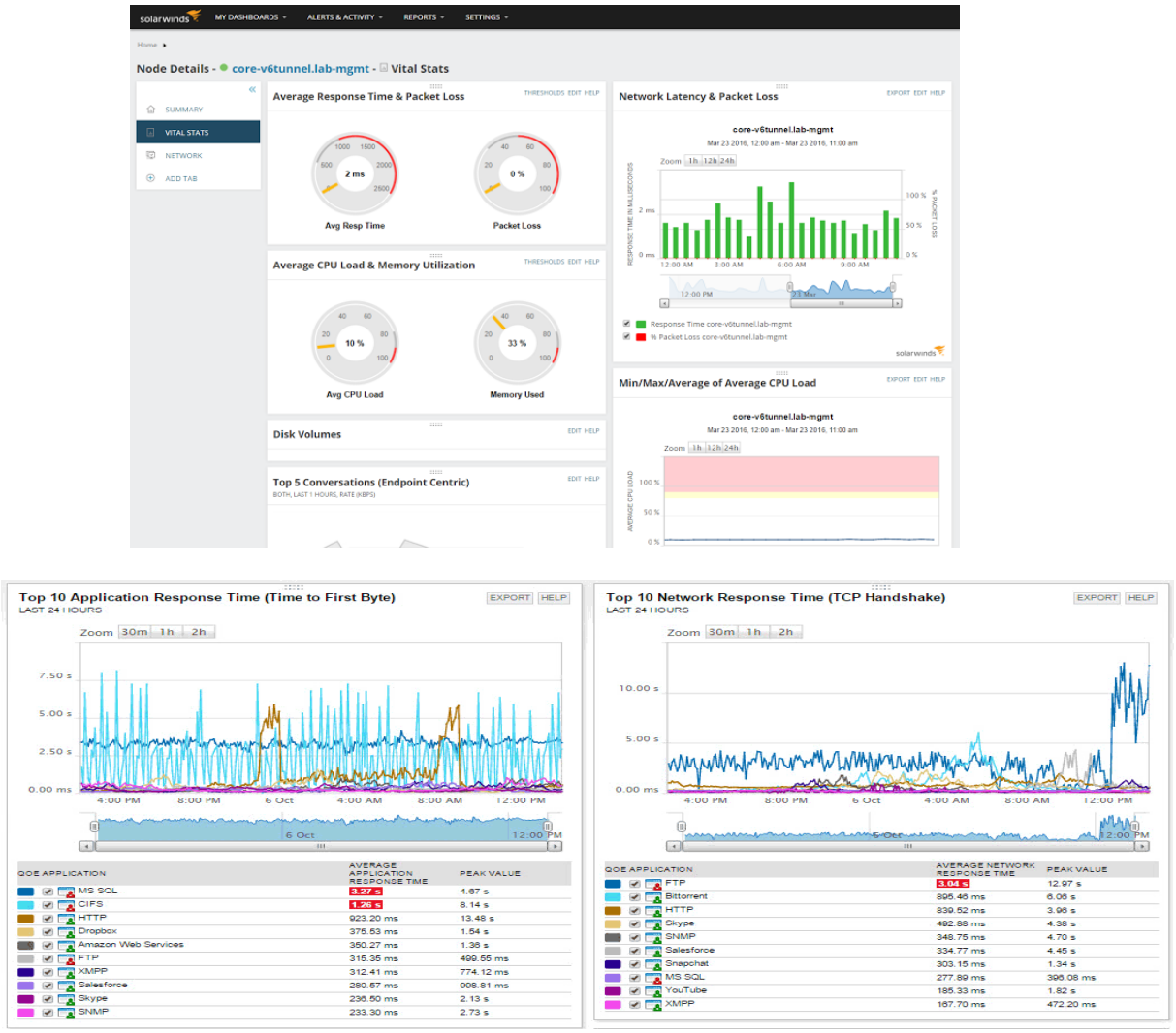


Figura 4: Dashboards SolarWinds. Imagen tomada de [9]

AXIROS: Con una línea AXess enfocada en la gestión de dispositivos, una línea Axact que incorpora conectividad para IoT y finalmente una línea AXtract basada en monitoreo de dispositivos.

axiros
Lasting Advantage

Figura 5: Logo Axiross. Imagen tomada de [10]

La línea de monitoreo Axtract involucra completamente el software y no requiere de complementos adicionales. Posee la capacidad de exportar datos en diferentes formatos y también la importante característica de alertar para un soporte y alertamiento proactivo, esta herramienta posee la capacidad de obtener información desde diferentes protocolos como https, ssh, snmp, etc.

Figura 6: Dashboards Axiros Axtract. Imagen tomada de [10]

Axtract posee la habilidad de agregación de datos en crudo, mejorando el rendimiento de la captura de información, permite lograr gran escalabilidad para redes complejas, prioridad en la recolección de datos, entre varias características.

ELK: Esta herramienta Open Source basada en la combinación de tres proyectos funciona inicialmente con un motor de búsqueda basado en JSON (JavaScript Object Notation). El motor es nutrido tras una importante ingesta de datos vía pipeline, que es una serie de procesos y comandos conectados donde la salida de uno es la entrada de otro. Luego permite visualizar de distintas maneras estos datos a través de la última herramienta. Debe tenerse en cuenta el orden de los proyectos Elasticsearch, Logstash y Kibana.

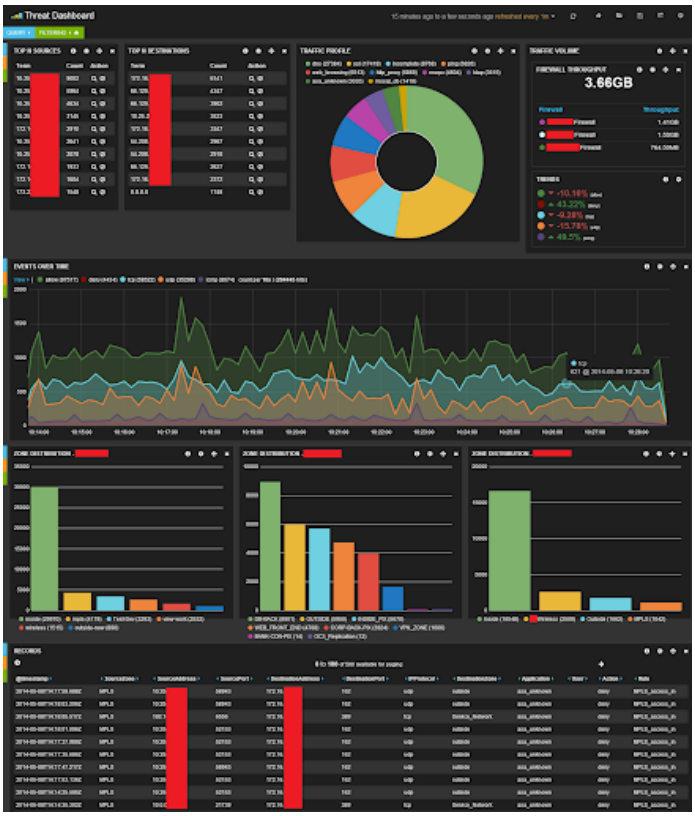


Figura 7: Dashboards ELK. Imagen tomada de [11]

ELK permite hacer marcación de logs a través de Kibana, además de obtenerlos en tiempo real y poder estructurar los elementos como con su IP o el tipo de evento. También es posible buscar y visualizar inmediatamente tendencias en los datos y no tener que buscarlos y compararlos manualmente. Logstash permite el procesamiento de la data mediante pipelines antes de que sean ingestados en Elasticsearch, permitiendo a su vez búsquedas fáciles y centralizadas. Incluso, de manera a destacar, ELK tiene la capacidad de moldear automáticamente el comportamiento de los datos que provienen de Elasticsearch y alertar de problemas en tiempo real.

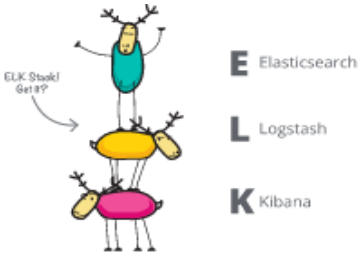


Figura 8: Proyectos ELK. Imagen tomada de [4]

El monitoreo bajo ELK se presenta en base a diferentes logs obtenidos con la cadena de herramientas, pudiendo ser capturados desde Kubernetes, MySQL, Apache, Windows, SNMP, etc.

ELK maneja un plugin para el protocolo SNMP, que en general obtiene información relacionada al estado actual de los dispositivos de operación mediante sus versiones v1, v2c o la más segura v3 y los protocolos de capa de transporte TCP o UDP. Su último release v1.2.7 fue lanzado el día 10-05-2020. [12]

Enfocándose en el alcance propuesto para este proyecto y la implicación en gastos por un producto licenciado, tomando en cuenta el tiempo de gracia de una licencia de prueba de 30 días, s*e optará por el uso del software de monitoreo ELK* ya que sus características y reciente plugin de aplicación con SNMP se adapta a las necesidades que la empresa y el proyecto necesitan.

Es posible integrar ELK junto con el software de emulación GNS3 que es comúnmente utilizado como plataforma de pruebas y de estudio. Sin embargo, el funcionamiento del mismo fue encontrado bajo la aplicación de contenedores en Docker siendo algo fuera del alcance propuesto para este proyecto pero con precedente de uso. [13]

Tratándose de equipos de infraestructura de diversos usos, diversos fabricantes, diversas referencias, diversas versiones, entre otros, se debe de buscar algo en común con lo que puedan contar. El protocolo SNMP (Simple Network Management Protocol) ubicado en la capa de aplicación y estandarizado por la IAB en RFC1157 es quien permite el intercambio de información administrativa entre dispositivos de red. [14]

Sin embargo, para el caso de este proyecto se cuenta con algunos equipos de infraestructura de red con un release (versión) que no soporta algunas características de SNMP, es por ello que no es posible la inclusión del equipo como agente colector y emisor de datos a un administrador, además de ser

una tarea tediosa y que afecta la sincronización de la totalidad de infraestructura existente.

Por el contrario, se da pie a otra forma de trabajo de SNMP la cual permite que sea el administrador quien envíe las peticiones a los equipos, dado un determinado tiempo y una lista de características a recolectar. Por tanto, se tendría un trabajo centralizado tanto para el envío como para la recolección de los datos que luego podrán ser visualizados de una forma organizada y brindando mayor valor.

2. **Definición de arquitectura según reportes y existencias.**
   - **Automatización del proceso de conciliación de incidentes que afectan SLA.** Como acción inicial para la escogencia de la sede o tienda base a emular se optó por mejorar un proceso existente en la compañía de manera automatizada. Día a día se tiene un reporte de aquellas sedes que han sufrido afectaciones físicas o lógicas en su infraestructura de red, con base en esto tienen una afectación medida en tiempo sin operación de su enlace principal, de respaldo o ambos, destacando el último ya que la sede estaría aislada de la red principal. Por consiguiente, un porcentaje de este tiempo caído es asumido por la empresa proveedora y según su causa raíz se puede aumentar o disminuir según criterio del cliente y acuerdos previos. En consecuencia, la duración temporal total fuera de línea asumida de la sede comienza a sumar y a brindar un porcentaje de tiempo donde los enlaces fueron afectados respecto al total del mes.

     La indisponibilidad de la sede en el mes está sujeta a un acuerdo de nivel de servicio SLA (por sus siglas en inglés), que se pacta con el proveedor dependiendo de características de la sede como su topología, tamaño, importancia, entre otros. En contraste, debe de tenerse en cuenta que no toda causa raíz del problema es susceptible a un porcentaje de tiempo asumido por el proveedor, pues hay factores externos como pérdidas de fluido o daños a la infraestructura de conectividad por vandalismo que no son sujetos de penas. Pero existen otras causas a nivel técnico y físico de

dispositivos o medios de transmisión que sí son razones de porcentajes asumidos como fallas en protocolos, pérdidas de paquetes, desconfiguración de equipos y muchas más.

Figura 9: Proceso de conciliación entre Grupo Éxito y empresa proveedora. Imagen tomada de [15] [16] [17]

Dichos porcentajes de indisponibilidad generan un descuento en los servicios contratados dependiendo esencialmente de si el porcentaje de disponibilidad está por debajo del SLA y del valor contratado mensualmente por el servicio en la sede. Por otro lado, dependiendo del tipo de SLA fijado entre las partes y la cantidad de sedes que trabajan bajo ese porcentaje, los distintos incumplimientos de sedes por SLA generarán una indisponibilidad general de la red en el mes y es allí donde radica la importancia de este proceso de conciliación entre partes, ya que bajo la gestión de proveedores se mide el rendimiento del servicio ofertado durante el mes. Sin embargo, los incidentes pueden tener como responsable a la empresa proveedora o la misma compañía que contrata el servicio. Y es con lo anterior donde se debe de llevar un proceso similar para verificar individualmente aquellas sedes que más tiempo estuvieron fuera y las veces que ocurrió durante el mes, pero para el caso de la compañía.

Para el caso de la automatización el lenguaje de programación escogido fue Python. La razón de su escogencia radica en los conocimientos previos, su fuerte comunidad, las distintas librerías y su eficiencia de procesamiento. El método de implementación fue la programación funcional debido a la cantidad de procesos que debían ser repetidos una y otras vez, teniendo en cuenta las bases de datos de las sedes, sus respectivos enlaces y características. Como pruebas de validación se realizaron simultáneamente procesos de conciliaciones con el fin de comparar el método anterior

basado en macros de Excel y el nuevo basado en Python. Los resultados dados indican una reducción de tiempos promedios de 20 a 4 horas.

En el proceso de automatización se cuenta con una entrada y una salida, siendo el ingresante un archivo CSV (comma-separated values) con información de la fecha, cada uno de los incidentes reportados del día y sus parámetros correspondientes, para el caso de la salida es nuevamente un archivo de Excel con todo el proceso realizado. El archivo cuenta con una hoja de los incidentes caracterizados, otras dos con uno y más incidentes caracterizados atribuidos al proveedor respectivamente, las siguientes dos iguales al caso anterior pero atribuido a la compañía y por último una hoja con la disponibilidad de la infraestructura de red del mes.

En el proyecto de este informe, este proceso ayudará con la escogencia de la sede base para la red a emular, ya que se encuentran distintas cuestiones de caída de enlaces donde sobresalen las fallas eléctricas.

Enlace	Topología	SLA	Fecha Caída	CAIDA	SUBIDA	% Conciliado	Responsable	
[redacted]	Mpls+Internet	99,8	2021-03-24	05:27:00	12:50:00	0,25	PROVEEDOR	

Nombre Sede	Enlace	SLA	Disponibilidad	Cumple	Valor Sede	Descuento	Valor A Pagar	Reportes
ÉXITO MACONDO	[redacted]	99,9	99,68784722	NO CUMPLE	$ 100.000,00	$ 25.000,00	$ 75.000,00	2

Nombre Sede	Enlace	SLA	Disponibilidad	Reportes
Carulla Pompeya	[redacted]	99,8	97,71329365	1

ITEM	DISPONIBILIDAD MES	SEDES QUE CUMPLEN	SEDES QUE INCUMPLEN
1	100,000%	2	0
2	99,997%	31	1
3	99,994%	5	1
4	100,000%	17	0
Promedio	99,998%	57	2
Total Descuento	$ 80.000,00		

Figura 10: Incidente, incidentes proveedor, incidentes compañía y disponibilidad. Recurso propio

- **Informe anual de fallas eléctricas.**
  Mensualmente se recibe un informe de las fallas eléctricas y estas incluyen información de la sede, la ciudad, tipo de falla y fecha en la que se desarrolló el incidente. Un informe anual se diseña con el fin de tipificar las afectaciones por sedes, meses, trimestres, ciudades y formatos. Para el caso del proyecto en cuestión se tomaron las sedes que más afectaciones han

tenido durante el año y se comparó con la actividad de automatización anterior.

SEDE	CIUDAD	PRIORIDAD	ID SERVICIO	CAUSA (I)	FECHA	Mes
ÉXITO MACONDO	Macondo	Prioridad 3	[redacted]	Instalaciones - Energía	30/10/2020	Octubre

Figura 11: Ejemplo de reporte de falla eléctrica. Recurso propio

Con los procesos de conciliación y fallas eléctricas se puede proceder a la fase de levantamiento de arquitectura y posteriormente de emulación.

- **Arquitecturas de red.**

  Una arquitectura de red es un plano de conexiones físicas y lógicas, indicando las tecnologías que admiten la infraestructura física y a los servicios y protocolos que van a trasladar los mensajes en toda esa infraestructura [18]. Con estos planos es posible mejorar el troubleshooting y la percepción de la red por parte de analistas propios y de proveedores. Dentro de Grupo Éxito existen un total de unas 550 sedes de distintos formatos a las cuales a la fecha les han sido levantadas sus infraestructuras.

  Para la representación de la arquitectura se usa la aplicación web https://app.diagrams.net/ a través de su galería de imágenes Network, por este medio es posible representar switches, routers, access points, nombrar interfaces, brindar información de IPs, cuadros para separaciones, entre muchas otras.

  La sede a emular, escogida según procesos y análisis de los anteriores, presenta la siguiente arquitectura:

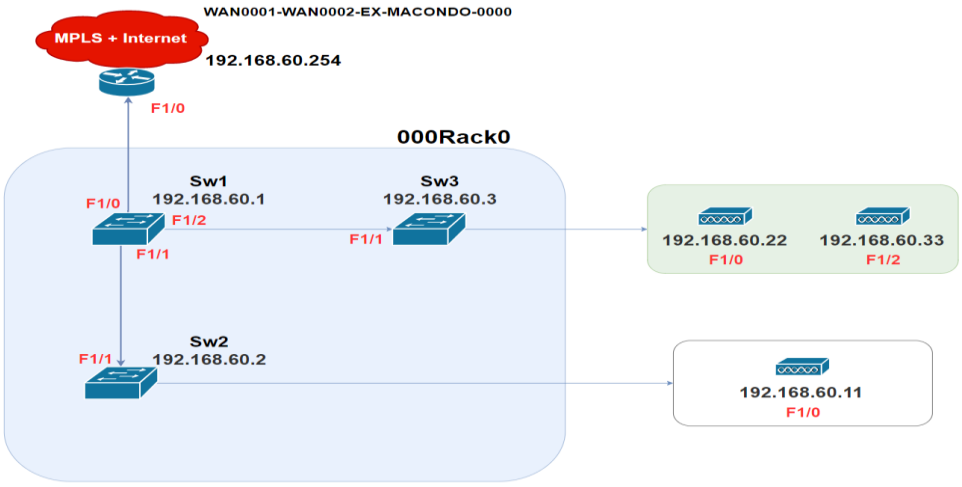


Figura 12: Arquitectura de red de la sede a emular. Recurso propio

3. Implementación de la herramienta de monitoreo.

- **Emulación de la arquitectura de red en GNS3.**

  En esta sección del proyecto fue necesario instalar el software de emulación GNS3 el cual permite implementar en el equipo de cómputo una topología de algunos elementos tecnológicos entre switches, routers y máquinas virtuales principalmente, igualmente permite emular varios dispositivos alojados en múltiples servidores o incluso en la nube. El fin del uso de GNS3 para este proyecto es virtualizar hardware costoso y poder trabajar con imágenes liberadas por Cisco para su uso. [19]

Figura 13: Logo GNS3. Recurso propio Imagen tomada de [19]

Según la arquitectura presentada anteriormente, dentro de la emulación de red de la sede debe de incluirse un router, tres switches y tres access points. Pero, debido a las restricciones de IOS dados por Cisco que involucran

licencia paga por el uso de releases de switches y access points se usará solo el de Router, teniendo en cuenta que el mismo puede emularse como un switch con opciones propias de GNS3. Durante la búsqueda de imágenes liberadas de switches se logró encontrar algunas que eran de uso experimental, con la salvedad del fabricante de no ser usadas comercialmente, desafortunadamente estas no podrían guardar configuraciones hechas a los dispositivos al iniciarse. La idea general es utilizar software y IOS libres en este proyecto para no incurrir en propiedad tecnológica licenciada.

La imagen del router a utilizar fue de la referencia c3640 de Cisco donde en cada uno de sus cuatro slots se utilizó un adaptador NM-1FE-TX que indica un puerto FastEthernet para transmisión y recepción. En el caso de switches GNS3 da la opción de EtherSwitch Router, lo cual indica que el router trabajará en modo switch y en su slot será insertado un NM-16ESW para un total de 16 puertos FastEthernet. Lastimosamente para el caso de los APs no se pudo encontrar una forma de emulación y estos fueron reemplazados por VPCs que son dispositivos finales con conectividad de red básica y, para el caso, son suficientes para evaluar conectividad.

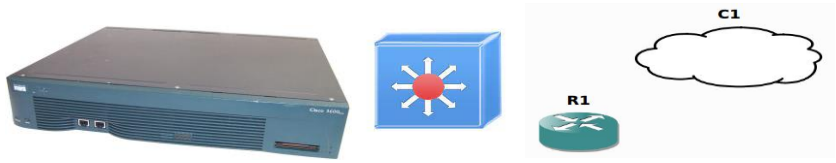


Figura 14: Router GNS3 y representación como EtherSwitch y Router en GNS3, representación de nube en GNS3. Tomado de [20][21]

Finalmente, la red emulada debe de estar conectada hacia la red del servidor donde estará ELK, teniendo conectividad extremo a extremo. GNS3 contiene la opción de Cloud la cual emula una red que conecta otras redes y fue de gran utilidad puesto que en cierta forma se pretende acercarse a lo real con un servidor y una sede en distintas redes.

Para el caso de la red exterior al router se tiene la dirección de red 192.168.68.0/24, por tanto al router en su interfaz f0/0 toma los parámetros de red (Incluida dirección IP) vía servidor DHCP y gracias a este indica su

gateway (puerta de enlace) que internamente irá a la nube para ser direccionado a las redes conectados a ella. El equipo que almacena el proceso contiene una interfaz virtual, con dirección IP 192.168.68.1, conectada a dicha nube y a su vez con alcance de conexión a la red 192.168.1.0/24 donde está la interfaz física del equipo y a su vez el servidor ELK. Y la sede emulada tendrá como dirección de red 192.168.60.0/24. Debe tenerse en cuenta lo mencionado anteriormente con los APs. Ver figura 15.

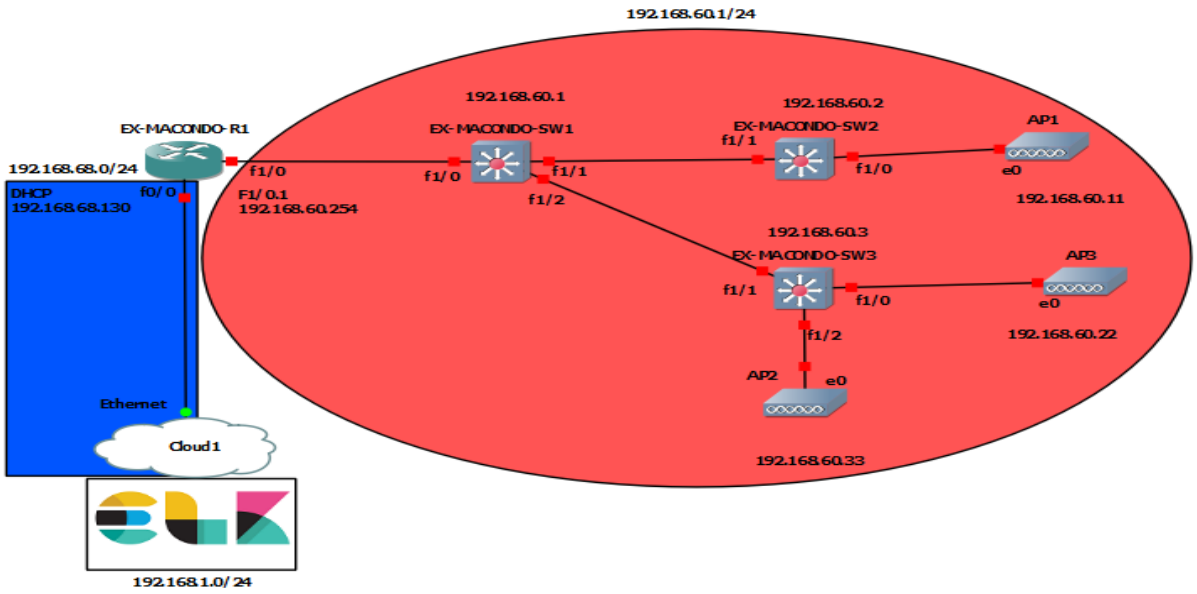


Figura 15: Representación sede emulado en GNS3. Recurso propio

En la configuración general de todos los equipos es necesario habilitar el siguiente código en el CLI (Command Line Interface) de los equipos snmp-server community Private3 RO con esto se habilita el protocolo SNMP con la comunidad (Parámetro de la versión v2c de SNMP) Privat3 en modo lectura (Read Only). En el Router se abordaron temas de brindar descripciones y direcciones IP a las dos interfaces, incluyendo la interfaz f1/0 que hace parte de la red de la tienda 192.168.60.0/24 y la f0/0 dada por DHCP y a su vez brindando la dirección IP de gateway llevando a dicha dirección de la nube el tráfico que desconozca su ruta.

Respecto a los Switches, aparte de la configuración SNMP se debe configurar en primer lugar una Vlan común para todos los switches y APs con una dirección de red 192.168.60.0/24, esta Vlan de administración será común para los dispositivos de la sede (como se ve en la figura 15 También debe de verificarse cuales interfaces serán troncales y cuáles serán de acceso dando configuración y descripción en los SW, para tal caso conexiones entre switches serán troncales y entre SW-R igual, mudando a acceso si es conexión entre SW y AP. Por último debe brindarse el gateway de esta red, siendo la interfaz f1/0 del router con dirección 192.168.60.254. Para este punto ya es posible tener conexión entre dispositivos de la red de la tienda.

En último lugar se debe de brindar conectividad entre la estación final donde estará alojado el servidor de ELK y la red de la sede, para ello se indicará cómo llegar desde la tabla de enrutamiento del sistema operativo del servidor. Para el caso de este proyecto la tabla de enrutamiento conocía la red 192.168.68.0/24 y se indicó que para alcanzar la red 192.168.60.0/24 debía llegar al router de la sede a través de la interfaz 192.168.68.130. Para este punto ya es posible tener conexión entre dispositivos de la red de la tienda y la dirección IP del servidor.

IP	DISPOSITIVO
192.168.68.130	EX-MACONDO-R1
192.168.60.1	EX-MACONDO-SW1
192.168.60.2	EX-MACONDO-SW2
192.168.60.3	EX-MACONDO-SW3

Tabla 1: Dispositivos a emular

- **Integración GNS3 - ELK**

  La instalación de ELK depende del sistema operativo donde será instalado, pero en la mayoría de los casos se basa en descargar desde consola o gráficamente y luego mediante CMD o Consola ejecutar el .bat o systemctl

start/enable respectivamente. Se sugiere buscar información de instalación y ejecución según el sistema operativo.

Para el proceso de integración y de recolección de registros es importante en primer lugar lanzar Elasticsearch y Kibana, recordando que ambos están enlazados y a disposición de los datos llevados a través de Logstash. Sin embargo, en primer lugar será necesario validar los MIBs (Management Information Base) a recolectar, recordando que este acrónimo se refiere a la base de información gestionada y organizada jerárquicamente permitiendo orden en el dispositivo de red. Cada variable de la MIB posee una ID de objeto (OID) y de esta forma identifican exclusivamente los objetos administrados por la MIB y se presentan en forma de árbol. [22]

En general los OID a usar llevaran igual inicio iso(1), org(3), dod(6), internet(1), mgm(2) o private(4) y mib-2(1) siendo así 1.3.6.1.2.1 o 1.3.6.1.4.1 y de allí en adelante. Debe tenerse en cuenta que, en general, los OID varían según el fabricante y el modelo del dispositivo y es por ello que un mismo OID puede no ser igual incluso en una misma marca.

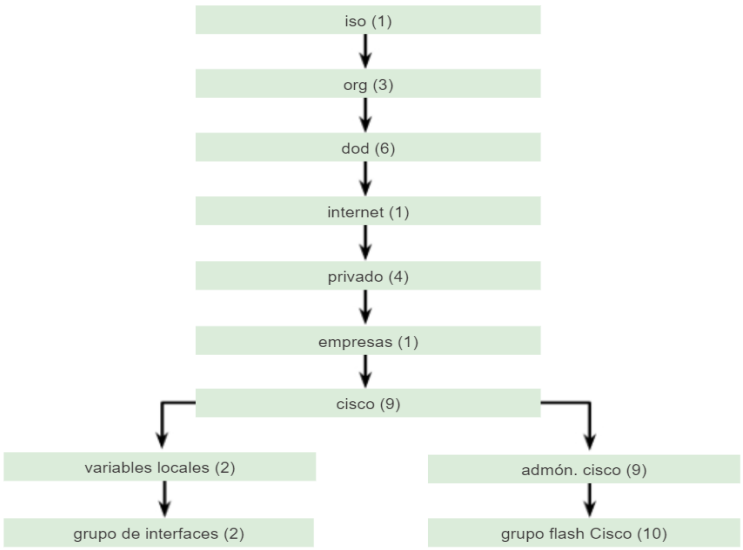


Figura 16: Estructura MIB CISCO. Tomado de [22]

Para el caso de los OID que se utilizarán como insumo de monitoreo se tiene:

OID	DESCRIPCIÓN	EJEMPLO
1.3.6.1.2.1.2.2.1.2	Descripción de la interfaz	FastEthernet0/0
1.3.6.1.2.1.31.1.1.1.18	Alías de la interfaz	EXMACONDO R0S3-F1-0 Access
1.3.6.1.2.1.2.2.1.8	Operatividad de la interfaz. 1 conectado y 2 no conectado.	1 ó 2
1.3.6.1.2.1.2.2.1.5	Velocidad el puerto	100,000,000
1.3.6.1.2.1.2.2.1.10	Octetos entrantes	265
1.3.6.1.2.1.2.2.1.16	Octetos salientes	110

Tabla 2: OID por cada interfaz del dispositivo

OID	DESCRIPCIÓN	EJEMPLO - TIPO
1.3.6.1.4.1.9.2.1.3	Hostname (Nombre del dispositivo)	EX-MACONDO-SW3
1.3.6.1.4.1.9.9.109.1.1.1.1.5	Porcentaje uso CPU	10
1.3.6.1.4.1.9.9.13.1.3.1.6	Estatus de temperatura. 1 normal, 2 advertencia, 3 crítica, 4 apagado, 5 no presente y 6 no funcional. [23]	1

Tabla 3: OID por dispositivo

Debe de tenerse en cuenta que al ser dispositivos emulados no tendrán acceso a artefactos que tienen los dispositivos reales, tal es el caso del medidor de temperatura y por ello se optó utilizar un valor que de estatus actual del equipo.

Ahora, se procederá a crear el archivo de Logstash que mediante el plugin (complemento) de SNMP ingresará a los dispositivos a buscar la información de las tablas anteriores. El plugin de Logstash incluye la importación de IETF MIBs mediante la llamada de los OID correspondientes y su archivo de configuración se basa en la típica estructura de Logstash *input{}, filter{} y output{}*, teniendo en cuenta que *snmp{}* irá dentro de *input{}*. [12]

Para el ingreso de los datos a buscar en *input{}* que a su vez incorpora el plugin de *snmp{}* primero se debe de especificar:

- **Host:** Especifica la lista de host a los cuales se les consultará los OID, teniendo en cuenta sus parámetros:
  - **Host:** Cada host debe de estar en el formato {tcp|udp}:{ip address}/{port}, por ejemplo *host => udp:192.168.1.2/161* que se refiere a la IP del agente por el puerto 161/162 de SNMP.
  - **Community:** Indica el nombre de la comunidad configurada en los dispositivos de red a capturar, por ejemplo *community => "public"*.
  - **Version:** Este parámetro se refiere a la versión de SNMP a usar, para tal caso la 1, v2c o la más segura 3. Ejemplo *version => "2c"*.
  - **Retries:** Número de intentos en caso de falla, por defecto serán dos y se llama así *retries => "2"*.
  - **Timeout:** Es el tiempo para acabar el intento de conexión, por defecto es 1000 milisegundos. *timeout => "1000"*.

- **Get:** Toma el OID en cuestión. Ejemplo: *get => ["1.3.6.1.2.1.1.1.0"]*.

- **Walk:** Toma todos los OID herederos desde OID en cuestión. Ejemplo: *walk => ["1.3.6.1.2.1.1"]*.

- **Table:** Permite nombrar tablas y obtener los OID que se le requieran en sus distintas columnas, es ideal cuando se trata de datos de diferentes dimensiones. Ejemplo: *tables => [{"name" => "interfaces" "columns" => ["1.3.6.1.2.1.2.2.1.1", "1.3.6.1.2.1.2.2.1.2", "1.3.6.1.2.1.2.2.1.5"]}, {"name" => "ltmPoolStatTable" "columns" => ["1.3.6.1.4.1.3375.2.2.5.2.3.1.1", "1.3.6.1.4.1.3375.2.2.5.2.3.1.6"]}]* [12]

Para el campo *filter{}*, que es la parte del Logstash.conf que editará la información de los OID, incluirá el filtro *split{}* que separará todos los campos obtenidos en un OID de múltiples valores para dejarlos en un solo valor y registro. Para este caso los OID a aplicar el filtro son los de la tabla XXX de las interfaces. *field => "interfaces"*.

El campo de salida output{} incluirá dos partes:

- **Codec:** Es el encargado de codificar la información que llega del *filter{}* en este caso mediante el *rubydebug* que es enfocado a objetos y de esta manera entregar algo legible al Elasticsearch y posteriormente al Kibana.

- **Elasticsearch:** Luego de usar el Codec se indicará de que manera llegará la información al Elasticsearch.
  - **Action:** Define que se debe hacer con la data, en este caso será indexada. *action => "index"*.
  - **Host:** A donde serán enviados los registros y por cual puerto. *hosts => ["127.0.0.1:9200"]* para tal caso al locahost y por el puerto 9200 de Elasticsearch.
  - **Index:** Nombre clave con el cual serán individualizados los registros. *index => "snmpv2c"*.

La estructura general para el logstash.conf es la siguiente:

```
input {
  snmp {
hosts => [{host => "udp:192.168.68.130/161" community => "Privat3" version => "2c"  retries => 2  timeout => 1000}, ... ]

tables  => [ {"name" => "interfaces" "columns" => ["1.3.6.1.2.1.2.2.1.2", ... ]}, {"name" => "devices" "columns" => ["1.3.6.1.4.1.9.2.1.3", " ... ]} ]
  }
}
filter{
  split{
    field => "interfaces"
  }
}
output {
    stdout{
        codec => rubydebug
    }
    elasticsearch {
        action => "index"
        hosts => ["127.0.0.1:9200"]
        index => "snmpv2c"
    }
}
```

4. Evaluación y desempeño de la herramienta de monitoreo.

- **Exploración de la información recolectada en Kibana.**

  Lanzado el archivo de configuración de logstash con el comando "logstash -f losgtash.conf" en el mismo directorio donde fue guardado, luego

configurado y creado en Kibana el Index Pattern con el mismo nombre dado en el logstash.conf (con opción de timestamp) se podrán ver los registros capturados mediante el menú Analytics - Discover de Kibana. Los distintos logs que han llegado en los últimos 15 minutos, un año o ventana de tiempo personalizada se verán así:

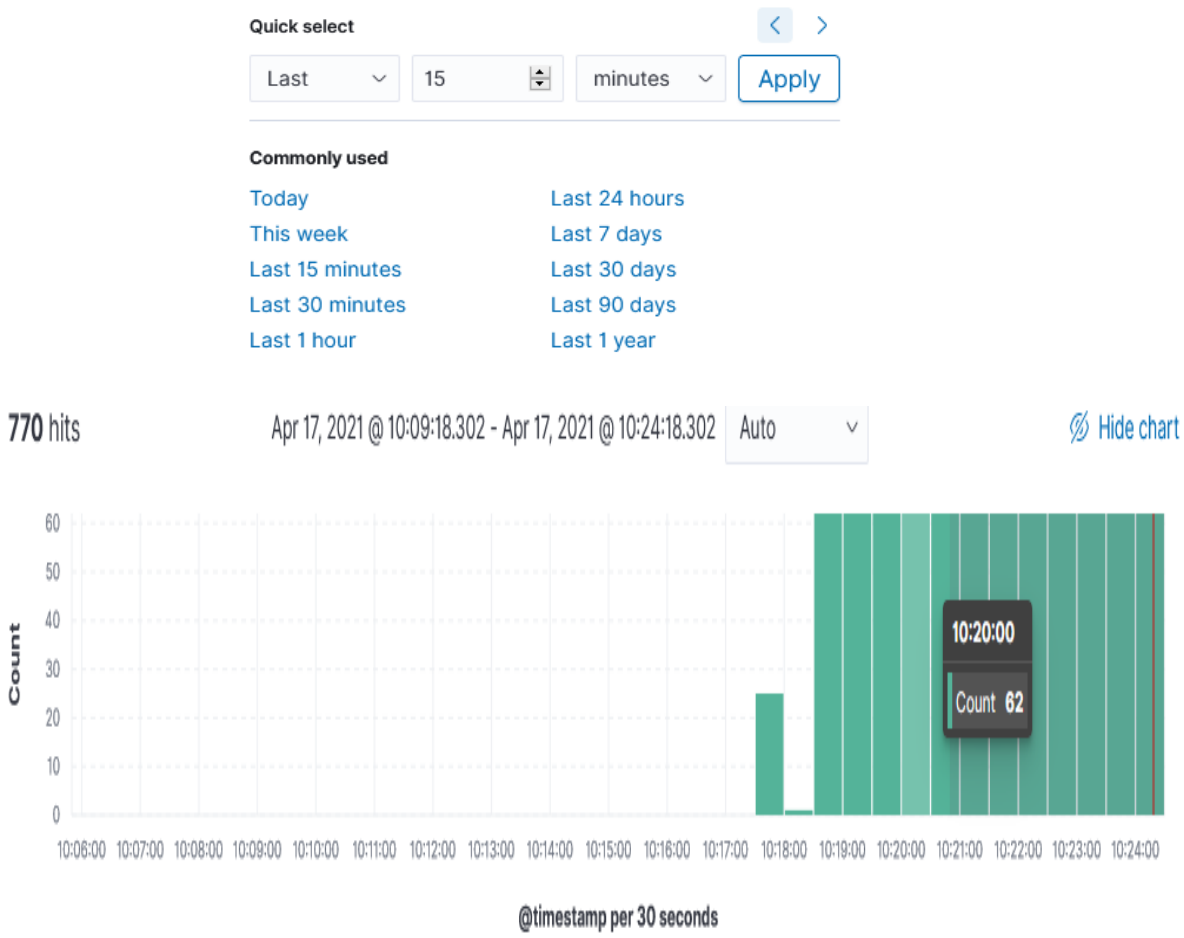


Figura 16: Logs generados y ventanas de tiempo. Recurso propio

Los logs individuales pueden verse en formato de tabla o formato JSON. Dentro de cada log puede verse como casos relevantes la interfaz en cuestión, su descripción, si está conectada o desconectada, bps en que trabaja el puerto, octetos que entran y que salen de la interfaz, en qué dispositivo está, uso de la de CPU, estado de la temperatura y su dirección IP.

Figura 17: Información relevante de un log en formato table. Recurso propio

- **Creación de visualizaciones y resultados.**
  Kibana dentro de su catálogo de herramientas incluye librerías de visualizaciones basadas en barras horizontales y verticales, indicadores de metas, mapas de calor, líneas, pasteles, tablas y áreas. Además incluye mapas, visualizaciones creadas desde código, filtros de controles, análisis en series de tiempo, etc.

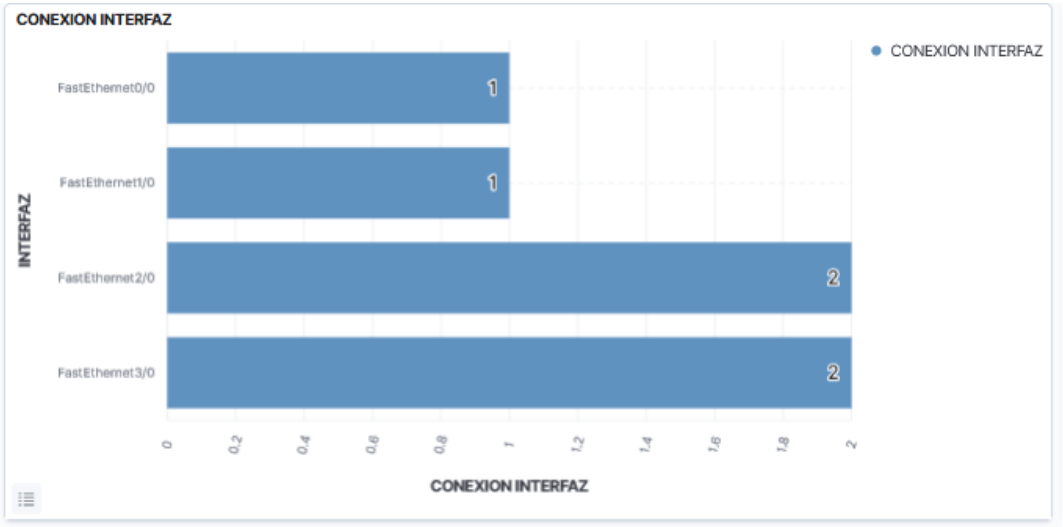
  Para el caso en cuestión, es suficiente con el uso de gráficas de barras horizontales y tablas, puesto que de tal manera pueden ser presentadas las métricas de cada dispositivo y de sus interfaces. Debe de aclararse que estas visualizaciones se deben presentar en conjunto para brindar mejores métricas y análisis, con el agregado de que Kibana permite realizar filtrados interactivos en las visualizaciones. Es importante tener en cuenta que estas visualizaciones trabajan con distintas métricas de los valores que vienen desde SNMP, tal es el caso de mínimos, máximos, promedios, cuentas, muchos más pero en especial el último valor conocido (Top Hit o Last Value) que indicarán en tiempo real el estado de un dispositivo o su interfaz.

  El conjunto de visualizaciones dan como resultado un Dashboard (Tablero), el cual podrá trabajar con la ventana de tiempo brindada (15 minutos, 1 horas, 7 días, entre otras). Para el caso de este tablero se usaran gráficas que indiquen el total de dispositivos de la arquitectura y los octetos que entran y salen de la sede, también se muestra el uso de CPU y estado de la temperatura por dispositivo. Además por cada interfaz se muestra su estado

de conexión, su cantidad de octetos salientes y entrantes y una tabla al final con la totalidad de sus interfaces, descripción y velocidad de la tecnología usada en bps. El siguiente tablero muestra el resultado obtenido a través de todo procedimiento anterior:

Figura 18: Dashboard general de dispositivos e interfaces Éxito Macondo. Recurso propio

Seleccionando el dispositivo de router como filtro se obtienen los siguientes resultados:



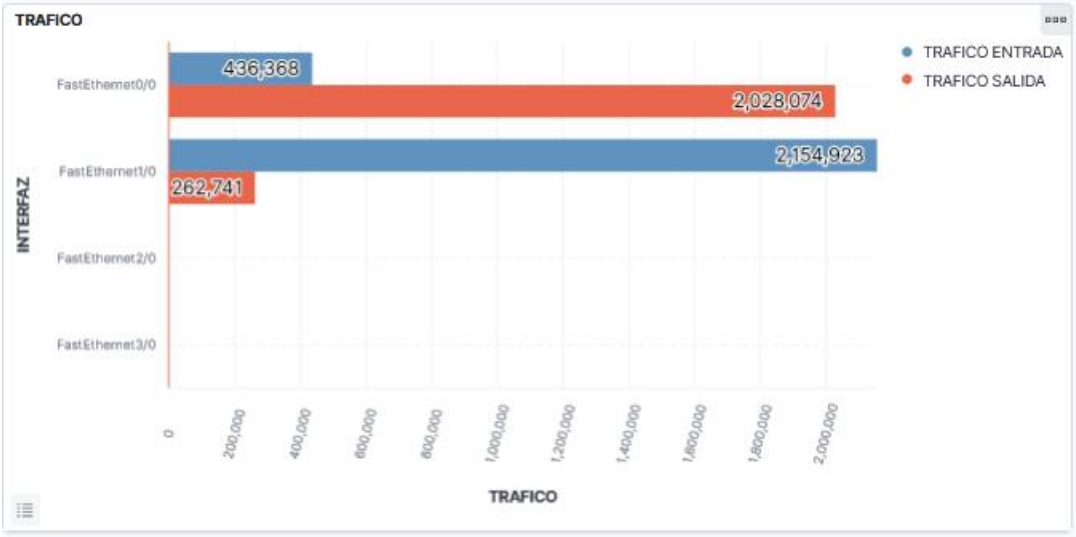


TABLA INTERFAZ

INTERFAZ	DESCRIPCION	VELOCIDAD BPS
FastEthernet0/0	ENL CONECTADO A NUBE	100,000,000
FastEthernet1/0	ENL CONECTADO A LAN SEDE	100,000,000
FastEthernet2/0		100,000,000
FastEthernet3/0		100,000,000

Figura 19: Dashboard filtrado. Recurso propio

VI. Conclusiones.

El uso de herramientas de monitoreo como ELK permite un análisis reactivo ante cambios en la normal operatividad del sistema visualizados en los tableros disponibles. Sin embargo, un mejor manejo a experimentar con base en lo ya obtenido hasta ahora se lograría con un enfoque más proactivo es con el uso de alertas reales, ya que si no están bien configuradas podría tratarse con falsas alarmas y de allí desgaste de recursos.

Para el caso los access point, tan importantes y numerosos en la compañía, su inclusión en monitoreo de sedes reales debe de ser una prioridad. Debido a la dificultad de obtener los releases de estos instrumentos de una forma licenciada no fue posible incluirlos en este proyecto que fue basado en la emulación. En contraste, pueden hacerse pruebas de laboratorio para la obtención de los OID en equipos reales y es allí donde se debe tener en cuenta las referencias de los distintos dispositivos switches, router, access points y demás donde los OID pueden cambiar.

El plan de acción recomendado a este punto del proyecto es seguir mejorando la visualización del tablero, incluyendo cambios de colores en las gráficas respectos a los niveles perjudiciales para los equipos y así alertar al analista del cambio avistado. Temas como un alto porcentaje del uso de la cpu, sobrepasar un límite impuesto de la temperatura en caso de equipos reales, desconexiones de interfaces, bajo o muy alto número de octetos salientes o entrantes que su vez dan indicios del tráfico, entre muchos otros futuros valores como la memoria o un posible bloqueo de los equipos.

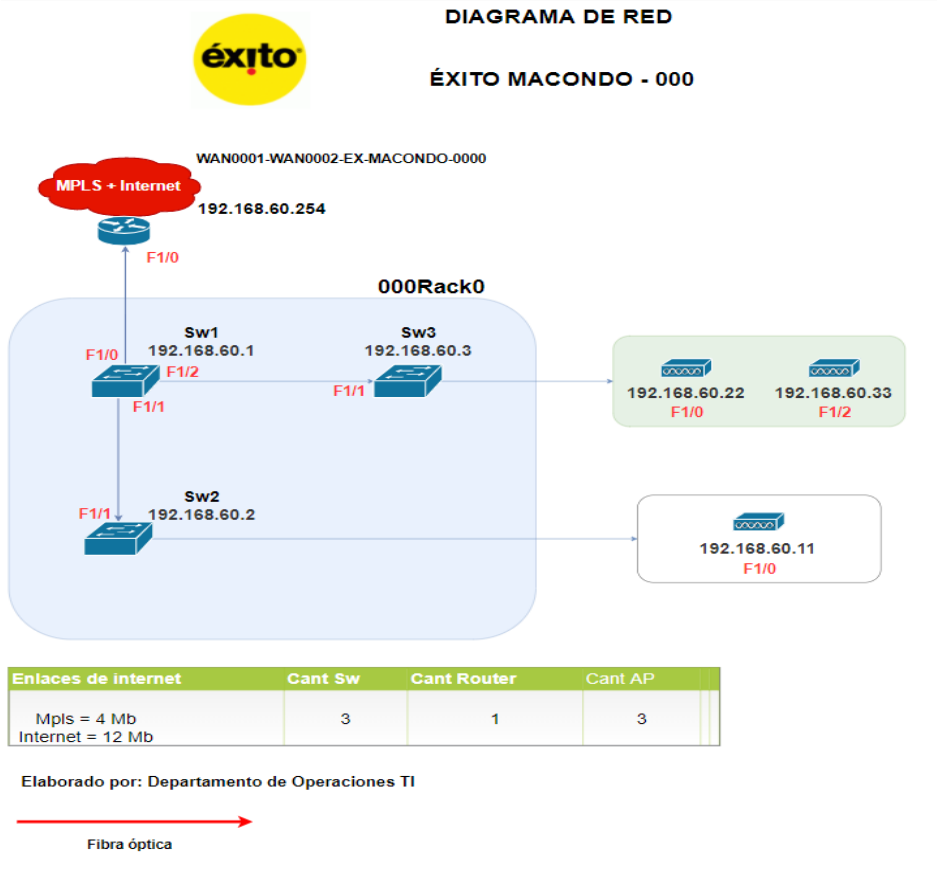
Los planes de acción, tomando en cuenta las alertas generadas, serán realizar reinicios de los equipos según grados de temperatura, saturación de puertos, alto consumo de CPU, entre muchas otras causas. También, con el fin de llevar un registro, distintas informaciones del estado de los equipos pueden nutrir una base de datos la cual puede ser usada para dar seguimiento a las fallas presentadas. Otra manera de mejorar el tiempo de acción dado es brindando una petición automática al software de gestión de eventos dado por la compañía.

Como paso final es importante tener en cuenta la capacidad analítica y el potencial con la que cuenta ELK en lo referente a aprendizaje de máquina. Con un constante flujo de datos separados y caracterizados, los algoritmos de ELK pueden detectar anomalías y casos atípicos en los momentos que sean necesarios según se pueda prever con datos actuales e históricos mediante aprendizaje no supervisado y supervisado.

Para Grupo Éxito y su enorme infraestructura será posible obtener los datos trabajados en este proyecto diferenciados según su sede de origen y pudiendo filtrar desde la generalidad de dicha tienda hasta una interfaz específica. En caso de replicarse en varias tiendas y equipos reales, junto con las recomendaciones anteriormente mencionadas los beneficios se verán radicados en corto plazo.

VII. Anexos.

1. Arquitectura de red emulada.



2. Manual de usuario proceso de conciliación.

Por motivos de seguridad corporativa el manual de usuario no podrá ser compartido en este informe, ya que presenta información de funciones y bases de datos claves de la infraestructura de red de las sedes de Grupo Éxito en todo el país.

Sin embargo, la entrada y salida del proceso ya fueron descritas en la sección "*Definición de arquitectura según reportes y existencias*" y su apartado "*Automatización del proceso de conciliación de incidentes que afectan SLA*".

VII. Referencias bibliográficas

[1] Infrastructure Monitoring. Retrieved 11 January 2020. From https://www.sumologic.com/glossary/infrastructure-monitoring/

[2] Sector Retail. Retrieved January 11 2020. From https://economipedia.com/definiciones/sector-retail.html

[3] The History of Monitoring Tools. Retrieved January 11 2020. From https://www.sumologic.com/blog/monitoring-tools-history/

[4] ¿Qué es el ELK Stack?. Retrieved January 11 2020. From, https://www.elastic.co/es/what-is/elk-stack

[5] About Cacti. Retrieved January 11 2020. From, https://www.cacti.net/

[6] What can Nagios help you do?. Retrieved January 11 2020. From, https://www.nagios.org/

[7] Prometheus Monitoring System And Time Series Database. Retrieved January 11 2020. From, https://prometheus.io/

[8] Zabbix The Enterprise-Class Open Source Network Monitoring Solution. Retrieved January 11 2020. From, https://www.zabbix.com/

[9] SolarWinds Network Availability Monitoring. Retrieved April 15 2020. From, https://www.solarwinds.com/es/network-performance-monitor/use-cases/network-availability-monitoring

[10] Axiros Axtract Monitoring And Management Of Customer QoE For Data And VoIP Services. Retrieved April 15 2020. From, https://www.axiros.com/products/axtract-qoe-monitoring

[11] ELK For Network Operations. Retrieved April 15 2020. From, https://operational.io/elk-for-network-operations/

[12] SNMP input plugin Logstash. Retrieved April 15 2020. From, https://www.elastic.co/guide/en/logstash/current/plugins-inputs-snmp.html

[13] Monitoring your network infrastructure with ELK stack. Retrieved April 15 2020. From, https://gns3.com/community/blog/monitoring-network-infratsructur

[14] ¿Qué es SNMP?. Retrieved April 15 2020. From, https://www.manageengine.com/es/network-monitoring/what-is-snmp.html

[15] Grupo Éxito. Retrieved April 15 2020. From, https://www.grupoexito.com.co/es

[16] Fundamentos de SLA: Una guía completa. Retrieved April 15 2020. From, https://freshservice.com/es/sla/

[17] Concepto azul de service provider de internet. Retrieved April 15 2020. From, https://es.dreamstime.com/concepto-azul-de-service-provider-internet-image108220869

[18] Características de la Arquitectura de Red. Retrieved April 16 2020. From, http://cidecame.uaeh.edu.mx/lcc/mapa/PROYECTO/libro35/12_caractersticas_de_la_arquitectura_de_red.html

[19] Getting Started with GNS3. Retrieved April 16 2020. From, https://docs.gns3.com/docs/

[20] Cisco cisco3640 3600 Series Router. Retrieved April 16 2020. From, https://www.amazon.com/cisco-cisco3640-3600-series-router/dp/b0000516jw

[21] Conectar a internet un router de GNS3 (GNU/LINUX). Retrieved April 17 2020. From, https://rm-rf.es/conectar-a-internet-un-router-de-gns3-gnulinux/

[22] Funcionamiento de SNMP. Retrieved April 17 2020. From, https://www.itesa.edu.mx/netacad/networks/course/module8/8.2.1.6/8.2.1.6.html#:~:text=La%20MIB%20organiza%20variables%20de,ID%20de%20objeto%20(OID).&text=En%20la%20ilustraci%C3%B3n%201%2C%20se,definida%20por%20Cisco%20Systems%2C%20Inc

[23] Temperature Status (Cisco) Check. Retrieved April 17 2020. From, https://documentation.n-able.com/remote management/userguide/Content/Services_TempStatusCisco.htm