Diseño e Implementación de una solución de gestion centralizada de logs de aplicaciones, sistemas y dispositivos basada en Logstash que permita la creación de cuadros de mando para explorar, analizar y monitorear eventos de seguridad.

Nombre Estudiante: Byron Alfonso Carrión Ramírez

Programa: Máster Universitario en Seguridad de las Tecnologías de la Información y de las Comunicaciones (MISTIC)

Nombre Consultor: Pau del Canto

Centro: Ancert (Agencia Notarial de Certificación)

Fecha entrega: 15 de junio de 2015

FICHA DEL TRABAJO FINAL

Título del trabajo:	Diseño e Implementación de una solución de gestion centralizada de logs de aplicaciones, sistemas y dispositivos basada en Logstash que permita la creación de un cuadro de mando para explorar, analizar y monitorear eventos de seguridad.
Nombre del autor:	Byron Alfonso Carrión Ramírez
Nombre del consultor:	Pau del Canto
Fecha de entrega (mm/aaaa):	06/2015
Área del Trabajo Final:	
Titulación:	Máster Universitario en Seguridad de las Tecnologías de la Información y de las Comunicaciones (MISTIC)

Resumen del Trabajo (máximo 250 palabras):

Actualmente en relación a la creciente demanda por obtener información de seguridad reelevante a partir de la enorme cantidad de registros de logs generados por las diferentes fuentes existentes en un entorno TIC se evidencia una escasez de soluciones de gestión de logs que se puedan ajustar a la realidad de la pequeña y mediana empresa. Este trabajo de investigación aplicada evalúa la pertinencia del proyecto de código abierto Logstash para el procesamiento, transporte, almacenamiento, búsqueda y anàlisis de logs. Esta evaluación incluye el anàlisis, diseño e implantación de la solución en una organización escogida.

Los resultados obtenidos demuestran la capacidad de LogStash, en términos de escalabilidad, rendimiento, seguridad y operatividad, para gestionar la vasta cantidad de logs de seguridad generados en las diferentes fuentes de la empresa. Además se evidenció durante el desarrollo del proyecto que gran parte del éxito o fracaso de la implantación de un sistema de gestión de logs depende fundamentalmente de su alineamiento a los objetivos de seguridad de la empresa.

Abstract (in English, 250 words or less):

Currently in relation to the increasing demand to get relevant security information from the massive amount of logs generated in the multiple sources available in the IT environment there is a scarcity of log management alternatives suitable for small and medium-sized enterprises. This applied research project evaluates the feasibility of the open source log management project LogStash to transport, process, store, search and analyze logs. This

evaluation process includes the analysis, design and implementation of the solution into a specific organization.
The results show the scalability, performance, security and usability of the LogStash solution to manage the huge amount of logs generated in the different sources of the business. Nevertheless it was reflected in the course of the development of the project that the success or failure of the log management deployment mainly depends in its alignment to the security goals of the enterprise,

Palabras clave (entre 4 y 8):

Logs, Gestión de Logs, ELK, Análisis de Logs, ElasticSearch, LogStash, Kibana

Índice

Lista de figuras

1. Introducción

1.1 Contexto y justificación del Trabajo

Actualmente la mayoría de sistemas de información y comunicación dejan evidencias de su estado, operación y resultados en forma de logs con el objetivo de ofrecer a administradores, desarrolladores, operadores y usuarios información detallada respecto a su funcionamiento. Esta información procesada a través de los mecanismos adecuados podría convertirse en una base de datos de eventos con utilidad en diversos fines, entre los cuales se encuentran: Administración de recursos, detección de intrusiones, la resolución de problemas, análisis forense y auditorías.

Lamentablemente, en la mayoría de los casos se desaprovecha esta enorme capacidad en potencia que ofrecen los logs debido a una extendida falta de conocimiento que oscila entre la ignorancia de la existencia de estas fuentes de información y el desconocimiento de mecanismos de gestión eficientes y eficaces.

Con respecto a los mecanismos de gestión de logs el principal inconveniente es que la ausencia de un modelo de referencia estándar para su gestión obliga el diseño de arquitecturas propietarias, en el mejor de los escenarios basadas en guías de buenas prácticas, que no garantizan una definición común de formato, sintaxis, contenido, método de transporte o almacenamiento de los logs.

No obstante que en el mercado actual existen diversas aplicaciones comerciales muy completas orientadas al manejo de logs, éstas se enfocan principalmente a medianas y grandes empresas capaces de afrontar los elevados costos asociados a su implementación y soporte.

Por este motivo la meta de esta investigación es analizar y evaluar una solución integral de código abierto que ofrezca niveles escalabilidad, rendimiento y facilidad de operación en la gestión centralizada de logs disponible para escenarios de pequeña y mediana empresa.

1.2 Objetivos del Trabajo

Objetivo General

- Diseñar e Implementar una solución de gestion centralizada de logs de aplicaciones, sistemas y dispositivos basada en Logstash que permita la creación de cuadros de mando para explorar, analizar y monitorear eventos de seguridad.

Objetivos Específicos

- Analizar la arquitectura de Logstash para el Procesamiento y Transporte de Logs.
- Implementar la solución SIEM ELK (ElasticSearch + LogStash + Kibana) combinando la funcionalidad de Logstash con las capacidades de almacenamiento, búsqueda y análisis de ElasticSearch, más las capacidades de visualización y exploración de

Kibana.
- Integrar la implementación ELK con herramientas avanzadas de correlación de eventos, análisis estadístico y monitoreo en tiempo real.

1.3 Enfoque y método seguido

El proyecto consiste de una investigación aplicada con un enfoque cualitativo sobre el problema de la Gestión de Logs. Durante el desarrollo de la investigación se evaluará la factibilidad, pertinencia y aplicabilidad de la solución ELK basada en Logstash para la centralización, análisis y monitoreo de eventos, para lo cual se definirá un escenario modelo de aplicación en base a los requerimientos obtenidos a partir de las necesidades identificadas durante el diseño.

1.4 Productos Obtenidos

El principal producto obtenido es el proceso de evaluación de la solución de código abierto ELK orientada a pequeñas y medianas organizaciones que no pueden afrontar todos los costos incluidos en la adquisición de sistemas de gestión de logs propietarios de larga escala.

1.5 Estructura de la memoria

En el capítulo 2 se definen los conceptos y terminología básicos que se requieren para aproximarse hacia el estudio de la gestión de logs. Además se desarrolla una exploración superficial de las alternativas disponibles actualmente en el mercado.
El capítulo 3 define el escenario de aplicación para la evaluación de la solución ELK. En esta descripción se incluyen la planificación y el diseño de la arquitectura de gestión de logs ELK aplicada al escenario específico.
En el capítulo 4 se procede a la implementación de la solución definida en el capítulo anterior, que incluye la configuración de las diferentes fuentes de generación de logs identificadas, además de la estructura de LogStash para la entrada, filtro y salida hacia la base de datos de búsqueda ElasticSearch.
El capítulo 5 detalla el empleo de la herramienta de visualización y exploración de los datos almacenados en ElasticSearch, Kibana para la creación de una consola de control. Además se introduce el anàlisis avanzado de los mediante la herramienta de correlación SEC y la herramienta de monitoreo Nagios.
Finalmente en las conclusiones se procede a evaluar los resultados de la solución con relación a los objetivos definidos en la etapa de planificación.

2. Principios básicos de la Gestión de Logs.

Previo a la evaluación de la solución ELK para la gestión de Logs es imprescindible conocer la terminología, conceptos y soluciones que se emplearán en el desarrollo de este proyecto.

2.1 Ciclo de Vida de un Log.

El log como unidad de registro de eventos ocurridos en sistemas y redes posee un ciclo de vida que inicia en su generación en la fuente de información hasta su eliminación.

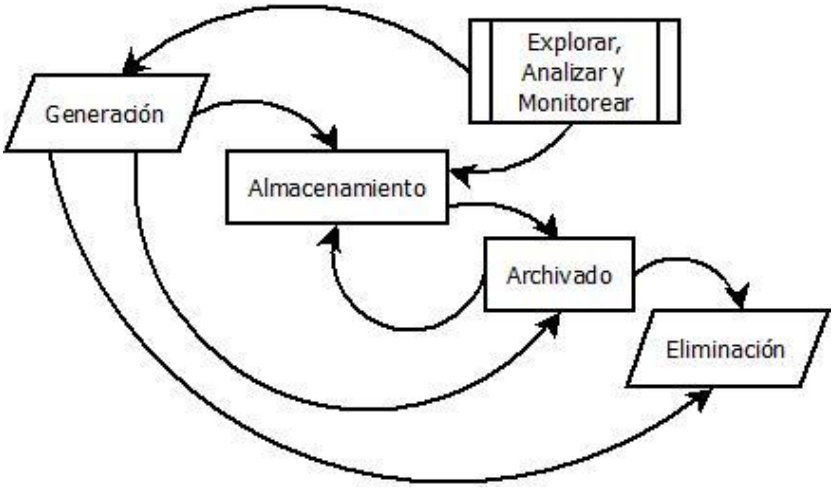


Ilustración 1: Ciclo de Vida de un Log

2.1.1 Generación de Logs

En un entorno TIC la mayoría de elementos son capaces de generar diferentes tipos de logs. Con referencia a [1, 32] estos logs se pueden clasificar en:

- Logs de Seguridad, que se enfocan en eventos de detección y respuesta ante ataques, infección de código malicioso, robo de datos y otros incidentes de seguridad.
- Logs de Operaciones, que se produce para proveer información útil respecto a la ejecución de tareas y procesos en los sistemas.
- Logs de Depuración de Aplicaciones, este tipo específico de logs se utiliza por programadores en ambientes de desarrollo (aunque su empleo no se recomienda también se pueden habilitar en ambientes de producción) para la verificación de la funcionalidad de la aplicación evaluada.

De acuerdo a [1, 35] la función de generación de logs incluye tres componentes básicos: el mecanismo de transporte, la sintaxis y el formato del log, además de una taxonomía de eventos.

Componente	Descripción	Protocolos
Mecanismo de transporte	Define la forma de mover los logs de una ubicación a otra.	Syslog (UDP, TCP, Encrypted), SOAP over HTTP, SNMP.

Formato y Sintaxis	Define la estructura interna del log, básicamente cómo están formados, cómo se almacenan, y como se presentan al usuario para su consumo.	Syslog, W3C ELF, Apache Logs, Cisco SDEE/CIDEE, XML, y muchos formatos abiertos y propietarios.
Taxonomía de Eventos	Define el tipo de información registrada en los logs.	No existe un estándar definido.

Con respecto a la taxonomía de eventos, aunque es cierto que no hay oficialmente un estándar definido para especificar los eventos que se registran en los logs, en el siguiente listado [1, 45-46] se detallan las operaciones básicas presentes en los diferentes tipos de logs (Seguridad, Operaciones y Depuración de Aplicaciones):

Operaciones	**Descripción**
Gestión de Cambios	Registro de cambios al sistema, a los componentes, cuentas de usuario y actualizaciones.
Autenticación y Autorización.	Registro de decisiones de autenticación y autorización (especialmente en cuentas privilegiadas).
Acceso a Datos y a Sistemas	Registro de los accesos a los datos y a las aplicaciones.
Gestión de Amenazas.	Registro de alertas de intrusión y actividades que violan la política de seguridad.
Gestión del Rendimiento y de la Capacidad	Registro de actividades que dan una medida del rendimiento de los sistemas y de sus capacidades disponibles.
Gestión de la Disponibilidad y de la Continuidad del Negocio	Registro de mensajes relacionados al estado del sistema en un momento específico.
Errores y Fallas	Registro de errores y fallas en los sistemas.
Mensajes de Depuración	Registros de mensajes utilizados para verificar la funcionalidad de un sistema o aplicación.

2.1.2 Almacenamiento, Archivado y Eliminación de Logs

Debido a la disponibilidad exigida por los requerimientos de anàlisis, los logs recolectados desde sus fuentes de generación primarias necesitan de algún tipo de almacenamiento secundario.
Sin embargo, la elección del formato y el tipo de almacenamiento secundario a utilizar debe ser el resultado de la aplicación de una política de retención de logs, que además de definir el almacenamiento a utilizar, permite especificar detalles respecto a la rotación, el archivado y la eliminación de los logs.
De acuerdo a [1, 72] toda política de retención de logs debería considerar los siguientes aspectos:

Aspectos	**Descripción**
Evaluar requerimientos de cumplimiento estándares aplicables	Actualmente existen varios estándares disponibles en el mercado: PCI DSS, NERC, etc.
Revisión de la política organizacional con respecto al riesgo aceptado	El tiempo de retención de los logs varia de acuerdo al tiempo que la organización considera útiles los logs (especialmente de seguridad).
Número de fuentes y cantidad de logs generados	Permite calcular la cantidad de espacio de almacenamiento requerido por unidad de tiempo para mantener los logs.
Revisión de la tecnologías de almacenamiento disponibles, y al	Permite seleccionar la mejor alternativa de almacenamiento en función del precio,

alcance de la organización.	capacidad, y velocidad de acceso y recuperación de datos.

Con relación a las tecnologías disponibles actualmente para ofrecer almacenamiento se debería considerar las alternativas con presencia “en la nube”, como por ejemplo hadoop (https://hadoop.apache.org/). Estas soluciones ofrecen altos niveles de escalabilidad, disponibilidad y rendimiento.

2.1.3 Exploración, Análisis y Monitoreo de Logs

Para que la información contenida en los logs recolectados y almacenados desde las diversas fuentes de generación tenga utilidad para la organización debe estar disponible para el acceso de los usuarios autorizados.
Sin embargo, en la mayoría de ocasiones, debido a la enorme cantidad de información almacenada, el simple acceso y exploración manual de logs se convierte en una tarea poco eficiente además de improductiva.
Estos escenarios requieren la adopción de técnicas de anàlisis que procesen los logs existentes de acuerdo a un criterio derivado de los objetivos específicos a la organización.

Técnicas	**Descripción**	**Herramientas**
Filtro de Logs	Permite seleccionar un subconjunto de logs de acuerdo a un criterio específico.	grep, awk, sed, Kibana.
Técnicas Estadísticas	Permite aplicar funciones estadísticas al conjunto universo de logs con el objetivo de obtener resúmenes.	Kibana.
Correlación de Logs	Permite relacionar diferentes logs mediante un criterio determinado.	SEC.
Minería de Datos	Permite el descubrimiento de patrones y relaciones en los datos recolectados.	

Adicionalmente existe la posibilidad de que la solución de gestión de logs interactue con el sistema de alertas y monitoreo, proporcionado información útil contenida en los logs.

2.2 Fuentes Generadoras de Logs.

Los múltiples sistemas en capacidad de generar logs se denominan fuentes generadoras de logs, y se clasifican generalmente de acuerdo a dos criterios: por el tipo de sistema y por el tipo de mecanismo utilizado.

2.2.1 Clasificación por el tipo de sistema

Con relación al tipo de sistema las fuentes de generación de logs se clasifican en: Dispositivos de Red y Seguridad, Sistemas Operativos y Aplicaciones.
Cada fuente genera información específica relacionada a su tipo:

Fuente Generadora	**Información generada**
Dispositivos de Red y Seguridad	Logins y Logouts.
	Conexión establecida a un servicio.
	Total de bytes transferidos
	Reinicialización.
	Cambios de configuración.

Sistemas Operativos	Autenticación.
	Inicio, apagado y reinicialización del sistema.
	Inicio, apagado y reinicialización de servicios.
	Finalización inesperada de servicios.
	Mensajería de estado.
Aplicaciones	Actividades de usuarios.
	Actividades de usuarios privilegiados.
	Actividades críticas.
	Reconfiguraciones.

2.2.2 Clasificación por el tipo de mecanismo utilizado

De acuerdo al mecanismo utilizado las fuentes de generación de logs se clasifican en dos categorías generales: ***push-based*** y ***pull-based***. [1, 51-52]
En la categoría ***push-based*** se encuentran todas las fuentes que utilizan Syslog, SNMP y Eventos Windows como mecanismos de generación y transporte.

Mecanismo	**Descripción**	**Tipo de Sistema**
Syslog	Originalmente utilizado en Sistemas Unix, comprende un servicio (syslogd) que escucha peticiones en el puerto UDP 514 y un archivo de configuración (/etc/syslog.conf). Adoptado como estándar para la generación de logs en dispositivos de interconexión por IETF en el RFC 3194 y 5424.	Dispositivos de Red y Seguridad, Sistemas Operativos y Aplicaciones tipo (*nix).
SNMP	El protocolo SNMP se utiliza para la configuración y consulta de estados de dispositivos de red y seguridad. De acuerdo al RFC, SNMP define dos roles: un administrador y un agente.	Dispositivos de Red y Seguridad.
Eventos Windows	Sistema de generación y recolección de logs para sistemas Microsoft. Incluye eventos de Aplicaciones, Seguridad y Sistema.	Sistemas Operativos Windows, Aplicaciones

En la categoría ***pull-based*** se utilizan aplicaciones especificamente desarrolladas para obtener los logs directamente desde las fuentes generadoras, o desde bases de datos asociadas a las fuentes.

2.3 Soluciones para la Gestión de Logs

De acuerdo a [1, 243] existen tres alternativas para atender la gestión de los logs en una organización: desarrollar una solución personalizada, adquirir una solución lista, o contratar un servicio externo de gestión de logs.

Solución	**Ventaja Principal**	**Desventaja Principal**
Desarrollo	- Solución a la medida de los requerimientos de la organización, y libertad en la selección de herramientas.	- Se requiere asignar recursos y tiempo al proyecto. Además de no contar con soporte externo.
Adquisición	- Solución lista y probada, con actualizaciones y soporte disponibles.	- Además de los costos involucrados, la organización debe adaptarse a los requisitos de la solución.

Contratación	- No se requieren dedicar recursos ni tiempo al mantenimiento y operación de la solución.	- Riesgos de seguridad debido a la pérdida de control de los datos.

Además de las alternativas seleccionadas en este proyecto: ElasticSearch, LogStash y Kibana; existen múltiples herramientas, propietarias y de código abierto o gratis, disponibles para implementar una solución de gestión de logs [1, 247-265].

Herramienta	Tipo	Licencia	Descripción
grep	Análisis Simple de Logs	Código Abierto	Permite la búsqueda de patrones en archivos de logs planos de texto http://gnuwin32.sourceforge.net/packages/grep.htm.
awk	Extracción y Análisis Simple de Logs	Código Abierto	Permite extraer y procesar información de archivos de logs planos de texto. http://www.gnu.org/software/gawk/manual/gawk.html.
Microsoft Log Parser	Análisis Simple de Eventos Windows.	Gratis	Permite filtrar la información contenida en fuentes de logs en ambientes Windows. http://www.microsoft.com/en-us/download/details.aspx?id=24659.
sed	Búsqueda, Extracción y Sustitución Simple de Logs	Código Abierto	Permite extraer y procesar información de archivos de logs planos de texto. http://www.gnu.org/software/sed/manual/sed.html.
syslog	Centralización de Logs	Código Abierto	Permite recolectar los logs de múltiples fuentes y centralizarlas en un servidor.
rsyslog	Centralización de Logs	Código Abierto	Permite recolectar los logs de múltiples fuentes y centralizarlas en un servidor.
Snare	Centralización de Eventos Windows	Gratis	Permite recolectar los eventos Windows de múltiples fuentes y centralizarlas en un servidor en un formato compatible con syslog. http://www.intersectalliance.com/projects/index.html.
OSSEC	Recolección y Análisis de Logs	Código Abierto	Permite la centralización, almacenamiento y retención de logs, además de la creación de reglas para analizarlos. http://www.ossec.net
OSSIM	SIEM	Código Abierto y Comercial	Permite descubrimiento de activos, evaluación de vulnerabilidades, detección de amenazas, monitoreo de comportamiento e inteligencia de seguridad. http://www.alienvault.com
Splunk	Recolección y Análisis de Logs	Gratis y Comercial	Permite la centralización, almacenamiento y anàlisis de logs. http://www.splunk.com
NetIQ Sentinel	SIEM	Gratis y Comercial	Permite, además de la gestión de logs, detección de anomalías y manejo de identidades. https://www.netiq.com/products/sentinel/
IBM QRadar	Recolección y Análisis de Logs	Comercial	Permite la centralización, almacenamiento y anàlisis de logs. http://q1labs.com/products.aspx
Loggly	Proveedor de servicios de gestión de logs en la nube	Comercial	Permite la gestión de logs en la nube. http://loggly.com

3 Definición del Escenario de Aplicación

3.1 Planificación de la Gestión de Logs

En la fase de planificación se incluyen las siguientes tareas:

- Planificación del Proyecto
- Definición de objetivos.
- Definición de roles y responsabilidades.
- Selección de fuentes generadoras.

3.1.1 Planificación del Proyecto

Actividad	Inicio	Fin	# de Horas
Conceptos Teóricos acerca de la Gestión de Logs	16/03/2015	20/03/2015	20
Definición del Escenario de Aplicación	21/03/2015	29/03/2015	30
Recolección, Almacenamiento y Búsqueda Centralizada de Logs	30/03/2015	17/04/2015	70
Entrega PEC2	Diseño e Implementación de la Arquitectura Configuración de las Fuentes de Logs. Almacenamiento Centralizado de Logs.		
Análisis y Monitoreo de Eventos de Seguridad	18/04/2015	18/05/2015	115
Entrega PEC3	Cuadro de Mando para la Visualización y Exploración de Eventos. Cuadro de Mando de Monitoreo y Generación de Alertas.		

Name	Begin date	End date
Conceptos Teóricos acerca de la Gestión de Logs	3/16/15	3/20/15
Definición del Escenario de Aplicación	3/21/15	3/28/15
Recolección, Almacenamiento y Búsqueda Centralizada de Logs	3/30/15	4/16/15
Entrega PEC2	4/17/15	4/17/15
Diseño e Implementación de la Arquitectura	4/17/15	4/17/15
Configuración de las Fuentes de Logs	4/17/15	4/17/15
Almacenamiento Centralizado de Logs	4/17/15	4/17/15
Análisis y Monitoreo de Eventos de Seguridad	4/17/15	5/16/15
Entrega PEC3	5/18/15	5/18/15
Cuadro de Mando para Visualización de Eventos	5/18/15	5/18/15
Cuadro de Mando de Monitoreo	5/18/15	5/18/15
Memoria Final	3/16/15	6/13/15
Entrega PEC4	6/15/15	6/15/15
Presentación/Video	6/16/15	6/22/15
Defensa del TFM	6/23/15	7/3/15

Ilustración 2:Planificación del Proyecto

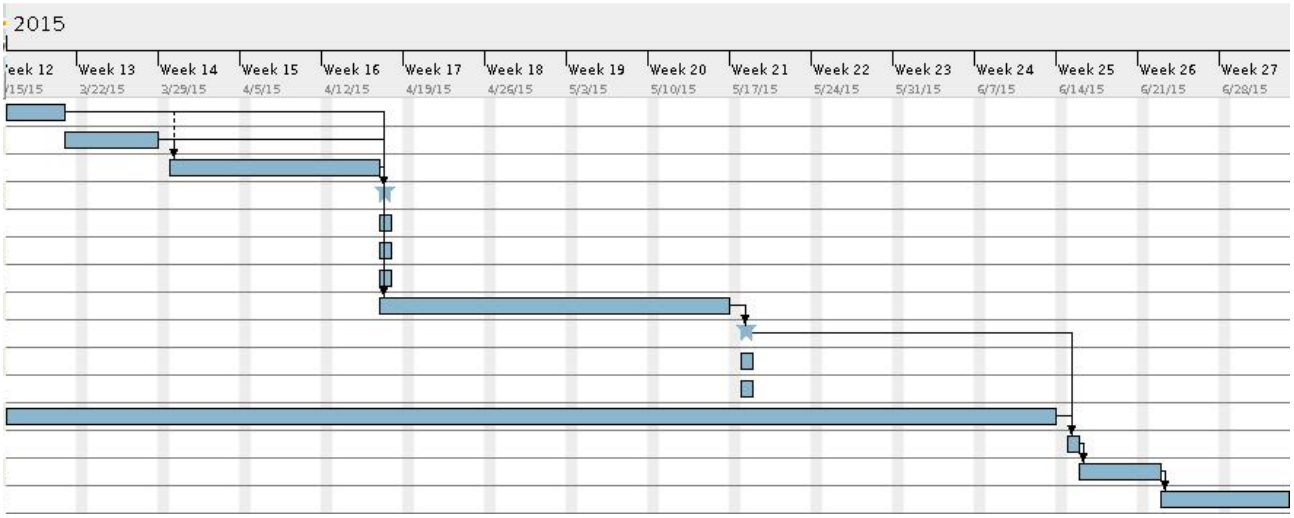


Ilustración 3: Diagrama de Gantt del Proyecto

3.1.2 Definición de Objetivos de la Gestión de Logs

No obstante la extrema importancia de realizar una adecuada gestión de los logs en toda organización, la implementación de una solución debería ser el resultado de un proyecto que empiece por un análisis de las metas y requerimientos que exige cada escenario desde su particular situación y la identificación de los objetivos generales y específicos a conseguir mediante el análisis de la información recolectada. La correcta definición de estos objetivos permitirá a la organización:

- Alinear la política de gestión de logs respecto a la política de gestión de las TIC.
- Justificar la asignación de recursos ante el alto mando ejecutivo.
- Orientar el desarrollo del proyecto hacia la consecución de objetivos específicos.
- Evaluar el funcionamiento de la solución con relación a las metas establecidas.

3.1.2.1 Objetivos Generales de la Gestión de Logs

Estos objetivos determinarán las metas y propósitos de la gestión de los logs:

- Complementar la funcionalidad proporcionada por las herramientas de monitoreo y alertas existentes en la organización mediante la información presente en los logs.
- Disponer de una herramienta de generación de reportes que permita la exploración y recuperación en línea e histórica de información requerida por auditoria.

3.1.2.2 Objetivos Específicos de la Gestión de Logs

Estos objetivos definirán las fuentes generadoras de log y el tipo de información que se necesitan recolectar, además del procesamiento de los datos y las consultas y reportes que se requieren obtener:

- Detectar la presencia o inminencia de errores en dispositivos de red y seguridad, sistemas operativos y aplicaciones.
- Detectar los accesos e intentos de acceso no autorizados de administración a los switches.
- Detectar los accesos e intentos de acceso no autorizados a la red inalámbrica.
- Detectar los accesos e intentos de acceso no autorizados a las sesiones de red Windows de los usuarios internos.
- Detectar los accesos e intentos de acceso no autorizados a los servicios locales ofrecidos en la red interna.
- Visualizar la distribución del consumo de la conexión de Internet por parte de los usuarios internos.
- Visualizar la distribución del consumo de los servicios ofrecidos en la red interna.

3.1.3 Identificación de Roles y Responsabilidades para la Gestión de Logs

La identificación de los diferentes tipos de usuarios del sistema de gestión de logs facilita la asignación de responsabilidades respecto a la operación del sistema, además permite definir los roles específicos para el consumo de la información obtenida.
Aunque este proceso de identificación es pertinente a cada organización, a continuación se presenta un listado de roles y responsabilidades comúnmente utilizadas: [1, 369]

Rol	**Responsibilidad**
Administrador del Sistema	Gestión y Mantenimiento del sistema.
Equipo de Seguridad	Configuración de logging en dispositivos de seguridad.
Ejecutivos	Consumidores de alto nivel de los resultados del sistema.
Administrador de TI	Encargado de gestionar la estandarización del uso del sitema.
Soporte Técnico	Consumidores operativos de información generada por el sistema.
Equipo de Respuesta a Incidentes	Consumidores investigativos de los resultados del sistema.
Auditores	Verifican que el sistema se encuentre acorde a regulaciones.

3.1.4 Selección de las Fuentes Generadoras de Logs

A partir de la definición de los objetivos específicos de la gestión de logs se pueden identificar las fuentes generadoras de logs y el tipo de información que se encuentran disponibles en la organización.
Sin embargo, el proceso de identificación de las fuentes disponibles de logs exige el conocimiento básico de la topología de la organización.
3.1.4.1 Topología Lógica de la Organización

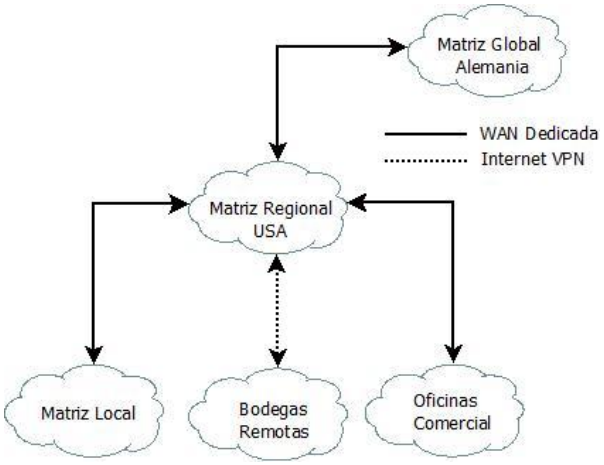


Ilustración 4: Topología Lógica de la Organización

El alcance del proyecto incluye a la Matriz Local, Oficinas Comercial y Bodegas Remotas, por lo que se requiere identificar los servicios ofertados en cada localidad. Estos servicios se clasifican en los servicios gestionados localmente por la administración local y los servicios gestionados globalmente a los que la administración local no tiene acceso.
Los servicios gestionados globalmente disponibles para las localidades incluyen:

- Controlador de Dominio Distribuido Microsoft.
- ERP Transaccional Cliente/Servidor.
- Proxy Sophos para la administrar la conexión local a Internet.
- Correo electrónico Lotus Notes.
- Sistema de Antivirus Mcaffe

* Debido al tipo de interconexión (WAN Dedicada e Internet VPN) entre las localidades se requiere minimizar la recarga en los enlaces de con tráfico asociado a la solución de gestión de logs.

3.1.4.2 Topología Lógica de la Matriz Local

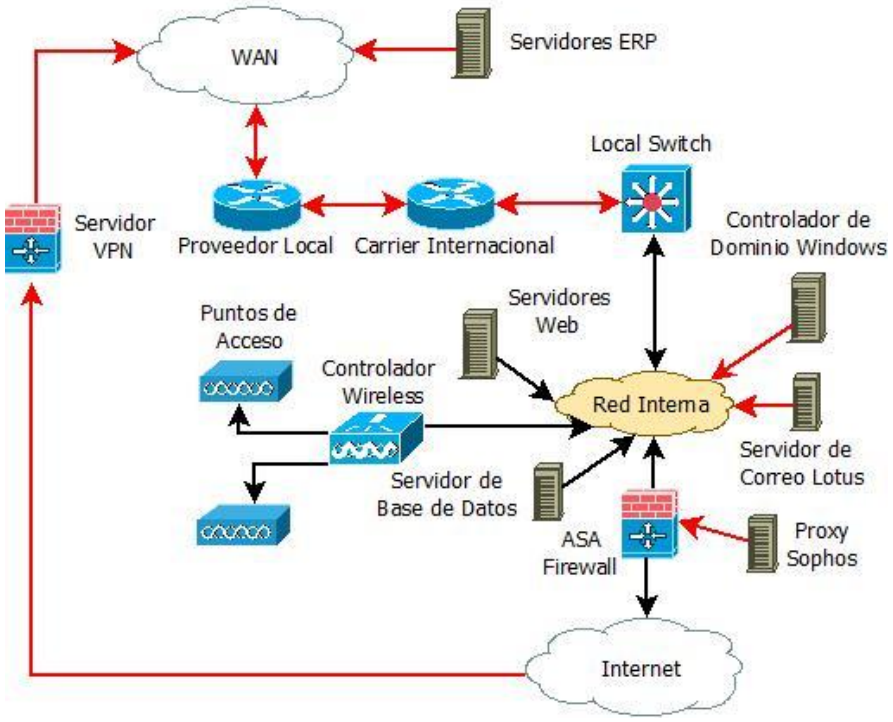


Ilustración 5: Topología Lógica de la Matriz Local

En la topología lógica de la Matriz Local los recursos gestionados localmente se encuentran marcados con enlaces en negrita, mientras que los recursos gestionados globalmente utilizan enlaces en rojo.
Los servicios gestionados localmente que ofrece la organización en esta localidad a sus usuarios incluyen:

- Enlace de Respaldo a través de Internet mediante VPN (Cisco ASA Firewall).
- Base de datos para aplicaciones locales (Servidor Microsoft SQL Server).
- Aplicaciones web (Servidores Microsoft IIS y Apache httpd).
- Acceso Inalámbrico (Controlador Inalámbrico y Punto de Acceso)
- Acceso Alámbrico Ethernet (Switches).

3.1.4.3 Topología Lógica de las Oficinas Comercial

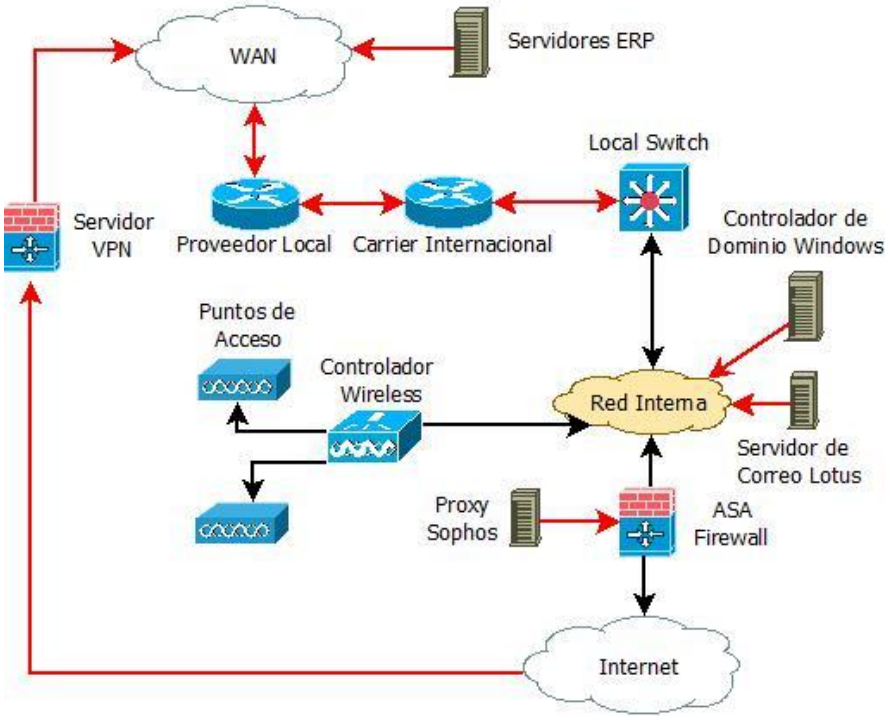


Ilustración 6: Topología Lógica de la Oficinas Comercial

Igual que en la topología anterior, la topología lógica de las Oficinas Comercial los recursos gestionados localmente se encuentran marcados con enlaces en negrita, mientras que los recursos gestionados globalmente utilizan enlaces en rojo.
Los servicios gestionados localmente que ofrece la organización en esta localidad a sus usuarios incluyen:

- Enlace de Respaldo a través de Internet mediante VPN (Cisco ASA Firewall).
- Acceso Inalámbrico (Controlador Inalámbrico y Punto de Acceso)
- Acceso Alámbrico Ethernet (Switches).

3.1.4.4 Topología Lógica de las Bodegas

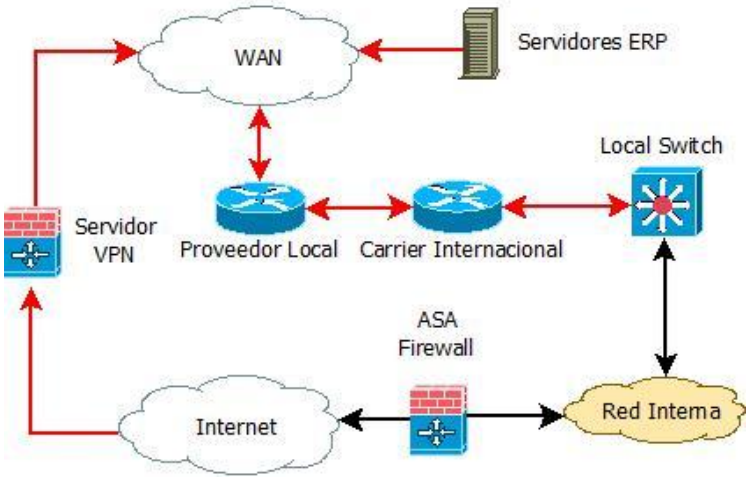


Ilustración 7: Topología Lógica de las Bodegas

Igual que en las topologías anteriores, la topología lógica de las Bodegas los recursos gestionados localmente se encuentran marcados con enlaces en negrita, mientras que los recursos gestionados globalmente utilizan enlaces en rojo.
Los servicios gestionados localmente que ofrece la organización en esta localidad a sus usuarios incluyen:

- Enlace de Respaldo a través de Internet mediante VPN (Cisco ASA Firewall).
- Acceso Alámbrico Ethernet (Switch).

3.1.4.5 Fuentes Generadoras de Logs

Las potenciales fuentes de recolección de logs corresponden a dispositivos de red y seguridad, sistemas operativos y aplicaciones que proporcionan los diferentes servicios identificados en cada una de las localidades de la organización:

Tipo de Fuente	Fuente	Modelo
Dispositivos de Red y Seguridad	Switches	Cisco WS-C3750X-24
		Cisco WS-C2960S-48TS-L
		3COM 5500 3CR17254-91
		3COM 4500 3CR17561-91
		DLink DES-3028
	Firewall	Cisco ASA 5505 Series
	Wireless LAN Controller (WLC)	Cisco AIR-CT2504-K9
	Monitor de Red	Ntopng Traffic Monitor
Sistemas Operativos	RHEL	RHEL Server 6.0
	Windows Server	2012 R2 Standard
	Windows Client	Windows 7 Enterprise
Aplicaciones	Bases de Datos	Microsoft SQL Server 2012
	Servidores Web	Apache httpd 2.2.15
		IIS 8

No todas las potenciales fuentes de logs identificadas se utilizarán sino exclusivamente aquellas cuya información contenida en los logs permitan

alcanzar los objetivos específicos planteados. Por lo tanto se requiere verificar que fuentes se requieren para alcanzar cada uno de los objetivos:

Objetivos Específicos	**Switches**	**Firewall**	**WLC**	**Monitor de Red**	**RHEL**	**Windows Server**	**Windows Client**	**Databases**	**Web Servers**
Detectar la presencia o inminencia de errores en dispositivos de red y seguridad, sistemas operativos y aplicaciones.	X	X	X	X	X	X		X	X
Detectar los accesos e intentos de acceso no autorizados de administración a los dispositivos de interconexión	X	X	X						
Detectar los accesos e intentos de acceso no autorizados a la red inalámbrica.			X						
Detectar los accesos e intentos de acceso no autorizados a las sesiones de red Windows de los usuarios internos						X			
Detectar los accesos e intentos de acceso no autorizados a los servicios locales ofrecidos en la red interna					X	X		X	X
Visualizar la distribución del consumo de la conexión de Internet por parte de los usuarios internos				X					
Visualizar la distribución del consumo de los servicios ofrecidos en la red interna.					X	X		X	X

4 Diseño del Sistema de Gestión de Logs

En la fase de diseño se incluyen las siguientes tareas:

- Evaluación de alternativas y selección de la solución.
- Definición de una política.
- Diseño de la arquitectura.
- Listado del software a utilizar.

4.1 Evaluación y Selección de la Solución de Gestión de Logs

Aunque este análisis trata acerca de la gestión de logs mediante la herramienta de código abierto Logstash, el proceso de selección de una solución que se ajuste a los requerimientos y presupuesto específicos de la organización es una tarea ardua que incluye la evaluación de las alternativas disponibles en el mercado.
La evaluación de los componentes de la solución debería incluir el análisis de los siguientes parámetros: [1, 371-372]

- Costo total de propiedad (TCO).
- Tipo de Licencia y Soporte ofrecido.
- Actualizaciones de la aplicación.
- Características principales de la solución.
- Documentación y Capacitación de los usuarios.
- Extensibilidad de la solución.

Además, en caso de las soluciones disponibles no satisfagan los requerimientos, o el presupuesto de la organización; actualmente existe la posibilidad de tercerizar los servicios de gestión de logs mediante un Managed Security Service Provider.
En este proyecto se evaluará la funcionalidad de LogStash como parte de la solución SIEM ELK (ElasticSearch, LogStash y Kibana).

4.1.1 LogStash

LogStash (https://www.elastic.co/products/logstash) is una aplicación java de código abierto escrita en JRuby con el objetivo de transportar, recolectar, filtrar e indexar logs. La arquitectura de LogStash está compuesta por tres componentes principales en forma de plugins: Entrada, Filtro y Salida.

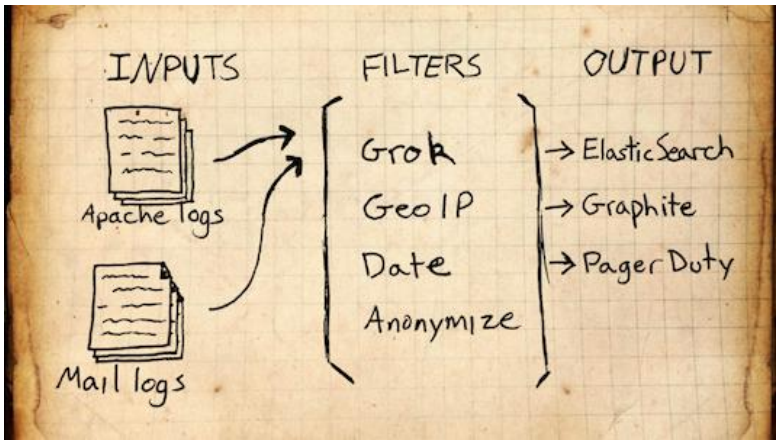


Ilustración 8: Arquitectura de LogStash

https://www.elastic.co/assets/blte4300041993ef383/Logstash%20Image.png

Los plugins de Entrada habilitan el soporte de LogStash para diversas fuentes generadoras de logs. La lista completa de plugins soportados se puede consultar en https://www.elastic.co/guide/en/logstash/current/input-plugins.html.
Los plugins de Filtro permiten el procesamiento, incluidos filtrado y normalización, de los logs recolectados en las fuentes. La lista completa de plugins está accesible en https://www.elastic.co/guide/en/logstash/current/filter-plugins.html.
Los plugins de Salida posibilitan el envío de los logs recolectados y procesados a su destino final. La lista completa de plugins es accesible en https://www.elastic.co/guide/en/logstash/current/output-plugins.html.

4.1.2 ElasticSearch

ElasticSearch (https://www.elastic.co/products/elasticsearch) es una base de datos no relacional (NoSQL) de almacenamiento con funciones incorporadas de búsqueda de texto y análisis de datos.
Las principales funcionalidades que ofrece ElasticSearch incluyen: acceso y análisis a los datos en tiempo real, escalabilidad a través de una arquitectura distribuida, alta disponibilidad, múltiples índices(Multitenancy), orientación a documentos (JSON), interfaz para el desarrollo de aplicaciones (RESTful API).

4.1.3 Kibana

Kibana (https://www.elastic.co/products/kibana) es una herramienta de visualización y exploración de datos. Entre las principales características que presenta se encuentran: integración completa con ElasticSearch, vistas personalizadas, análisis incorporado, soporte multiorigen.

4.2 Definición de una Política de Gestión de Logs

Una política de gestión de logs incluye la definición de un conjunto de procedimientos que resuman las necesidades, requerimientos y recursos de la organización. La política debe especificar:

- La información que deben generar las fuentes.

- Los mecanismos de transporte que se utilizarán para recolectar la información desde las fuentes hacia la central de almacenamiento.
- El tiempo de retención de la información en las fuentes y en la central de almacenamiento..

Para la determinación de los logs que deben generar las diversas fuentes se utilizará la clasificación que propone la RFC 5424 (https://tools.ietf.org/html/rfc5424) correspondiente al estándar Syslog:

Código	Severidad	Descripción
0	Emergencia (Emergency)	Sistema no utilizable
1	Alerta (Alert)	Acción requerida inmediatamente.
2	Crítica (Critical)	Condiciones críticas
3	Error (Error)	Condiciones de error.
4	Advertencias (Warning)	Condiciones de advertencia.
5	Noticia (Notice)	Condiciones normales.
6	Informacional (Informational)	Mensajes informacionales
7	Depuración (Debug)	Mensajes de depuración.

*Con relación al objetivo específico: “Detectar la presencia o inminencia de errores en dispositivos de red y seguridad, sistemas operativos y aplicaciones” todas las fuentes involucradas deberán generar mínimo logs con código 3 (Error).

La información requerida de las fuentes de generación de log para el resto de objetivos específicos se lista a continuación:

Objetivos Específicos	Fuente	Información requerida	Mecanismos
Detectar los accesos e intentos de acceso no autorizados de administración a los dispositivos de interconexión.	Switches, Firewall, WLC	El acceso a estos dispositivos se realiza a través de SSH por lo que se necesita la IP y el usuario que intenta la conexión.	Syslog, Logs mediante ACLs
Detectar los accesos e intentos de acceso no autorizados a la red inalámbrica.	WLC	En el acceso fallido a la red inalámbrica se requiere registrar la dirección MAC del cliente que realiza el intento.	Syslog
Detectar los accesos e intentos de acceso no autorizados a las sesiones de red Windows de los usuarios internos	Windows Server	Se require registrar la dirección IP, el usuario windows y el dominio en el que se intenta la conexión.	GPOs.
Detectar los accesos e intentos de acceso no autorizados a los servicios locales ofrecidos en la red interna	RHEL, Windows Server, Database, Web Server	Se requiere registrar la dirección IP del cliente y el servicio solicitado.	Syslog, Configuración de Logging en cada Aplicación.
Visualizar la distribución del consumo de la conexión de Internet por parte de los usuarios internos	Monitor de Red	Se requiere registrar la IP origen y destino, el puerto origen y destino, además del número de bytes transmitidos y recibidos.	Monitoreo de Puertos.
Visualizar la distribución del consumo de los servicios ofrecidos en la red interna.	RHEL, Windows Server, Database, Web Server	Se requiere registrar la dirección IP del cliente y el servicio solicitado.	Syslog, GPOs.

El desarrollo de una política de retención de los logs en las fuentes y en la central de almacenamiento debe tomar en cuenta los siguientes parámetros: [1, 374-]

- La existencia y requerimientos exigidos por estándares de cumplimiento adoptados por la organización. Por ejemplo PCI DSS (Payment Card Industry Data Security Standard).
- El nivel de riesgo aceptado por la organización con relación al periodo de tiempo en que se desee analizar la información contenida en los logs.
- Cantidad total de logs generados por las diversas fuentes, generalmente medida en bytes por unidad de tiempo.
- Mecanismos de almacenamiento y archivado de información disponibles en la organización.

Con respecto a la gran cantidad de información que generan las fuentes de logs, la aplicación LogStash, además de proporcionar mecanismos para procesamiento y transporte de logs, funciona como un filtro con reglas programadas para mantener la información que servirá para cumplir los requerimientos especificados en la política de logs de la organización, y descartar aquella información que no se va a utilizar.
Aunque en la organización no existen estándares de cumplimiento respecto a los logs generados por las diferentes fuentes, a partir de las necesidades de análisis de la información recolectada se ha creado una matriz de retención de logs:

Fuente	Período de Retención	Archivado
Switches	1 mes	No
Firewall	1 mes	No
WLC	1 mes	No
Monitor de Red	1 mes	No
RHEL	1 mes	No
Windows Server	1 mes	No
Databases	1 mes	No
Web Servers	1 mes	No

* No obstante esta información podria sufrir modificaciones a corto plazo debido a que la organización está a punto de entrar en un proceso de auditorías internas de TI gestionadas por la Matriz Global.

4.3 Diseño de la Arquitectura

Las características de la organización se ajustan a un diseño centralizado en la Matriz Local de la organización con colectores ubicados en las Bodegas Remotas y en las Oficinas Comercial.

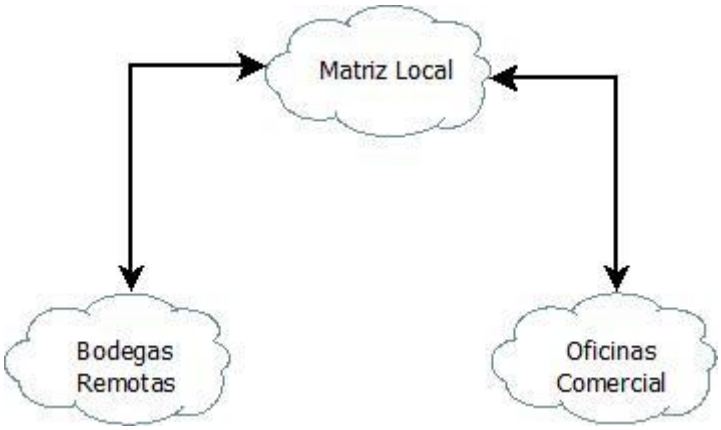


Ilustración 9: Diseño de la Arquitectura

En la Matriz Local se implementará la solución ELK mediante un recolector con Logstash que filtrará la información transmitida desde las fuentes de logs y almacenará los resultados en una instancia de ElasticSearch. Las fuentes se clasifican en dos tipos: fuentes que poseen una instalación local de LogStash (logstash-forwarder) para la transmisión y fuentes que utilizarán un agente de transporte diferente de LogStash (logstash-forwarder)

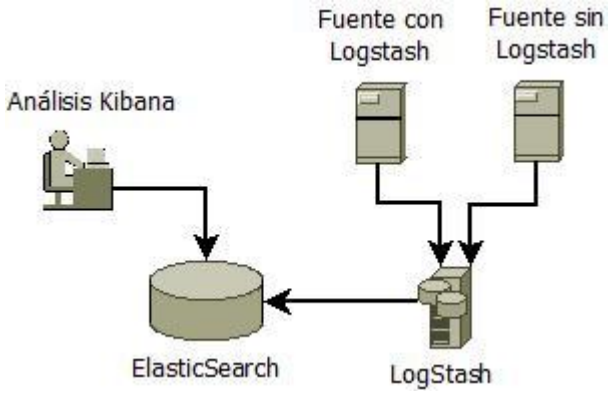


Ilustración 10: Diseño para la Matriz

En las Oficinas Comercial y Bodegas se recolectará y filtrará la información transmitida desde las fuentes con y sin LogStash (logstash-forwarder) y se almacenará temporalmente en memoria (Redis Server) para su posterior almacenamiento en la instancia central de ElasticSearch en la Matriz Local.

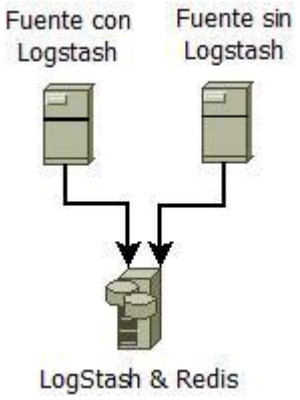


Ilustración 11: Diseño para las Oficinas Comercial y Bodegas

El diseño completo de la solución para la organización integra los módulos de la Matriz Local junto con los de las Oficinas Comercial y las Bodegas.

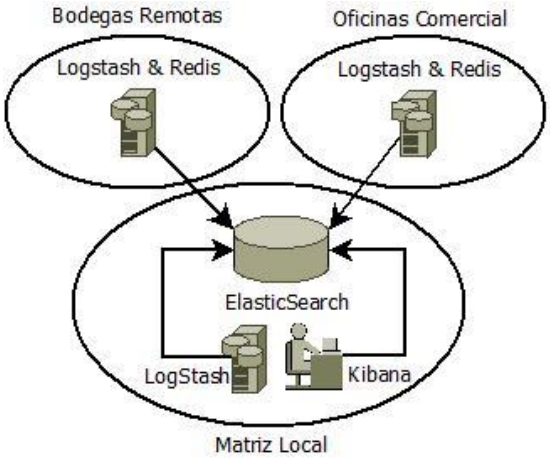


Ilustración 12: Diseño Completo de la Solución

4.3.1 Listado del Software a utilizar

Adicionalmente a las aplicaciones incluidas en ELK: ElasticSearch, LogStash y Kibana, la implementación de la solución utilizará las siguientes herramientas:

Tipo de Fuentes	Software	Descripción
Dispositivos de Red y Seguridad	syslog (https://tools.ietf.org/html/rfc5424)	Debido a que en estos dispositivos no es posible instalar LogStash* para el transporte de logs se utilizar la capacidad de transporte incluida en el protocolo syslog.
Sistemas Operativos-GNU/Linux	logstash-forward (https://github.com/elastic/logstash-forwarder/blob/master/README.md)	Logstash-forwarder es una herramienta de LogStash para recolectar localmente los logs y luego transportarlos.
Sistemas Operativos Windows	nxlog (http://nxlog.org/products/nxlog-community-edition)	Herramienta multiplataforma para recolectar localmente los logs y luego transportarlos.
Monitor de Red	ntopng (http://www.ntop.org/products/traffic-analysis/ntop/)	Aplicación de recolección y análisis de tráfico.
Caché	Redis (http://redis.io/)	Aplicación para caché y almacenamiento temporal de objetos.
Correlación	SEC (http://simple-evcorr.sourceforge.net/)	Herramienta para procesamiento y correlación avanzada de eventos.
Monitoreo	Nagios XI (https://www.nagios.com/products/nagiosxi)	Herramienta para monitoreo

* Logstash puede funcionar como un agente instalado en las fuentes que soporten Java. Sin embargo, en este proyecto se ha decidido utilizar otras aplicaciones (logstash-forwarder y nxlog) que funcionan como agentes en lugar de LogStash. Los dos motivos principales para esta elección están relacionados con Java, en primer lugar por la gran cantidad de recursos que utiliza, y además por las múltiples vulnerabilidades de seguridad.

5. Recolección y Almacenamiento Centralizado de Logs

5.1 Implementación de la Arquitectura

5.1.1 Instalación del Servidor Logstash

Para detalles de la instalación y configuración del servidor LogStash por favor consultar el Anexo 1.

5.1.2 Instalación del Servidor ElasticSearch

Para detalles de la instalación y configuración del servidor ElasticSearch por favor consultar el Anexo 2.

5.2 Configuración de las Fuentes Generadoras de Logs

La configuración de las fuentes generadoras de logs se realizará utilizando diferentes mecanismos de transporte.

5.2.1 Mecanismos de Transporte para las Fuentes Generadoras de Logs

Tipo de Fuente	Fuente	Transporte
Switches	Cisco WS-C3750X-24	Syslog
	Cisco WS-C2960S-48TS-L	
	3COM 5500 3CR17254-91	
	3COM 4500 3CR17561-91	
	DLink DES-3028	
Firewalls	Cisco ASA 5505 Series	
Wireless LAN Controller	Cisco AIR-CT2504-K9	
Traffic	Ntopng Traffic Monitor	ElasticSearch
RHEL	RHEL Server 6.0	logstash-forwarder
Windows Server	2012 R2 Standard	Nxlog
Database	Microsoft SQL Server 2012	
Web Servers	Apache httpd 2.2.15	logstash-forwarder
	IIS 8	Nxlog

5.2.1.1 Configuración de Switches

Para facilitar el procesamiento de LogStash se exportarán los logs de cada tipo de switch a un puerto UDP diferente:

Tipo de Switch	Puerto UDP
Cisco	5514
3Com	514
DLink	55514

Configuración Switches Cisco

Los switches cisco modelos 3750X y 2960S con IOS 15.0 o superior para exportar logs al servidor LogStash requieren de la siguiente configuración:

Comando	Descripción
clock timezone ECT -5 0	Especificación de la zona horaria en la que se encuentra el dispositivo.
ntp server 10.204.64.198	Especificación del servidor NTP que se utilizará para gestionar los timestamps en los logs reportados.
logging facility local6	Especificar el tipo de facility que se utilizará para exportar los logs.
logging host 10.204.71.246 transport udp port 5514	Especificación del host al que se exportarán los logs (Servidor LogStash) el tipo de protocolo que se utilizará (UDP) y el número de puerto (5514).

A continuación se puede observar la configuración aplicada en uno de los switches mediante el comando *show logging*

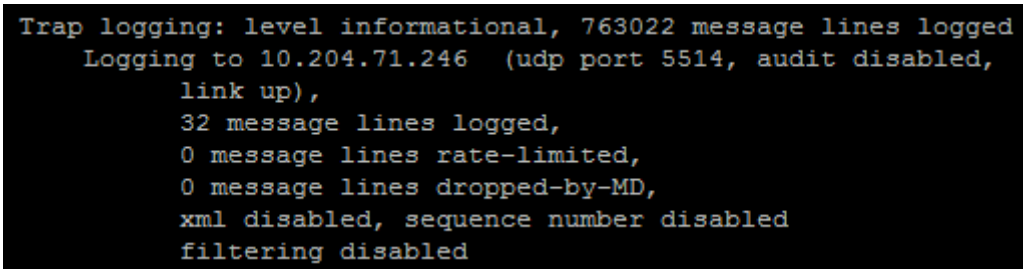


Ilustración 13: Captura resultado comando show login

Configuración Switches 3Com

Los switches 3Com modelos 5500 y 4500 requieren de la siguiente configuración:

Comando	Descripción
ntp-service unicast-server 10.204.64.198	Especificación del servidor NTP que se utilizará para gestionar los timestamps en los logs reportados.
info-center loghost 10.204.71.246 facility local6	Especificación del host al que se exportarán los logs (Servidor LogStash) el tipo de facility que se utilizará. No es posible especificar el protocolo (UDP) ni el puerto (514).

Los resultados de la configuración mediante el comando *display info-center unit 1*:

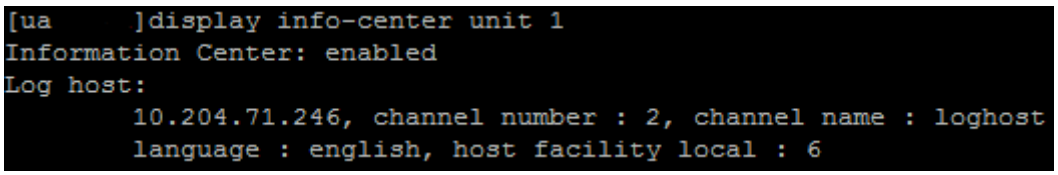


Ilustración 14: Captura resultado comando display info-center unit 1

Configuración Switches DLink

Los Switches DLink modelo DES-3028 requieren de la siguiente configuración:

Comando	Descripción
enable sntp	Habilitar del protocolo Simple NTP.
config time_zone operator - hour 5 min 0	Especificación de la zona horaria en la que se

	encuentra el dispositivo.
config sntp primary 10.204.64.198 secondary 0.0.0.0 poll-interval 720	Especificación del servidor NTP que se utilizará para gestionar los timestamps en los logs reportados.
enable syslog	Habilitar el protocol SYSLOG.
create syslog host 1 severity informational facility local6 udp_port 55514 ipaddress 10.204.71.246 state enable	Especificación del host al que se exportarán los logs (Servidor LogStash) el tipo de protocolo que se utilizará (UDP) y el número de puerto (55514).

A continuación se puede observar la configuración aplicada en uno de los switches mediante el comando *show syslog host*:

```
              :4#show syslog host
Command: show syslog host

Syslog Global State : Enabled

Host Id  Host IP Address  Severity        Facility  UDP port  Status
-------  ---------------  --------------  --------  --------  --------
1        10.204.71.246    Informational   Local6    55514     Enabled

Total Entries : 1
```

Ilustración 15: Captura resultado comando show syslog host

5.2.1.2 Configuración del Firewall ASA

La configuración para un firewall ASA 5505 (imagen asa915-k8.bin) es:

Comando	**Descripción**
ntp server 10.204.64.198	Especificación del servidor NTP que se utilizará para gestionar los timestamps en los logs reportados.
logging enable	Habilitar el proceso de logging.
logging timestamp	Agrega en los logs el timestamp del evento.
logging host inside 10.204.71.246 17/5515	Especificación del host al que se exportarán los logs (Servidor LogStash), el tipo de protocolo (17=UDP) y el número de puerto (5515).
logging trap notifications	Selecciona el logging de las excepciones que ocurran en el sistema.

A continuación el resultado de aplicar la configuración, comando *show logging*:

```
UAFW001# show logging
Syslog logging: enabled
    Facility: 20
    Timestamp logging: enabled
    Standby logging: disabled
    Debug-trace logging: disabled
    Console logging: disabled
    Monitor logging: disabled
    Buffer logging: level warnings, 90 messages logged
    Trap logging: level notifications, facility 20, 105 messages logged
        Logging to inside 10.204.71.246 udp/5515 errors: 1  dropped: 3
```

Ilustración 16: Captura resultado comando show logging

5.2.1.3 Configuración de Wireless LAN Controller

La configuración requerida para habilitar el servicio de syslog en el Wireless LAN Controller 2504 con imagen 7.5.102.0 es a través de la interfaz web. Lamentablemente en este modelo no es posible especificar el puerto en el que se escucha el servidor de syslog, por lo que utiliza por defecto UDP 514.

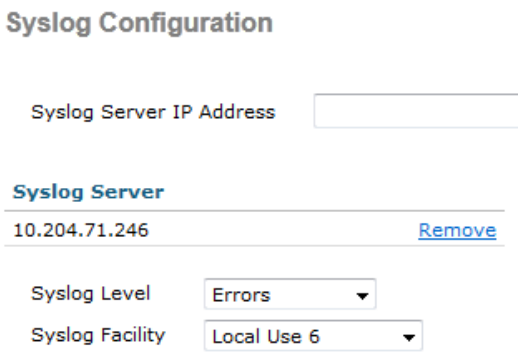


Ilustración 17: Captura configuración en el Wireless LAN Controller

5.2.1.4 Configuración de la aplicación de gestión de tráfico ntopng

La aplicación ntopng dispone de un módulo que exporta los flujos de tráfico capturados desde la red directamente a ElasticSearch en formato JSON normalizado por lo que se obviará el paso a través de LogStash.
El nombre del índice para ntopng será ntopng-*.

```
[root@uavl001 ntopng]# more ntopng.conf
-G=/var/tmp/ntopng.pid
-i=ens224
-m=10.204.64.0/21
-r=10.204.71.246:6379
--dump-flows=es;flows;ntopng;http://uavl002:9200/_bulk
```

Ilustración 18: Captura del contenido del archivo /etc/ntopng/ntopng.conf

5.2.1.5 Configuración de RHEL Server

El sistema RHEL tiene las siguientes características:

```
[root@ualm002 ~]# cat /etc/issue
Red Hat Enterprise Linux Server release 6.6 (Santiago)
Kernel \r on an \m

[root@ualm002 ~]# uname -a
Linux ualm002.                    2.6.32-431.5.1.el6.i686 #1 SMP Fri Jan 10 14:
47:32 EST 2014 i686 i686 i386 GNU/Linux
```

Ilustración 19: Captura del contenido del archivo /etc/issue y comando uname -a

Los logs que se van exportar hacia el servidor LogStash son:

- Logs del sistema (/var/log/messages)
- Logs del servidor httpd:
  - /var/log/httpd/access_log
  - /var/log/httpd/error_log

Para el transporte de logs hacia el servidor LogStash se utilizará la aplicación logstash-forwarder de acuerdo a lo establecido en la documentación en https://github.com/elastic/logstash-forwarder/blob/master/README.md.
Detalles de la instalación y configuración de logstash-forwarder consultar en el Anexo 3.

5.2.1.6 Configuración de Windows Server

La versión de Windows Server que se utilizará es la 2012 R2 con las siguientes características:

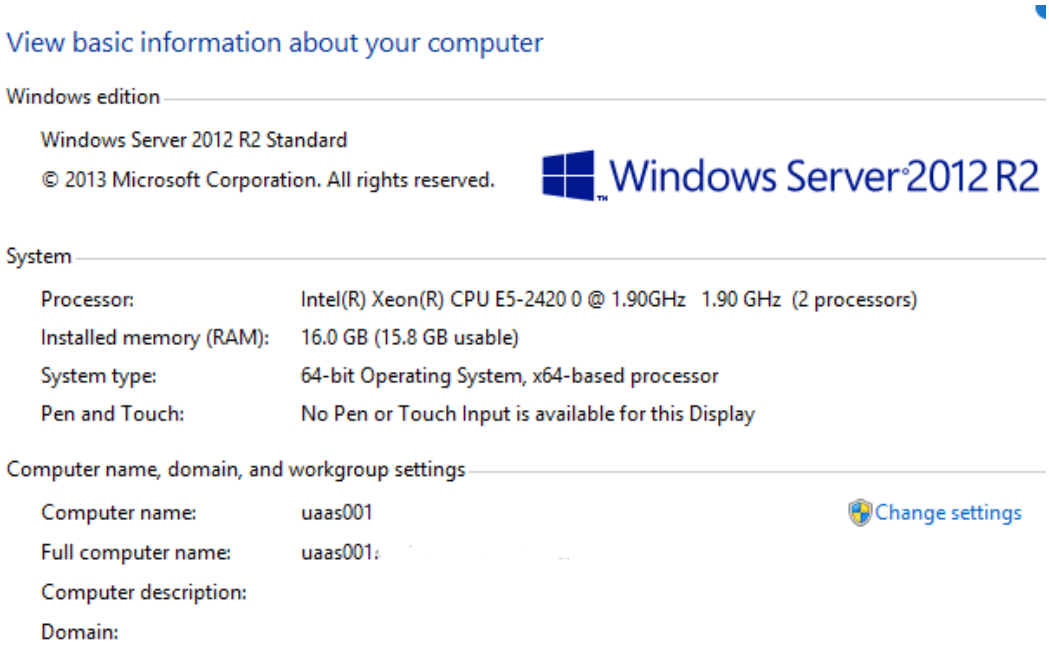
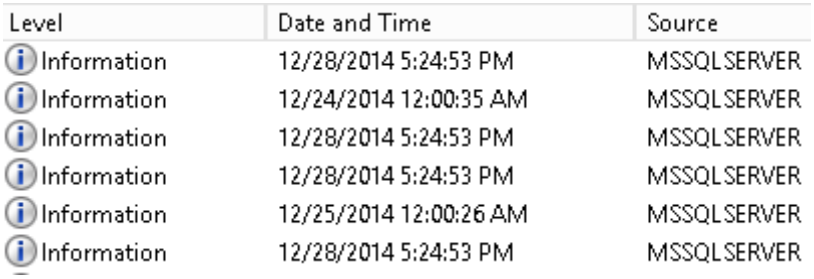


Ilustración 20: Captura del tipo de Sistema Operativo Windows

Para gestionar el transporte de los logs se utilizará la aplicación de recolección de los de código abierto nxlog. Detalles de la instalación y configuración de nxlog en el Anexo 4.

5.2.1.7 Configuración de Microsoft SQL Server

Microsoft SQL Server envía por defecto los logs de la base de datos al Windows Event Log del sistema por lo que la configuración de nxlog definida para Windows Server se utilizará para el transporte de los logs generados por SQL Server.



Level	Date and Time	Source
Information	12/28/2014 5:24:53 PM	MSSQLSERVER
Information	12/24/2014 12:00:35 AM	MSSQLSERVER
Information	12/28/2014 5:24:53 PM	MSSQLSERVER
Information	12/28/2014 5:24:53 PM	MSSQLSERVER
Information	12/25/2014 12:00:26 AM	MSSQLSERVER
Information	12/28/2014 5:24:53 PM	MSSQLSERVER

Ilustración 21: Captura del visor de sucesos del Sistema Operativo Windows

5.2.1.8 Configuración de Internet Information Server

Microsoft Internet Information Server almacena por defecto sus logs a un archivo local del sistema de archivos especificado en la configuración:

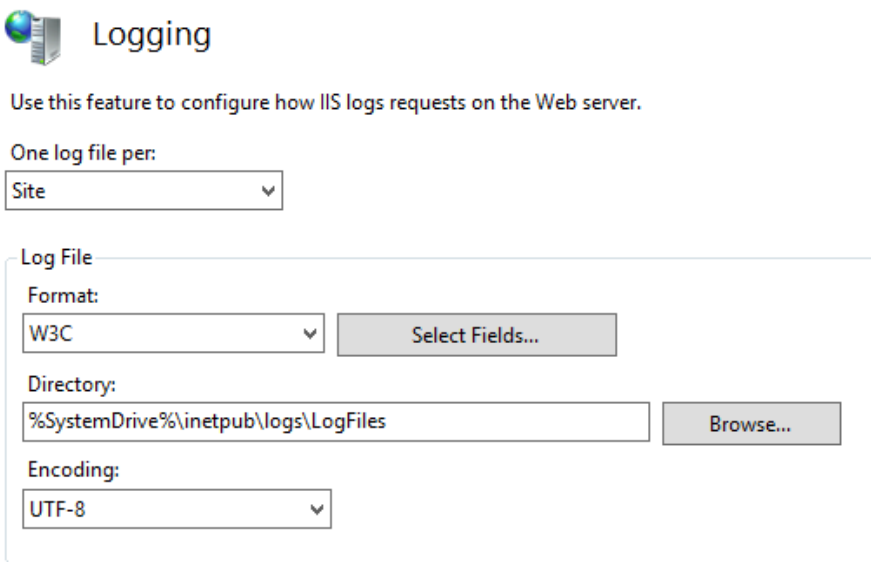


Ilustración 22: Captura de la configuración del Internet Information Server

En el presente caso el directorio correspondiente es: C:\inetpub\logs\LogFiles\W3SVC1

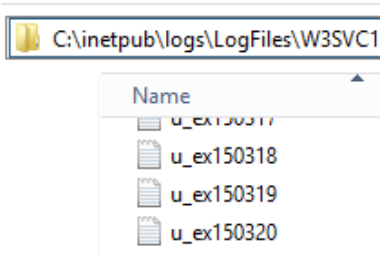


Ilustración 23: Captura del directorio donde se almacenan los logs de IIS

Aquí verificamos que la nomenclatura de los archivos es: *u_ex<año><mes><día>*.
El procesamiento de estos archivos se realizará mediante la aplicación recolectora de logs nxlog. El archivo de configuración nxlog.conf necesita incluir la extensión w3c:

```
<Extension w3c>
     Module xm_csv
     Fields $date, $time, $site, $dstip, $HTTPMethod, $URIStem,
$URIQuery, $port, $username, $srcip, $UserAgent, $HTTPStatus,
$SubStatus, $Win32Status
     FieldTypes string, string, string, string, string, string,
string, string, string, string, string, string, string, string,
     Delimiter  ' '
</Extension>
```

Ilustración 24: Captura del archivo de configuración nxlog.conf para IIS

En la sección de entrada se utiliza el módulo de archivo para leer los archivos de log, que se normalizan de acuerdo a los campos definidos en la extensión w3c, se concatenan la fecha y la hora, y se exporta en formato JSON:

```
<Input IIS_Log>
      Module     im_file
      File  "C:\\inetpub\\logs\\LogFiles\\W3SVC1\\u_ex*"
      ReadFromLast False
      SavePos    TRUE
      #Drop info legend lines
      Exec if $raw_event =~ /^#/ drop();\
           else\
           {\
                w3c->parse_csv();\
                $EventTime = parsedate($date + " " + $time);\
                $SourceName = "IIS";\
                $Message = to_json();\
           }
</Input>
```

Ilustración 25: Captura de la entrada para IIS en nxlog.conf

Estos eventos recolectados se enviarán al servidor LogStash puerto TCP 5515:

```
<Output IIS_Out>
    Module      om_tcp
    Host        uavl001
    Port        5515
</Output>
```

Ilustración 26: Captura de la salida para IIS en nxlog.conf

Finalmente se interconectan las secciones de Entrada y Salida:

```
<Route 2>
    Path        IIS_Log => IIS_Out
</Route>
```

Ilustración 27: Captura del enrutamiento para IIS en nxlog.conf

5.3 Configuración de LogStash

Debido a que LogStash tiene tres componentes: Entrada, Filtrado y Salida, se analizarán cada uno de ellos de manera independiente.
En relación a los diferentes módulos de entrada se requiere para cada uno de ellos configurar el firewall del sistema para permitir las conexiones entrantes. Para referencia de cómo realizarlo por favor revisar el Anexo 5.
Con respecto al componente de salida los resultados se indexarán y almacenarán en la instancia de ElasticSearch (10.204.71.247:9300) instalada previamente de acuerdo a la siguiente configuración:

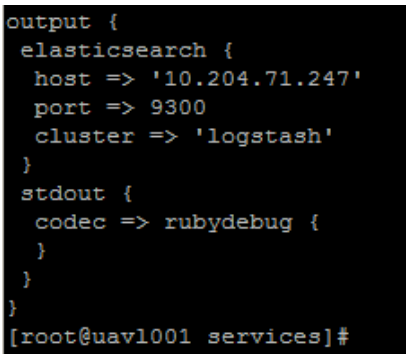


Ilustración 28: Captura de la configuración de salida LogStash

Adicionalmente en la sección salida (output) se agregó con fines de depuración el módulo stdout que además envía una copia de los registros de salida al archivo de log del sistema (/var/log/messages).
Finalmente es necesario recordar que para que los cambios en la configuración tengan efecto es necesario reiniciar el servicio mediante el comando *systemctl restart logstash*.

5.3.1 Configuración de LogStash para los switches

Entrada

En el componente de entrada de LogStash para los switches se utilizará el módulo udp, aunque existe un módulo syslog aparentemente ideal para este escenario en la implementación presenta algunos inconvenientes. Un aspecto a destacar es que cada una de las secciones abrirá un puerto UDP y asignará la variable “type” correspondiente a cada log recibido.

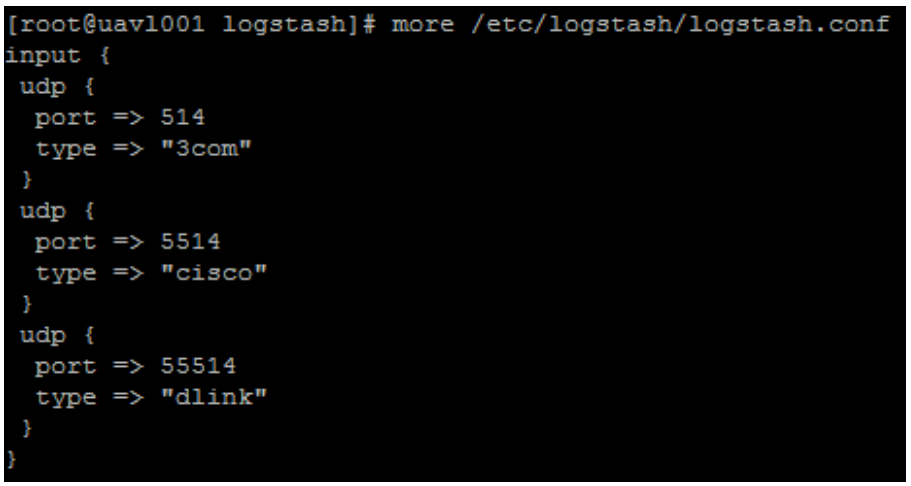


Ilustración 29: Captura de la configuración de entrada LogStash para switches

Por lo tanto al ejecutar el servicio logstash se puede verificar que existan tres procesos escuchando en los respectivos puertos mediante el comando *netstat –anu*:

```
[root@uavl001 logstash]# netstat -anu
Active Internet connections (servers a
Proto Recv-Q Send-Q Local Address
udp        0      0 0.0.0.0:37918
udp        0      0 0.0.0.0:5353
udp6       0      0 :::55514
udp6       0      0 :::5514
udp6       0      0 :::514
```

Ilustración 30: Captura del resultado del comando netstat -anu

Filtro

En el componente de filtrado de LogStash para los switches se utilizará el módulo grok con el objetivo de normalizar las entradas, a pesar que todas reportan su compatibilidad con el estándar RFC 3195 syslog, existen algunas incongruencias en sus respectivos formatos que se necesitan normalizar. Para ello se agregará el siguiente contenido en la sección filter del archivo de configuración /etc/logstash/logstash.conf

```
filter {
 if [type] == "cisco" {
  grok {
   match => [ "message", '%{GREEDYDATA}: %{CISCOTIMESTAMP:localtime}: %%{WORD:fa
cility}-%{INT:severity}-%{WORD:mnemonic}: %{GREEDYDATA:description}' ]
   tag_on_failure => ["parsing_cisco_failed"]
  }
 }else if [type] == "3com" {
  grok {
   match => [ "message", '<%{INT}>%{3COMTIMESTAMP:localtime} %{IPORHOST:hostname
} %%10%{GREEDYDATA:facility}/%{INT:severity}/%{GREEDYDATA:mnemonic}\(l\):- 1 -%{
GREEDYDATA:description}' ]
   tag_on_failure => ["parsing_3com_failed"]
  }
 }else if [type] == "dlink" {
  grok {
   match => [ "message", '<%{INT}>%{SYSLOGTIMESTAMP:localtime} %{IPORHOST:hostna
me} %{WORD:facility}: %{GREEDYDATA:description}' ]
   tag_on_failure => ["parsing_dlink_failed"]
  }
 }
}
```

Ilustración 31: Captura de la configuración de filtro LogStash para switches

Verificación

La validez de la configuración se verificará monitoreando el archivo de log del sistema mediante el comando: *tail –f /var/log/messages*, lo que permite visualizar los registros de salida en formato JSON:

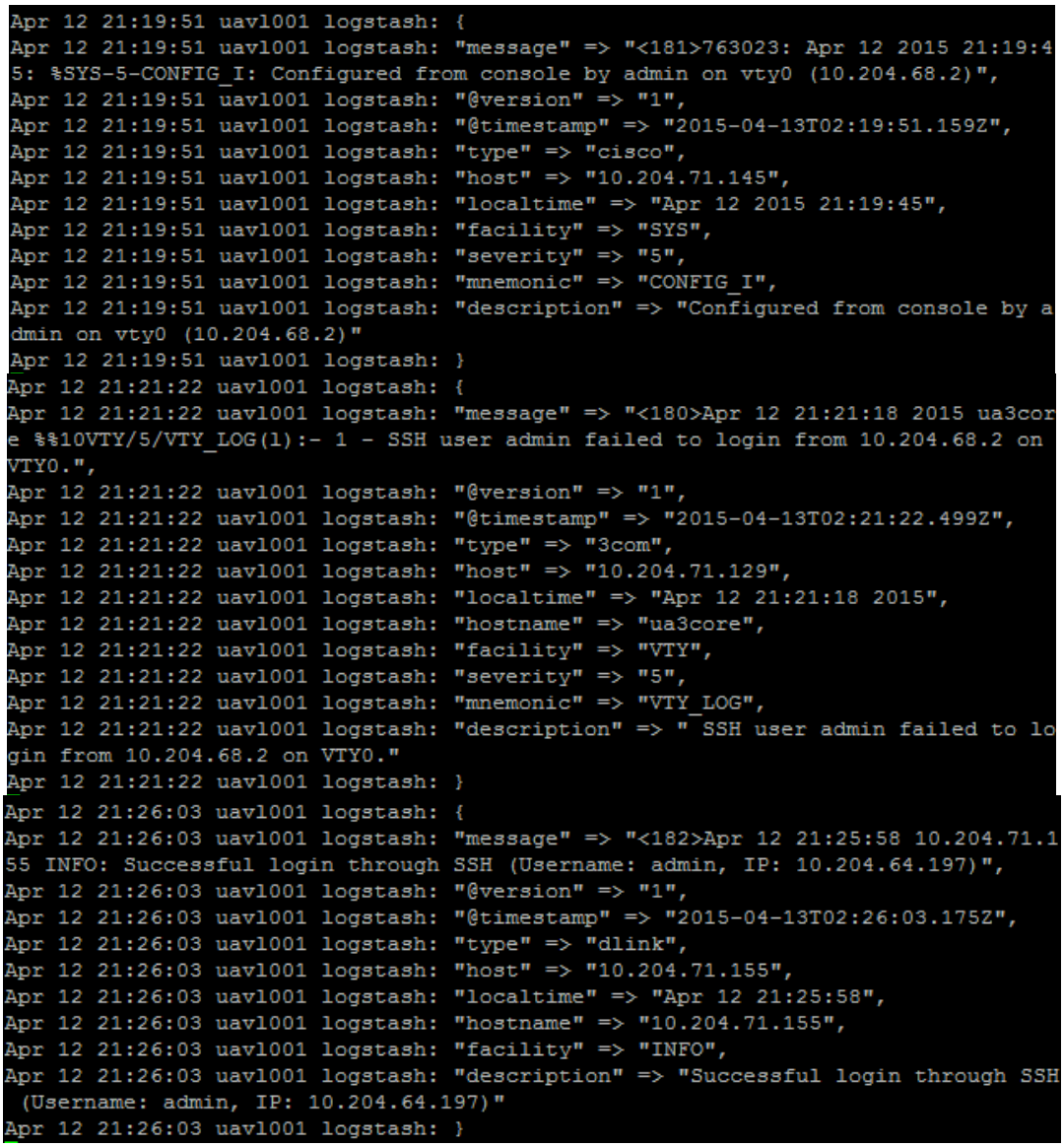


Ilustración 32: Captura de los eventos de switches por LogStash

ElasticSearch por defecto indexará el contenido de acuerdo al siguiente formato: logstash-yyyy.mm.dd. Por lo tanto una consulta al índice logstash-2015.04.13 en el servidor ElasticSearch muestra el siguiente resultado:

```
[root@uavl001 ~]# curl --noproxy 10.204.71.247 http://10.204.71.247:9200/logstas
h-2015.04.13/_search?pretty
{
  "took" : 3,
  "timed_out" : false,
  "_shards" : {
    "total" : 5,
    "successful" : 5,
    "failed" : 0
  },
  "hits" : {
    "total" : 15,
```

Ilustración 33: Captura de los registros almacenados en ElasticSearch

5.3.2 Configuración de LogStash para el firewall ASA

Entrada

En el componente de entrada de LogStash para el firewall ASA se utilizará el módulo udp con puerto 5515 y se asignará a la variable “type” el valor “asa”.

```
udp {
 port => 5515
 type => "asa"
}
```

Ilustración 34: Captura de la entrada de LogStash para ASA

Filtro

En el componente de filtrado de LogStash se utilizará el módulo grok con el objetivo de normalizar las entradas que provengan del firewall ASA.

```
}else if [type] == "asa" {
 grok {
  match => [ "message", '<%{INT}>%{CISCOTIMESTAMP:localtime}: %%{WORD:facility}
-%{INT:severity}-%{WORD:mnemonic}: %{GREEDYDATA:description}' ]
 }
```

Ilustración 35: Captura del filtro de LogStash para ASA

Salida

Los resultados se indexarán y almacenarán en la instancia de ElasticSearch (10.204.71.247:9300) instalada previamente.

Verificación

La validez de la configuración se verificará monitoreando el archivo de log del sistema mediante el comando: *tail –f /var/log/messages*, lo que permite visualizar los registros de salida en formato JSON:

```
Apr 14 17:07:07 uavl001 logstash: "message" => "<163>Apr 14 2015 17:07:01: %ASA-
3-710003: TCP access denied by ACL from 122.228.207.77/34664 to outside:186.101.
55.188/22\n",
Apr 14 17:07:07 uavl001 logstash: "@version" => "1",
Apr 14 17:07:07 uavl001 logstash: "@timestamp" => "2015-04-14T22:07:07.521Z",
Apr 14 17:07:07 uavl001 logstash: "type" => "asa",
Apr 14 17:07:07 uavl001 logstash: "host" => "172.31.255.254",
Apr 14 17:07:07 uavl001 logstash: "localtime" => "Apr 14 2015 17:07:01",
Apr 14 17:07:07 uavl001 logstash: "facility" => "ASA",
Apr 14 17:07:07 uavl001 logstash: "severity" => "3",
Apr 14 17:07:07 uavl001 logstash: "mnemonic" => "710003",
Apr 14 17:07:07 uavl001 logstash: "description" => "TCP access denied by ACL fro
m 122.228.207.77/34664 to outside:186.101.55.188/22"
Apr 14 17:07:07 uavl001 logstash: }
```

Ilustración 36: Captura de los eventos de ASA por LogStash

5.3.3 Configuración de LogStash para Wireless LAN Controller

Entrada

En el componente de entrada de LogStash para el Wireless LAN Controller se utilizará el módulo udp con puerto 514; sin embargo, es necesario tener presente que debido a que la configuración del modelo 2504 no permite especificar ni el protocolo ni el puerto se deberá utilizar los valores por defecto: UDP/514. Por este motivo es necesario realizar un pequeño cambio en la configuración, los dispositivos 3com tampoco permiten especificar el protocolo o el puerto por lo que también utiliza los valores por defecto (UDP/514). Se reemplazará *type => "3com"* por *type=> "syslog".*

```
udp {
 port => 514
 type => "syslog"
}
```

Ilustración 37: Captura de la entrada de LogStash para WLC

Filtro

En el componente de filtrado de LogStash es necesario reemplazar la sección correspondiente a "3com":

```
 }else if [type] == "3com" {
  grok {
   match => [ "message", '<%{INT}>%{3COMTIMESTAMP:localtime} %{IPORHOST:hostname
} %%10%{GREEDYDATA:facility}/%{INT:severity}/%{GREEDYDATA:mnemonic}\(l\):- 1 -%{
GREEDYDATA:description}' ]
   tag_on_failure => ["parsing_3com_failed"]
  }
```

Ilustración 38: Captura del filtro de LogStash para 3Com

El contenido ahora incluye un condicional para diferenciar los logs de los dispositivos 3com de los logs del Wireless LAN Controller, el parámetro que se utiliza para este propósito es la dirección IP del WLC (10.204.71.146):

```
 }else if [type] == "syslog" {
  if [host] == "10.204.71.146" {
        grok {
                match => [ "message", '<%{INT}>%{IPORHOST:hostname}: \*%{WORD}:
%{CISCOTIMESTAMP:localtime}: #%{GREEDYDATA:facility}-%{INT:severity}-%{GREEDYDAT
A:mnemonic}: %{GREEDYDATA:description}' ]
        }
  }else{
        grok {
                match => [ "message", '<%{INT}>%{3COMTIMESTAMP:localtime} %{IPOR
HOST:hostname} %%10%{GREEDYDATA:facility}/%{INT:severity}/%{GREEDYDATA:mnemonic}
\(l\):- 1 -%{GREEDYDATA:description}' ]
        }
  }
```

Ilustración 39: Captura del filtro de LogStash para WLC

Verificación

La validez de la configuración se verificará monitoreando el archivo de log del sistema mediante el comando: *tail –f /var/log/messages*, lo que permite visualizar los registros de salida en formato JSON:

```
Apr 15 14:44:03 uavl001 logstash: "message" => "<179>UAWL001: *Dot1x_NW_MsgTask_
0: Apr 15 14:40:48.983: #DOT1X-3-INVALID_WPA_KEY_MSG_STATE: Received invalid EAP
OL-key M2 msg in START  state - invalid secure bit; KeyLen 24, Key type 1, clien
t e0:9d:31:51:e4:b0",
Apr 15 14:44:03 uavl001 logstash: "@version" => "1",
Apr 15 14:44:03 uavl001 logstash: "@timestamp" => "2015-04-15T19:44:03.159Z",
Apr 15 14:44:03 uavl001 logstash: "type" => "syslog",
Apr 15 14:44:03 uavl001 logstash: "host" => "10.204.71.146",
Apr 15 14:44:03 uavl001 logstash: "hostname" => "UAWL001",
Apr 15 14:44:03 uavl001 logstash: "localtime" => "Apr 15 14:40:48.983",
Apr 15 14:44:03 uavl001 logstash: "facility" => "DOT1X",
Apr 15 14:44:03 uavl001 logstash: "severity" => "3",
Apr 15 14:44:03 uavl001 logstash: "mnemonic" => "INVALID_WPA_KEY_MSG_STATE",
Apr 15 14:44:03 uavl001 logstash: "description" => "Received invalid EAPOL-key M
2 msg in START  state - invalid secure bit; KeyLen 24, Key type 1, client e0:9d:
31:51:e4:b0"
Apr 15 14:44:03 uavl001 logstash: }
```

Ilustración 40: Captura de los eventos de WLC por LogStash

5.3.4 Configuración de LogStash para Windows Server

Entrada

De acuerdo a la configuración de nxlog para Windows Server el componente de entrada de LogStash correspondiente deberá utilizar el módulo tcp en el puerto 514 con formato JSON. Además para identificación posterior se asignará a la variable “type” el valor “windows”. Por lo tanto se agregará el siguiente contenido en la sección input del archivo de configuración /etc/logstash/logstash.conf:

```
tcp {
 port => 514
 type => "windows"
 format => 'json'
}
```

Ilustración 41: Captura de la entrada de LogStash para Windows Server

Filtro

En el componente de filtrado de LogStash se utilizará el módulo grokmutate con el objetivo de modificar ligeramente las entradas recibidas. La mayor ventaja es que las entradas ya se encuentran formateadas en formato JSON.
Por lo tanto se agregará el siguiente contenido en la sección filter del archivo de configuración /etc/logstash/logstash.conf

```
}else if [type] == "windows" {
       # Incoming Windows Event logs from nxlog
       # The EventReceivedTime field must contain only digits
       mutate {
       # Lowercase some values that are always in uppercase
              lowercase => [ "EventType", "FileName", "Hostname", "Severity" ]
       }
       mutate {
       # Set source to what the message says
              rename => [ "Hostname", "@source_host" ]
       }
       mutate {
       # Rename some fields into something more useful
       rename => [ "Message", "@message" ]
       rename => [ "Severity", "eventlog_severity" ]
       rename => [ "SeverityValue", "eventlog_severity_code" ]
       rename => [ "Channel", "eventlog_channel" ]
       rename => [ "SourceName", "eventlog_program" ]
       rename => [ "SourceModuleName", "nxlog_input" ]
       rename => [ "Category", "eventlog_category" ]
       rename => [ "EventID", "eventlog_id" ]
       rename => [ "RecordNumber", "eventlog_record_number" ]
       rename => [ "ProcessID", "eventlog_pid" ]
       }
       mutate {
       # Remove redundant fields
       remove => [ "SourceModuleType", "EventTimeWritten", "EventTime", "EventReceivedTime", "EventType" ]
       }
```

Ilustración 42: Captura del filtro de LogStash para Windows Server

Verificación

La validez de la configuración se verificará monitoreando el archivo de log del sistema mediante el comando: *tail –f /var/log/messages*, lo que permite visualizar los registros de salida en formato JSON:

```
Apr 13 16:47:02 uavl001 logstash: "eventlog_severity" => "info",
Apr 13 16:47:02 uavl001 logstash: "eventlog_severity_code" => 2,
Apr 13 16:47:02 uavl001 logstash: "eventlog_channel" => "Security",
Apr 13 16:47:02 uavl001 logstash: "eventlog_program" => "Microsoft-Windows-Security-Auditing",
Apr 13 16:47:02 uavl001 logstash: "nxlog_input" => "EventLog",
Apr 13 16:47:02 uavl001 logstash: "eventlog_category" => "Process Creation",
Apr 13 16:47:02 uavl001 logstash: "eventlog_id" => 4688,
Apr 13 16:47:02 uavl001 logstash: "eventlog_record_number" => 756803,
Apr 13 16:47:02 uavl001 logstash: "eventlog_pid" => 4
Apr 13 16:47:02 uavl001 logstash: }
```

Ilustración 43: Captura de los eventos de Windows Server por LogStash

5.3.5 Configuración de LogStash para Internet Information Server

Entrada

De acuerdo a la configuración de nxlog para Windows Server el componente de entrada de LogStash correspondiente deberá utilizar el módulo tcp en el puerto 5515 con formato JSON. Además para identificación posterior se asignará a la variable “type” el valor “windows”. Por lo tanto se agregará lo siguiente en la sección input del archivo de configuración /etc/logstash/logstash.conf:

```
tcp {
 port => 5515
 type => "iis"
 format => 'json'
}
```

Ilustración 44: Captura de la entrada de LogStash para IIS

Filtro

En el componente de filtrado de LogStash se utilizará el módulo grok con el objetivo de normalizar las entradas que provengan del servidor IIS.

```
 }else if [type] == "iis" {
  grok {
   match => ["message", '%{TIMESTAMP_ISO8601:localtime} %{IPORHOST:hostname} %{W
ORD:method} %{URIPATH:page} %{NOTSPACE:querystring} %{NUMBER:port} %{NOTSPACE:us
ername} %{IPORHOST:clienthost} %{NOTSPACE:useragent} %{NOTSPACE:application} %{N
UMBER:response} %{NUMBER:subresponse} %{NUMBER:scstatus} %{NUMBER:time_taken}' ]
  }
```

Ilustración 45: Captura del filtro de LogStash para IIS

Verificación

La validez de la configuración se verificará monitoreando el archivo de log del sistema mediante el comando: *tail –f /var/log/messages*, lo que permite visualizar los registros de salida en formato JSON:

```
Apr 15 18:17:25 uavl001 logstash: "message" => "2015-03-05 20:59:08 10.204.71.227 PO
illa/4.0+(compatible;+MSIE+7.0;+Windows+NT+6.1;+WOW64;+Trident/4.0;+SLCC2;+.NET+CLR+
NET4.0E;+.NET+CLR+1.1.4322) http://uaas001/WebAsistencias/frmReporteDiario.aspx 200
Apr 15 18:17:25 uavl001 logstash: "@version" => "1",
Apr 15 18:17:25 uavl001 logstash: "@timestamp" => "2015-04-15T23:17:24.093Z",
Apr 15 18:17:25 uavl001 logstash: "host" => "10.204.71.227:55279",
Apr 15 18:17:25 uavl001 logstash: "type" => "iis",
Apr 15 18:17:25 uavl001 logstash: "log_timestamp" => "2015-03-05 20:59:08",
Apr 15 18:17:25 uavl001 logstash: "site" => "10.204.71.227",
Apr 15 18:17:25 uavl001 logstash: "method" => "POST",
Apr 15 18:17:25 uavl001 logstash: "page" => "/WebAsistencias/frmReporteDiario.aspx",
Apr 15 18:17:25 uavl001 logstash: "querystring" => "-",
Apr 15 18:17:25 uavl001 logstash: "port" => "80",
Apr 15 18:17:25 uavl001 logstash: "username" => "                    ",
```

Ilustración 46: Captura de los eventos de IIS por LogStash

5.3.6 Configuración de LogStash para RHEL

Entrada

Con el objetivo de aceptar los logs transmitidos por el servicio de logstash-forwarder instalado en el servidor RHEL es necesario agregar una entrada tipo lumberjack en el puerto 5516, indicando además la ubicación del certificado de servidor y su clave para validar la proceso de autenticación del cliente.

```
lumberjack {
 port => 5516
 ssl_certificate => "/etc/logstash/uavl001.crt"
 ssl_key => "/etc/logstash/uavl001.key"
}
```

Ilustración 47: Captura de la entrada de LogStash para RHEL

Filtro

En el componente de filtrado de LogStash se utilizará el módulo grok para cada uno de los tres tipos de log definidos en el recolector de logstash-forwarder con el objetivo de normalizar las entradas:

```
}else if [type] == "rhel" {
 grok {
  match => ["message", '%{SYSLOGTIMESTAMP:localtime} %{IPORHOST:hostname} %{NOT
SPACE:facility}: %{GREEDYDATA:description}' ]
 }
}else if [type] == "apache_error" {
 grok {
  match => ["message", '\[%{APACHETIMESTAMP:localtime}\] \[%{WORD:severity}\] %
{GREEDYDATA:description}' ]
 }
}else if [type] == "apache_log" {
 grok {
  match => ["message", '%{COMBINEDAPACHELOG}' ]
 }
}
```

Ilustración 48: Captura del filtro de LogStash para RHEL

Verificación

La validez de la configuración se verificará monitoreando el archivo de log del sistema mediante el comando: *tail –f /var/log/messages*, lo que permite visualizar los registros de salida en formato JSON:

```
Apr 16 15:26:17 uavl001 logstash: "@version" => "1",
Apr 16 15:26:17 uavl001 logstash: "@timestamp" => "2015-04-16T20:26:16.907Z",
Apr 16 15:26:17 uavl001 logstash: "file" => "/var/log/httpd/access_log",
Apr 16 15:26:17 uavl001 logstash: "host" => "ualm002.                    ",
Apr 16 15:26:17 uavl001 logstash: "offset" => "519228",
Apr 16 15:26:17 uavl001 logstash: "type" => "apache_log",
Apr 16 15:26:17 uavl001 logstash: "clientip" => "10.204.65.114",
Apr 16 15:26:17 uavl001 logstash: "ident" => "-",
Apr 16 15:26:17 uavl001 logstash: "auth" => "-",
Apr 16 15:26:17 uavl001 logstash: "timestamp" => "16/Apr/2015:15:08:57 -0500",
Apr 16 15:26:17 uavl001 logstash: "verb" => "GET",
Apr 16 15:26:17 uavl001 logstash: "request" => "/capacitacion_reportes/js/functi
ons.js",
Apr 16 15:26:17 uavl001 logstash: "httpversion" => "1.1",
Apr 16 15:26:17 uavl001 logstash: "response" => "304",
```

Ilustración 49: Captura de los eventos de RHEL por LogStash

6. Análisis y Monitoreo de Eventos

El anàlisis y monitoreo de eventos constitute la etapa de consumo de la información obtenida mediante la gestión de las fuentes generadoras de logs.
No obstante existen múltiples herramientas que se pueden integrar con la solución elegida para evaluación, en el presente escenario se utilizarán tres:

- Kibana, al formar parte de la solución ELK para la exploración, visualización y creación de cuadros de mando.
- SEC (Simple Event Correlator), para la correlación de eventos.
- Nagios, al ser la herramienta de Monitoreo y Alertas existente en la organización de evaluación,

A continuación se presentarán los objetivos específicos de la evaluación de la solución ELK en la organización seleccionada en relación con las herramientas que se utilizarán para su obtención:

Objetivos Específicos	Kibana	SEC	Nagios
Detectar la presencia o inminencia de errores en dispositivos de red y seguridad, sistemas operativos y aplicaciones.	X	X	X
Detectar los accesos e intentos de acceso no autorizados de administración a los switches.	X		
Detectar los accesos e intentos de acceso no autorizados a la red inalámbrica.	X		
Detectar los accesos e intentos de acceso no autorizados a las sesiones de red Windows de los usuarios internos.	X		
Detectar los accesos e intentos de acceso no autorizados a los servicios locales ofrecidos en la red interna.	X		
Visualizar la distribución del consumo de la conexión de Internet por parte de los usuarios internos.	X		
Visualizar la distribución del consumo de los servicios ofrecidos en la red interna.	X		

6.1 Visualización y Exploración de Eventos mediante Kibana.

La solución ELK incluye además de las aplicaciones LogStash y ElasticSearch a Kibana. Kibana es una herramienta web que permite visualizar y explorar los datos almacenados en una base de datos no relacional ElasticSearch. Adicionalmente ofrece la posibilidad de crear cuadros de mando (dashboards) a partir de consultas almacenadas.

6.1.1 Instalación de Kibana

Para detalles de la instalación y configuración del servidor Kibana por favor consultar el Anexo 6.

6.1.2 Configuración de los índices de ElasticSearch

Los resultados de la recolección, filtrado y normalización de logs realizadas mediante LogStash están centralizadas en una instancia de base de datos no relacional. ElasticSearch organiza los datos recolectados en índices con el formato logstash-YYYY.MM.DD.

La configuración de Kibana requiere que se específique el nombre del índice (se admite el uso de comodines) que hará referencia a los datos almacenados en la base:

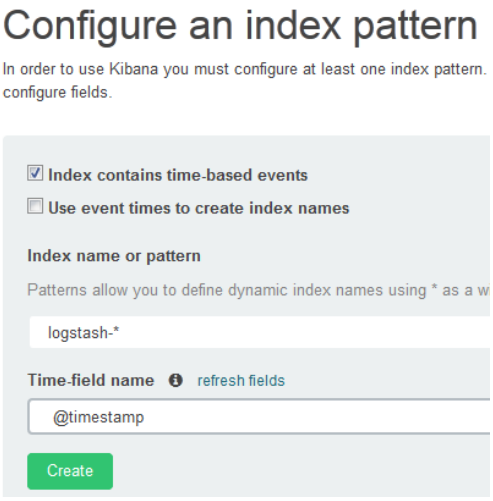


Ilustración 50: Captura de la configuración de índices en Kibana

6.1.3 Exploración de Eventos

A continuación se puede iniciar la exploración de los datos almacenados en ElasticSearch mediante la opción “Discover”:

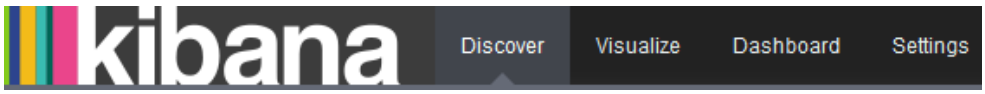


Ilustración 51: Captura de la opción Discover en Kibana

En esta opción se exploran los eventos recolectados y almacenados en un determinado período de tiempo:

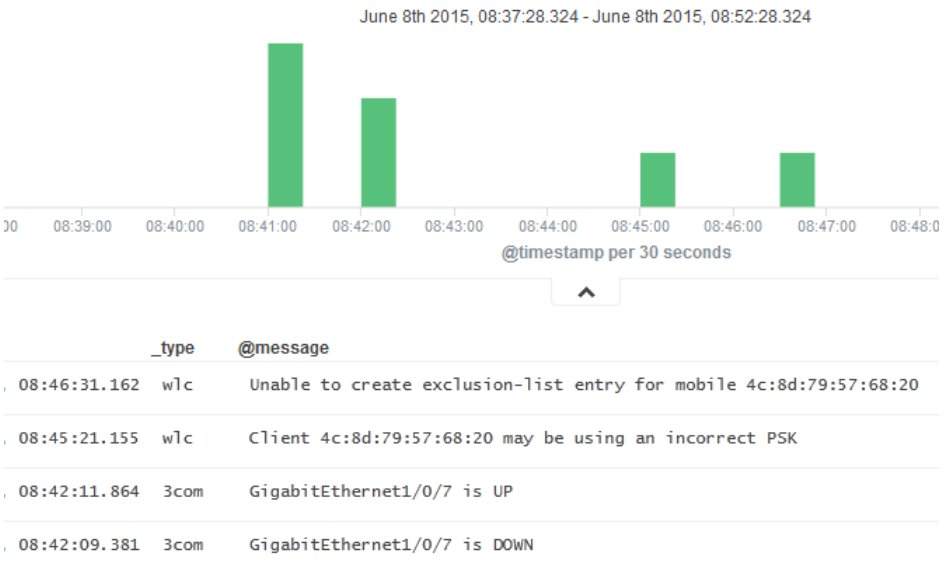


Ilustración 52: Captura de los resultados de la opción Discover en Kibana

El filtro de período se puede seleccionar en la opción “Time Filter” en la zona superior derecha de la pàgina:

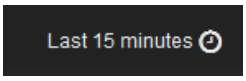


Ilustración 53: Captura del "Time Filter" en Kibana

Dónde se puede seleccionar una opción existente:

Ilustración 54: Captura de la opción existente "Time Filter" en Kibana

O se puede ingresar manualmente el período de tiempo requerido:

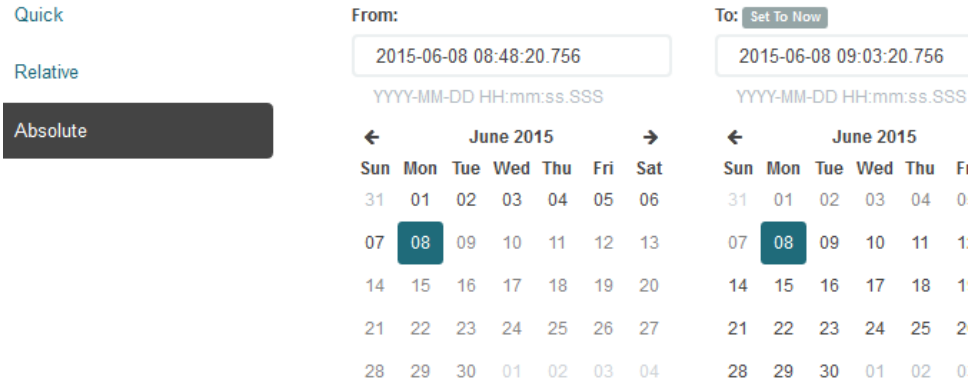


Ilustración 55: Captura de la opción manual "Time Filter" en Kibana

Kibana además permite filtrar eventos de acuerdo a un campo específico existente en el índice registrado:

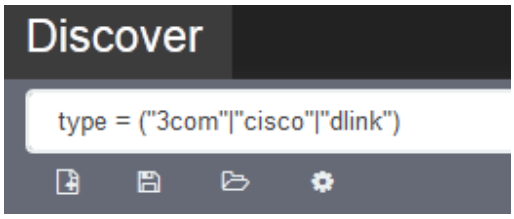


Ilustración 56: Captura del filtro general en Kibana

6.1.4 Visualización de Eventos

Sin embargo, aunque Kibana permite explorar y filtrar los eventos centralizados y almacenados en ElasticSearch mediante la opción "Discover", la verdadera potencialidad se encuentra en la posibilidad de generar resúmenes de los datos mediante funciones matemáticas y estadísticas básicas en la opción "Visualize"

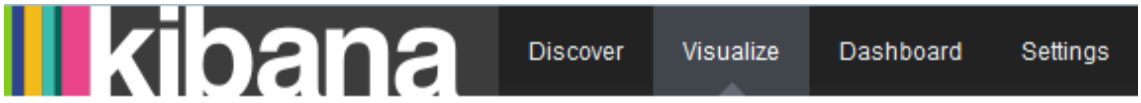


Ilustración 57: Captura de la opción Visualize en Kibana

El conjunto de visualizaciones que se creen de acuerdo a los requerimientos establecidos en los objetivos específicos pasarán luego al cuadro de mando de control.
Las opciones disponibles para crear visualizaciones incluyen las siguientes:

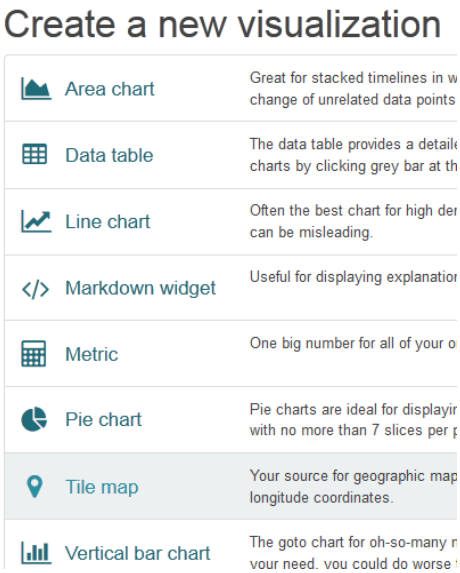


Ilustración 58: Captura de las visualizaciones existentes en Kibana

Debido a la naturaleza de los datos recolectados la mayoría de visualizaciones utilizarán el gráfico “Pie Chart”.
En esta ocasión la visualización representará la proporción de eventos generados por todas las fuentes de logs (índice LogStash) en un determinado período de tiempo:

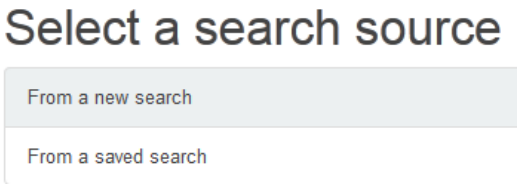


Ilustración 59: Captura de la selección de una búsqueda en Kibana

La métrica que se utilizará es el conteo de todos los registros almacenados, por lo que el filtro es “*”.

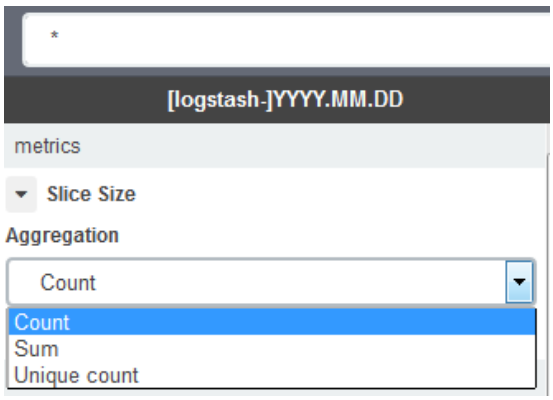


Ilustración 60: Captura de un filtro y una métrica en Kibana

La agregación se realizará en relación a campos en el índice seleccionado, en el presente caso el campo "_type" identifica al tipo de fuente generadora de logs.

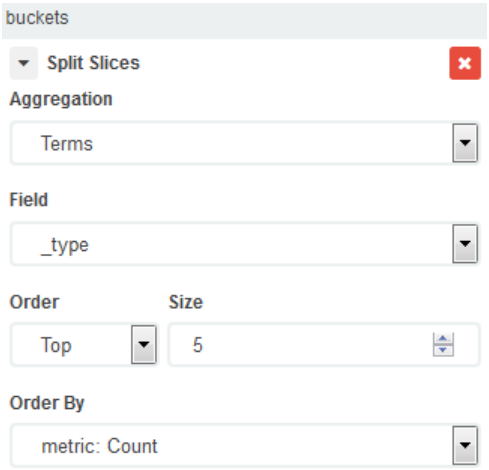


Ilustración 61: Captura de una agregación en Kibana

Estos parámetros seleccionados generan la siguiente visualización:

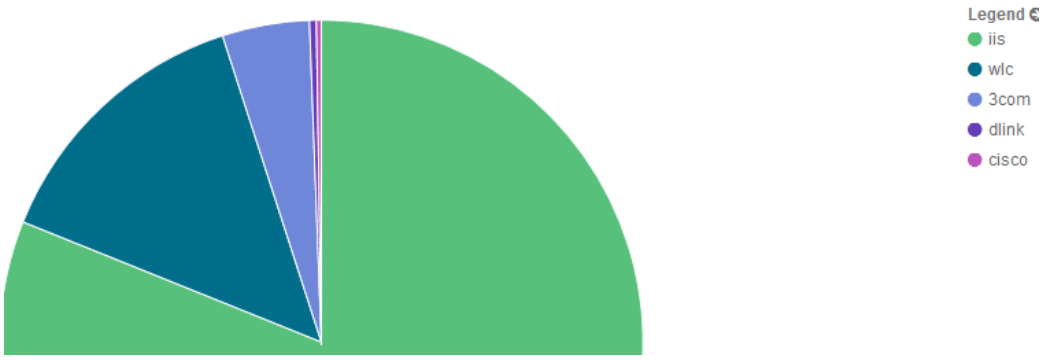


Ilustración 62: Captura de una visualización en Kibana

6.1.4.1 Creación de Cuadros de Mando

Las visualizaciones creadas y almacenadas se pueden organizar en cuadros de mandos para facilitar el acceso y el monitoreo mediante la opción "Dashboard":

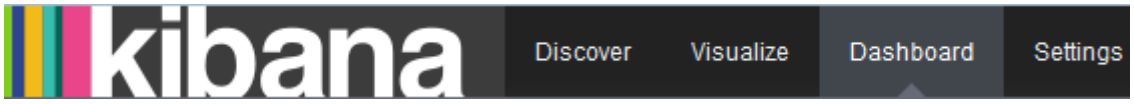


Ilustración 63: Captura de la opción Dashboard en Kibana

Las visualizaciones creadas para satisfacer los objetivos específicos propuestos son las siguientes:

Objetivo: Detectar los accesos e intentos de acceso no autorizados de administración a los switches.

Para cumplir este objetivo se agregará un filtro para los logs de las fuentes de los switches de acuerdo al type:

Cisco

```
 if [mnemonic] == "IPACCESSLOGNP" {
       grok {
                match => ["@message", '%{GREEDYDATA} 0 %{IPORHOST:source_host} %
{GREEDYDATA}']
       }
       mutate {
                replace => [ "@message", "Connection from %{source_host}" ]
                add_tag => [ "switch_connection" ]
       }
 }
```

Ilustración 64: Captura de la modificación en el filtro "cisco" en LogStash

3com

```
       if [mnemonic] == "LOGIN" {
                grok {
                        match => ["@message", '%{WORD}\(%{IPORHOST:source_host}\
) %{GREEDYDATA}' ]
                }
                mutate {
                        replace => [ "@message", "Connection from %{source_host}
" ]
                        add_tag => [ "switch_connection" ]
                }
       }else if [mnemonic] == "fsm_move" {
                drop {}
       }
```

Ilustración 65: Captura de la modificación en el filtro "3com" en LogStash

Dlink

```
if [@message] =~ /Successful login/ {
      grok {
               match => ["@message", '%{GREEDYDATA}: %{IPORHOST:source_host}' ]
      }
      mutate {
               replace => [ "@message", "Connection from %{source_host}" ]
               add_tag => [ "switch_connection" ]
      }
}
```

Ilustración 66: Captura de la modificación en el filtro "dlink" en LogStash

La métrica que se utilizará es el conteo de todos los registros almacenados, con el filtro: "tags in 'switch_connection'".

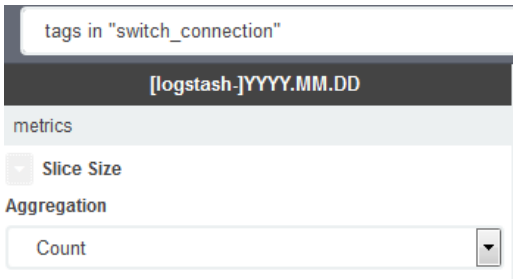


Ilustración 67: Captura del filtro y métrica para la visualización

La agregación se realizará en relación a campos en el índice seleccionado, en el presente caso el campo "source_host" identifica la IP origen de la conexión

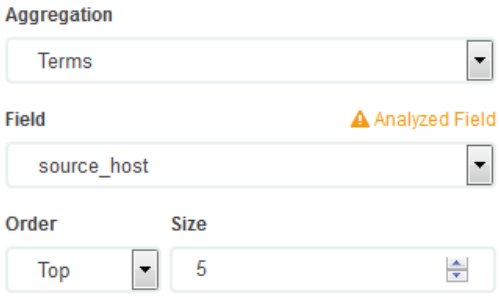


Ilustración 68: Captura de la agregación para la visualización

.Estos parámetros seleccionados generan la siguiente visualización:

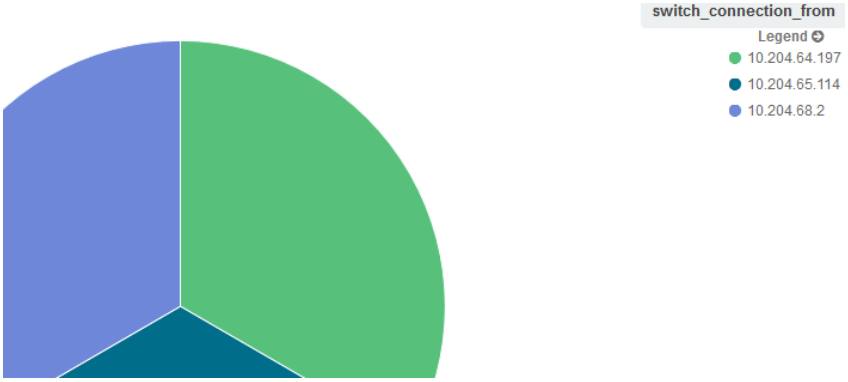


Ilustración 69: Captura de la visualización

Objetivo: Detectar los accesos e intentos de acceso no autorizados a la red inalámbrica.

Descripción	Filtro	Agregación
Contabilizar el número de accesos exitosos a la red inalámbrica con WPA2 con IEEE 802.1x.	type: wlc AND mnemonic: USER_AUTH_PASSED	count(source_host)
Contabilizar el número de accesos no exitosos a la red de invitados con WPA2 con IEEE 802.1x.	type: wlc AND mnemonic: AAA_MAX_RETRY	count(source_host)
Contabilizar el número de accesos no exitosos a la red de invitados con WPA2 PSK.	type: wlc AND mnemonic: PSK_CONFIG_ERR	count(source_host)

Objetivo: Detectar los accesos e intentos de acceso no autorizados a las sesiones de red Windows de los usuarios internos.

Lamentablemente este objetivo no se puede cumplir debido a que la política de seguridad global de la organizacional impide la instalación de software adicional en los servidores controladores de dominio.

Objetivo: Detectar los accesos e intentos de acceso no exitosos a los servicios locales ofrecidos en la red interna.

Descripción	Filtro	Agregación
Detectar los errores y accesos no exitosos a las aplicaciones web en Internet Information Server	type: iis AND (method: POST or method: GET) AND NOT response: 200	count(page)
Detectar las solicitudes de autenticación	type: apache_log AND (verb:	count(clientip)

a las aplicaciones web del servidor Apache Httpd.	POST or verb: GET) and response: 303	

Objetivo: Visualizar la distribución del consumo de los servicios ofrecidos en la red interna.

Descripción	Filtro	Agregación
Verificar la distribución en el uso de las diferentes aplicaciones web en Internet Information Server.	type: iis AND (method: POST or method: GET) AND response: 200	count(page)
Verificar la distribución en el uso de las diferentes aplicaciones web en Apache Httpd.	type: apache_log AND (method: POST or method: GET) AND response: 200	count(request)

Objetivo: Visualizar la distribución del consumo de la conexión de Internet por parte de los usuarios internos.

Para cumplir este objetivo se requiere utilizar el índice “ntopng-*” exportado directamente desde la aplicación ntopng. La métrica que se utilizará es la sumatoria de los bytes transferidos desde el servidor proxy 10.204.71.193 hacia los clientes con el filtro: “IPV4_SRC_ADDR=’10.204.71.193’".

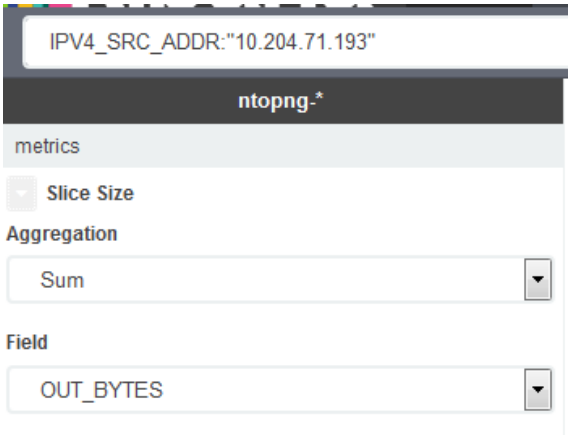


Ilustración 70: Captura del filtro y métrica para la visualización

La agregación se realizará con el campo “IPV4_DST_ADDR” del índice seleccionado, que identifica la IP destino de la conexión

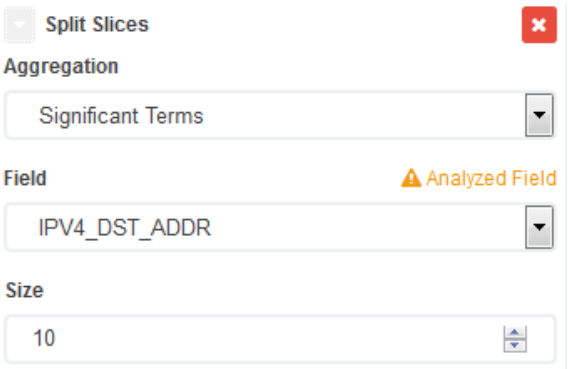


Ilustración 71: Captura de la agregación para la visualización

Estos parámetros seleccionados generan la siguiente visualización:

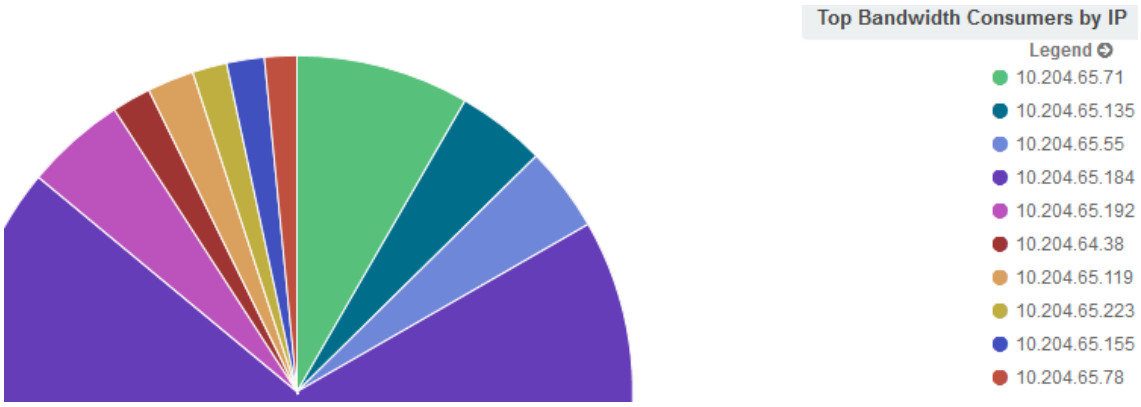


Ilustración 72: Captura de la visualización

Finalmente se organizarán las visualizaciones obtenidas mediante la creación de cuadros de mando de acuerdo a criterios específicos.

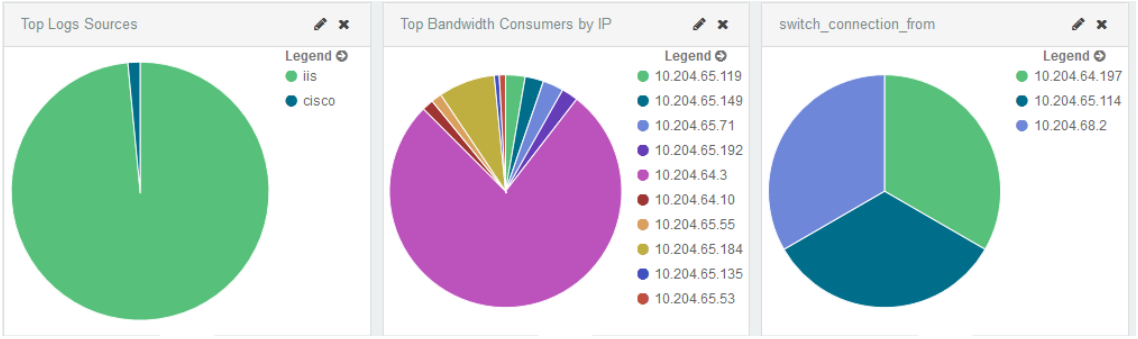


Ilustración 73: Captura del dashboard

7. Conclusiones

El desarrollo de este proyecto de Diseño e Implementación de una solución de gestion centralizada de logs basada en ELK ha generado las siguientes conclusiones agrupadas de acuerdo a su relación con el proyecto:

7.1 Conclusiones con respecto al tema del proyecto

- Se confirma la existencia de una enorme cantidad de información potencialmente reelevante para la seguridad de una organización en los registros de eventos producidos por sus diferentes sistemas. de una organización convierte a la gestión de logs en un mecanismo preventivo y detectivo fundamental en la implantación de una política de seguridad.
- Inicialmente se indicó que el principal inconveniente al momento de implantar un solución para la gestión de logs era la ausencia de un modelo de referencia estándar; sin embargo, durante el desarrollo de este proyecto se constató que el elemento fundamental, y por consiguiente más problemático, se encuentra en la definición de una política de gestión de logs. Una política que además de responder a las preguntas ¿Por qué?, ¿Qué?, ¿Cómo? Y ¿Dónde?; sirva como estrategia para el diseño e implantación de una solución de gestión de logs.
- La solución ELK (ElasticSearch, LogStash y Kibana) cumple con los requerimientos exigidos por un organización de tamaño medio (entre 100 y 1000 usuarios) con una infraestructura estándar. Con relación a los resultados de este proyecto un aspecto importante a remarcar es la funcionalidad de filtrar y procesar los logs recolectados desde las fuentes generadoras proporcionada por LogStash. Esta característica permite descartar una enorme cantidad de información irrelevante con relación a los objetivos específicos planteados en la política de gestión de logs.
- En la organización donde se implanto la solución se cumplieron dos objetivos: en primer lugar se dispone de una interfaz de consulta gráfica que permite verificar el cumplimiento de variables de seguridad definidas como prioritarias, además el sistema de gestión de logs mediante la solución ELK se transformó en una entrada que alimenta el sistema de monitoreo y alertas Nagios.

7.2 Conclusiones con respecto al desarrollo del proyecto

- El cumplimiento de los objetivos planteados inicialmente se puede concluir que la mayoría se cumplieron con resultados satisfactorios, de hecho el reporte presentado en el cuadro de mando de Kibana se utiliza actualmente como herramienta de monitoreo para verificar el estado de los sistemas analizados. Con relación a los pocos objetivos que no pudieron cumplirse, como por ejemplo la visualización de los accesos exitosos y no exitosos a las sesiones de trabajo en el dominio Windows, los motivos son de naturaleza administrativa antes que técnicos u operativos. Como muestra, en el caso presentado anteriormente, el motivo fue que los controladores de dominio se encuentran bajo una política se seguridad que impide la instalación de software externo al sistema, en este caso el agente de transporte de logs.

7.3 Conclusiones con respecto a derivaciones del proyecto

- Uno de los problemas identificados durante el desarrollo del proyecto fue es que la cantidad de tráfico generado por la transmisión de logs en la centralización de las fuentes podría saturar enlaces WAN de limitada capacidad, situación que crearía un escenario de competencia con el tráfico de usuario y podría afectar el rendimiento de las aplicaciones. Este inconveniente, común a todas las soluciones de gestión de logs, se podría resolver implementando en las localidades remotas dónde se produzca un intermediario (broker) con la capacidad de almacenar temporalmente la información recolectada desde la fuentes, y transmitirla fuera de línea hacia el servidor central de almacenamiento.
- Otro aspecto que podría suscitar un desarrollo posterior es el concerniente a la verificación de la escalabilidad de la solución ELK, especialmente en el asunto del almacenamiento. De acuerdo a su documentación, ElasticSearch ofrece la integración simple y directa de múltiples nodos al ***cluster*** definido en la configuración de las instancias operativas. Esta facilidad permite establecer dinámicamente la escalabilidad de la solución de acuerdo al rendimiento del sistema.
- En la organización se ha planteado la posibilidad de utilizar ELK como solución para la generación de un reporte regional de cumplimiento a nivel de las Américas. El reporte presenta información recolectada de múltiples fuentes respecto al estado actual de seguridad de dispositivos, sistemas y aplicaciones. Actualmente el reporte se realiza manualmente en una hoja electrónica y se utiliza como punto de partida de la auditoría de seguridad interna.

8. Glosario

Almacenamiento de Logs: Consiste en el empleo de un medio de almacenamiento permanente (archivos, bases de datos, etc.) para alojar logs generados y recolectados. Incluye la ejecución de tareas de rotación, archivado, comprensión, reducción, conversión, normalización y chequeo de integridad. [3, 3-3]

Análisis de Logs: Se refiere al anàlisis del contenido de los logs con el objetivo de interpretarlo y obtener información reelevante respecto a la administración de recursos, detección de intrusiones, resolución de problemas, anàlisis forense o auditorías dentro de la organización. Incluye la correlación de eventos, la visualización y exploración, y la generación de reportes. [3, 3-4]

Ciclo de Vida de un Log: Hace referencia a la serie de etapas de existencia del log en la organización desde su generación hasta su definitivo descarte y eliminación.

Eliminación de Logs: Hace referencia a la eliminación de las entradas de logs almacenadas correspondientes a un criterio, que generalmente incluye una fecha y una hora determinadas.

Evento: “Un evento es una singular ocurrencia dentro de un ambiente, que involucra usualmente un intento de cambio de estado. Un evento incluye una noción de tiempo, la ocurrencia, y una descripción pertinente al evento o al ambiente que pueda ayudar a explicar o entender las causas o efectos del evento.” [1, 30]

Fuente Generadora de Logs: Se refiere a todo dispositivo, sistema o aplicación que esté en la capacidad de registrar la ocurrencia de eventos.

Log: De acuerdo al NIST: “Un log es un registro de los eventos que ocurren dentro de los sistemas y redes de una organización”. [3, ES-1]

SIEM: Según [3, 3-9] un Security Information and Event Management (SIEM) es una solución de software que incluye recolección, visualización, almacenamiento y anàlisis de logs.

Transporte y Recolección de Logs: Incluyen los diversos mecanismos disponibles para mover los logs desde la fuente generadora hacia una ubicación diferente, generalmente centralizada.

9. Bibliografía

[1] Anton Chuvakin, Kevin J. Schmidt, Logging and Log Management: The Authoritative Guide to Understanding the Concepts Surrounding Logging and Log Management, Syngress, 1era. Edición, Diciembre 13, 2012.

[2] James Turnbull, The Logstash Book, Log management made easy, Amazon Digital Services, Inc., 1era. Edición, Diciembre 20, 2013.

[3] http://csrc.nist.gov/publications/nistpubs/800-92/SP800-92.pdf, NIST 800-92 “Guide to Computer Security Log Management”, Accedido por última vez el 12 de Junio del 2015.

10. Anexos

10.1 Anexo 1: Instalación del Servidor LogStash

La solución se implementará en una plataforma de virtualización Vmware ESXi 5.5 instalada en un servidor HP Proliant DL380 G5 con Procesador Quad-Core Intel Xeon, 8 Gigabytes en RAM y 500 Gigabytes en disco duro.
Por motivos de rendimiento y eficiencia se ha preferido separar los componentes de entrada y procesamiento de logs de los componentes de almacenamiento, exploración y análisis mediante la definición de dos servidores virtuales independientes basados en Red Hat Enterprise Linux versión 7.0.
El servidor virtual para Logstash se creó con las siguientes características:

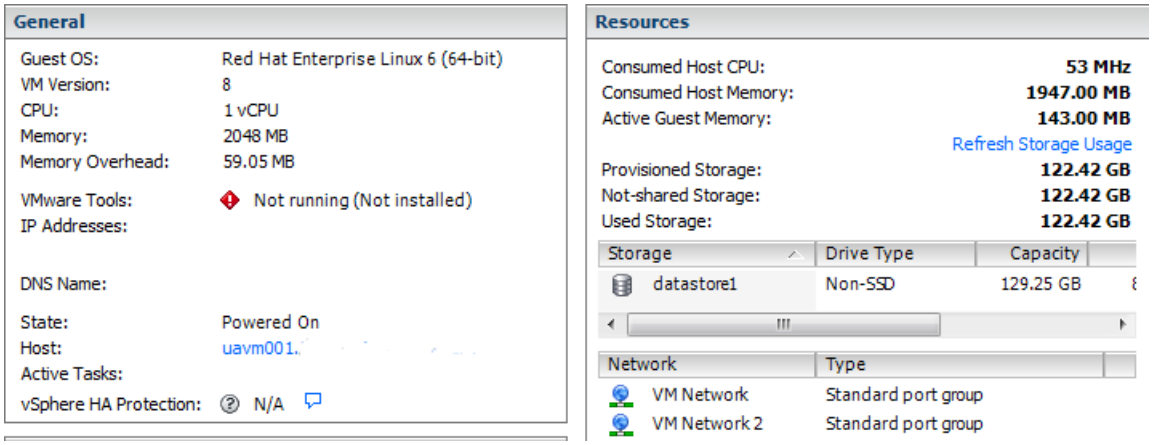


Ilustración 74: Captura de la configuración del servidor virtual LogStash

El procedimiento de instalación del sistema operativo RHEL 7.0 es el estándar que se puede encontrar en diversos sitios en Internet. Por ejemplo en http://www.tecmint.com/redhat-enterprise-linux-7-installation/ (Se seleccionó la opción “Instalación Mínima”)
El nombre del servidor es uavl001 (debido particularmente a la nomenclatura estándar vigente al interior de la organización).

```
[root@uavl001 ~]# uname -a
Linux uavl001.                          3.10.0-123.el7.x86_64 #1 SMP Mon May 5 11:1
6:57 EDT 2014 x86_64 x86_64 x86_64 GNU/Linux
```

Ilustración 75: Captura del resultado del comando uname -a en el servidor LogStash

La dirección IP del servidor es 10.204.71.246 (de igual manera siguiente el esquema de direccionamiento IP vigente en la organización).

```
[root@uavl001 ~]# ifconfig
ens192: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.204.71.246  netmask 255.255.255.128  broadcast 10.204.71.255
        inet6 fe80::20c:29ff:fef6:d473  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:f6:d4:73  txqueuelen 1000  (Ethernet)
        RX packets 118558516  bytes 15858350978 (14.7 GiB)
        RX errors 0  dropped 4324213  overruns 0  frame 0
        TX packets 125627516  bytes 46751321703 (43.5 GiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Ilustración 76: Captura del resultado del comando ifconfig en el servidor LogStash

El esquema de particionamiento incluye las siguientes particiones:

- /boot con 497 Megabytes.
- / con 21 Gigabytes.
- /var con 98 Gigabytes.

```
[root@uavl001 ~]# df -Th
Filesystem                    Type      Size  Used Avail Use% Mounted on
/dev/mapper/rhel_uavl001-root xfs        21G  2.5G   19G  12% /
devtmpfs                      devtmpfs  915M     0  915M   0% /dev
tmpfs                         tmpfs     921M     0  921M   0% /dev/shm
tmpfs                         tmpfs     921M   65M  857M   8% /run
tmpfs                         tmpfs     921M     0  921M   0% /sys/fs/cgroup
/dev/sda1                     xfs       497M  135M  363M  28% /boot
/dev/mapper/rhel_uavl001-var  xfs        98G  1.4G   96G   2% /var
```

Ilustración 77: Captura del resultado del comando df -Th en el servidor LogStash

Prerrequisitos a la instalación de Logstash

Previo a la instalación de la aplicación Logstash es necesario disponer de java, por lo que se necesita configurar el sistema de gestión de paquetes rpm yum.
En el caso particular de RHEL es necesario registrar el sistema en la Red Hat Network (RHN) para acceder a los repositorios oficiales: rhel-7-server y rhel-7-server-rh-common.
Necesarios estos dos últimos para instalar y resolver dependencias para
Adicionalmente se agregará el repositorio EPEL (Extra Packages for Enterprise Linux) para disponer de varios paquetes necesarios para la puesta en marcha de la solución.
La instalación se realiza mediante el comando: *yum install epel-release*.
Para encontrar el paquete de instalación es necesario que el repositorio oficial rhel-7-server-extras se encuentre activo.

```
[root@uavl001 yum.repos.d]# yum install epel-release
Loaded plugins: product-id, subscription-manager
Package epel-release-7-5.noarch already installed and latest version
```

Ilustración 78: Captura del resultado del comando yum install epel-release en el servidor LogStash

Luego de la instalación se puede verificar la creación del repositorio:

```
[root@uavl001 ~]# cd /etc/yum.repos.d/
[root@uavl001 yum.repos.d]# more epel.repo
[epel]
name=Extra Packages for Enterprise Linux 7 - $basearch
#baseurl=http://download.fedoraproject.org/pub/epel/7/$basearch
mirrorlist=https://mirrors.fedoraproject.org/metalink?repo=epel-7&arch=$basearch
failovermethod=priority
enabled=1
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-EPEL-7
```

Ilustración 79: Captura del contenido del archivo epel.repo en el servidor LogStash

Además es necesario tener en cuenta que los paquetes del repositorio EPEL mantienen dependencias que se encuentran en el repositorio oficial rhel-7-server-optional.

Ahora si se procederá a la instalación de la versión “open source” de java openjdk:

- En primer lugar se busca el paquete mediante el comando: *yum search openjdk*

  Que retorna el siguiente resultado:

```
java-1.7.0-openjdk.x86_64 : OpenJDK Runtime Environment
```

Ilustración 80: Captura de la versión de openjdk en el servidor LogStash

- Con el nombre del paquete se continúa a la instalación: yum install java-1.7.0-openjdk.x86_64
- Y finalmente a verificar la versión de java instalada:

```
[root@uavl001 yum.repos.d]# java -version
java version "1.7.0_75"
OpenJDK Runtime Environment (rhel-2.5.4.7.el7_1-x86_64 u75-b13)
OpenJDK 64-Bit Server VM (build 24.75-b04, mixed mode)
```

Ilustración 81: Captura del resultado del comando java -version en el servidor LogStash

Instalación de Logstash

Para el desarrollo del proyecto se ha decidido trabajar con la versión 1.4.2 de LogStash que se encuentra disponible en formato tar.gz.
Por lo tanto se procede a descargar el archivo con el comando: *curl -O https://download.elasticsearch.org/logstash/logstash/logstash-1.4.2.tar.gz*

```
[root@uavl001 tmp]# curl -O https://download.elasticsearch.org/logstash/logstash
/logstash-1.4.2.tar.gz
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 81.6M  100 81.6M    0     0   384k      0  0:03:37  0:03:37 --:--:--  504k
```

Ilustración 82: Captura de la descarga del archivo de instalación LogStash

La carpeta de instalación será /opt:

- Se procede a desempaquetar el archivo: *tar –xvzf logstash-1.4.2.tar.gz –C /opt*
- Y a crear un enlace: *ln –s /opt/logstash-1.4.2 /opt/logstash*

```
[root@uavl001 tmp]# ls /opt
logstash  logstash-1.4.2
[root@uavl001 tmp]# ls /opt/logstash
bin  lib  LICENSE  locales  patterns  README.md  spec  vendor
```

Ilustración 83: Captura del contenido del archivo de instalación LogStash

Finalmente para verificar la instalación se ejecutará la aplicación con la consola como entrada/salida:

```
[root@uavl001 tmp]# /opt/logstash/bin/logstash -e 'input { stdin {} } output { s
tdout {} }'
HOLA MUNDO
2015-04-11T23:59:34.968+0000 uavl001.                    HOLA MUNDO
```

Ilustración 84: Captura de una prueba de verificación de LogStash

A partir de los resultados obtenidos se puede constatar que la instalación de LogStash funciona correctamente, por lo que se puede proceder a la automatización de la ejecución del servicio

Automatización del servicio LogStash

La automatización de la ejecución del servicio LogStash se realizará mediante el sistema de gestión de servicios systemd. Para realizar esta tarea se utilizará el ejemplo que se encuentra en https://github.com/dkowis/smgl-systemdb/blob/master/logstash.service

- El primer paso es crear el archivo que identificará al servicio logstash en el directorio /usr/lib/systemd/system/:
  - *touch /usr/lib/systemd/system/logstash.service*.
- A continuación se agregará al archivo creado el siguiente contenido:

```
[Unit]
Description=Starts and stops a single logstash instance on this system
Documentation=http://www.logstash.net
BindTo=network.target
[Service]
User=root
Group=root
Type=simple
PIDFile=/var/run/logstash.pid
ExecStart=/opt/logstash/bin/logstash    -l    /var/log/logstash.log    -f
/etc/logstash/l
ogstash.conf
[Install]
WantedBy=multi-user.target
```

```
[root@uavl001 logstash]# more /usr/lib/systemd/system/logstash.service
[Unit]
Description=Starts and stops a single logstash instance on this system
Documentation=http://www.logstash.net
BindTo=network.target
[Service]
User=root
Group=root
Type=simple
PIDFile=/var/run/logstash.pid
ExecStart=/opt/logstash/bin/logstash -l /var/log/logstash.log -f /etc/logstash/l
ogstash.conf
[Install]
WantedBy=multi-user.target
```

Ilustración 85: Captura del script de inicialización de LogStash

* Inicialmente se configuró el servicio para ejecutarlo con el usuario logstash y el grupo logstash; sin embargo, debido a que es necesario utilizar en la configuración de LogStash puertos menores a 1024 se cambió el usuario y grupo a root (Solo el root puede abrir puertos menores a 1024).

- Luego es necesario realizar algunas tareas adicionales relacionadas al contenido del archivo:
  - Crear el directorio y el archivo para almacenar la configuración de logstash(parámetro –f):
    mkdir /etc/logstash
    touch /etc/logstash/logstash.conf
  - Con el único objetivo de probar el servicio se agregará las siguientes líneas al archivo de configuración /etc/logstash/logstash.conf

```
input {
        stdin { }
}
output {
        stdout {
                codec => rubydebug{
                }
        }
}
```

- Finalmente se prueba el servicio mediante los siguientes comandos:
  - *systemctl daemon-reload*
  - *systemctl enable logstash*
  - *systemctl start logstash*
  - *systemctl status logstash*

```
[root@uavl001 logstash]# systemctl daemon-reload
[root@uavl001 logstash]# systemctl start logstash
[root@uavl001 logstash]# systemctl status logstash
logstash.service - Starts and stops a single logstash instance on this system
   Loaded: loaded (/usr/lib/systemd/system/logstash.service; enabled)
   Active: active (running) since Sat 2015-04-11 19:36:06 ECT; 4s ago
     Docs: http://www.logstash.net
 Main PID: 21257 (java)
   CGroup: /system.slice/logstash.service
           └─21257 java -Xmx500m -XX:+UseParNewGC -XX:+UseConcMarkSweepGC -Dj...

Apr 11 19:36:06 uavl001.                    systemd[1]: Starting Starts an...
Apr 11 19:36:06 uavl001.                    systemd[1]: Started Starts and...
Hint: Some lines were ellipsized, use -l to show in full.
```

Ilustración 86: Captura del mecanismo de gestión de servicios para LogStash

10.2 Anexo 2: Instalación del Servidor ElasticSearch

El servidor virtual para ElasticSearch se creó con las siguientes características:

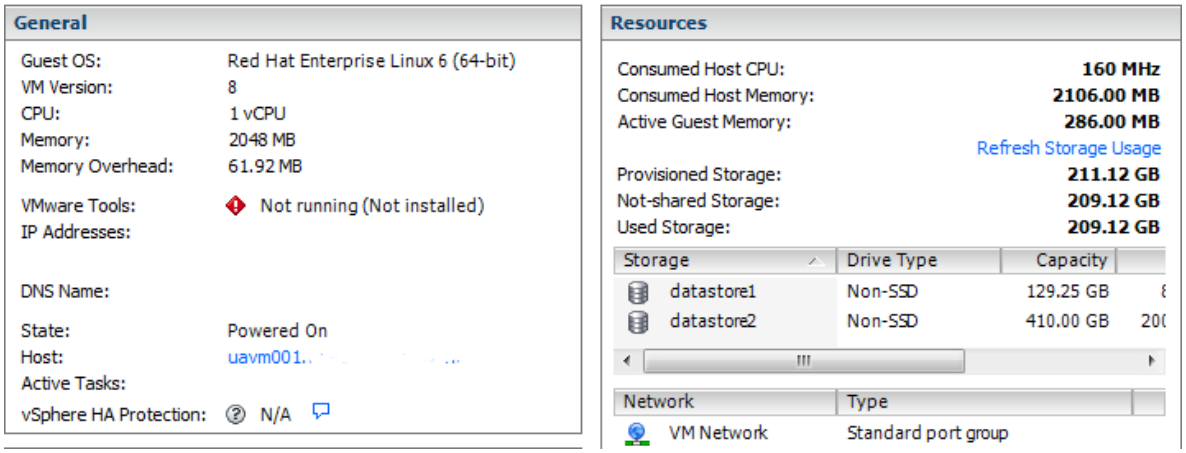


Ilustración 87: Captura de la configuración del servidor virtual ElasticSearch

El procedimiento de instalación del sistema operativo RHEL 7.0 es el estándar que se puede encontrar en diversos sitios en Internet. Por ejemplo en http://www.tecmint.com/redhat-enterprise-linux-7-installation/ (Se seleccionó la opción “Instalación Mínima”)
El nombre del servidor es uavl002 (debido particularmente a la nomenclatura estándar vigente al interior de la organización).

```
[root@uavl002 ~]# uname -a
Linux uavl002.                        3.10.0-123.el7.x86_64 #1 SMP Mon May 5 11:1
6:57 EDT 2014 x86_64 x86_64 x86_64 GNU/Linux
```

Ilustración 88: Captura del resultado del comando uname -a en el servidor ElasticSearch

La dirección IP del servidor es 10.204.71.247 (de igual manera siguiente el esquema de direccionamiento IP vigente en la organización).

```
2: ens192: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP
qlen 1000
    link/ether 00:0c:29:98:df:a8 brd ff:ff:ff:ff:ff:ff
    inet 10.204.71.247/25 brd 10.204.71.255 scope global ens192
       valid_lft forever preferred_lft forever
    inet6 fe80::20c:29ff:fe98:dfa8/64 scope link
       valid_lft forever preferred_lft forever
```

Ilustración 89: Captura del resultado del comando ip addr show en el servidor ElasticSearch

El esquema de particionamiento incluye las siguientes particiones:

- /boot con 197 Megabytes.
- / con 11 Gigabytes.
- /var con 197 Gigabytes.

```
[root@uavl002 ~]# df -Th
Filesystem                    Type      Size  Used Avail Use% Mounted on
/dev/mapper/rhel_uavl002-root xfs        11G  1.3G  9.5G  13% /
devtmpfs                      devtmpfs  915M     0  915M   0% /dev
tmpfs                         tmpfs     921M     0  921M   0% /dev/shm
tmpfs                         tmpfs     921M   17M  905M   2% /run
tmpfs                         tmpfs     921M     0  921M   0% /sys/fs/cgroup
/dev/mapper/rhel_uavl002-var  xfs       197G  341M  196G   1% /var
/dev/sda1                     xfs       197M  120M   78M  61% /boot
```

Ilustración 90: Captura del resultado del comando df -Th en el servidor ElasticSearch

Prerrequisitos previos a la instalación de ElasticSearch

Previo a la instalación de la aplicación ElasticSearch es necesario disponer de java, por lo que se necesita configurar el sistema de gestión de paquetes rpm yum.
En el caso particular de RHEL es necesario registrar el sistema en la Red Hat Network (RHN) para acceder a los repositorios oficiales: rhel-7-server y rhel-7-server-rh-common.
Necesarios estos dos últimos para instalar y resolver dependencias para
Adicionalmente se agregará el repositorio EPEL (Extra Packages for Enterprise Linux) para disponer de varios paquetes necesarios para la puesta en marcha de la solución.
La instalación se realiza mediante el comando: *yum install epel-release*.
Para encontrar el paquete de instalación es necesario que el repositorio oficial rhel-7-server-extras se encuentre activo.

```
[root@uavl002 ~]# yum install epel-release
Loaded plugins: product-id, subscription-manager
Package epel-release-7-5.noarch already installed and latest version
```

Ilustración 91: Captura del resultado del comando yum install epel-release en el servidor ElasticSearch

Luego de la instalación se puede verificar la creación del repositorio:

```
[root@uavl002 ~]# cd /etc/yum.repos.d/
[root@uavl002 yum.repos.d]# more epel.repo
[epel]
name=Extra Packages for Enterprise Linux 7 - $basearch
#baseurl=http://download.fedoraproject.org/pub/epel/7/$basearch
mirrorlist=https://mirrors.fedoraproject.org/metalink?repo=epel-7&arch=$basearch
failovermethod=priority
enabled=1
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-EPEL-7
```

Ilustración 92: Captura del contenido del archivo epel.repo en el servidor ElasticSearch

Además es necesario tener en cuenta que los paquetes del repositorio EPEL mantienen dependencias que se encuentran en el repositorio oficial rhel-7-server-optional.

Ahora si se procederá a la instalación de la versión "open source" de java openjdk:

- En primer lugar se busca el paquete mediante el comando: *yum search openjdk*

  Que retorna el siguiente resultado:

```
java-1.7.0-openjdk.x86_64 : OpenJDK Runtime Environment
```

Ilustración 93: Captura de la versión de openjdk en el servidor ElascticSearch

- Con el nombre del paquete se continúa a la instalación: yum install java-1.7.0-openjdk.x86_64
- Y finalmente a verificar la versión de java instalada:

```
[root@uavl002 yum.repos.d]# java -version
java version "1.7.0_75"
OpenJDK Runtime Environment (rhel-2.5.4.7.el7_1-x86_64 u75-b13)
OpenJDK 64-Bit Server VM (build 24.75-b04, mixed mode)
```

Ilustración 94: Captura del resultado del comando java -version en el servidor ElasticSearch

Instalación de ElasticSearch

Con el objetivo de automatizar la instalación y mantenimiento de ElasticSearch se utilizarán los repositorios mantenidos por http://www.elasticsearch.org para lo que se realizarán las siguientes tareas:

- Descargar e instalar la clave GPG:
  - *rpm --import https://packages.elasticsearch.org/GPG-KEY-elasticsearch*

```
[root@uavl002 yum.repos.d]# rpm --import https://packages.elasticsearch.org/GPG-KEY-elasticsearch
```

Ilustración 95: Captura de la importación de la llave del repositorio

- Crear manualmente el repositorio elasticsearch.repo en el directorio /etc/yum.repos.d/
  - touch /etc/yum.repos.d/elasticsearch.repo
- Agregar el siguiente contenido en el archivo elasticsearch.repo

```
[elasticsearch-1.5]
name=Elasticsearch repository for 1.5.x packages
baseurl=http://packages.elasticsearch.org/elasticsearch/1.5/centos
gpgcheck=1
gpgkey=http://packages.elasticsearch.org/GPG-KEY-elasticsearch
enabled=1
```

- A continuación se procede a instalar el paquete elasticsearch

```
[root@uavl002 yum.repos.d]# yum install elasticsearch
Loaded plugins: product-id, subscription-manager
Package elasticsearch-1.5.1-1.noarch already installed and latest version
```

Ilustración 96: Captura de la instalación del paquete ElasticSearch

- Luego verificamos la instalación de ElasticSearch

```
[root@uavl002 yum.repos.d]# /usr/share/elasticsearch/bin/elasticsearch -v
Version: 1.5.1, Build: 5e38401/2015-04-09T13:41:35Z, JVM: 1.7.0_75
```

Ilustración 97: Captura de la verificación del funcionamiento de ElasticSearch

Configuración Básica de ElasticSearch

A menos que se desee aplicar una configuración específica que se ajuste a las características del escenario con el objetivo de optimizar su ejecución (y se disponga del conocimiento necesario para hacerlo) el servicio ElasticSearch se puede ejecutar sin ninguna modificación. En esta ocasión se modificarán dos parámetros de identificación en el archivo de configuración /etc/elasticsearch/elasticsearch.yml:

- `cluster.name: logstash`
- `node.name: "UOC"`

```
################################### Cluster

# Cluster name identifies your cluster for a
# multiple clusters on the same network, mak
#
#cluster.name: elasticsearch
cluster.name: logstash


#################################### Node ##

# Node names are generated dynamically on st
# from configuring them manually. You can ti
#
#node.name: "Franz Kafka"
node.name: "UOC"
```

Ilustración 98: Captura de la configuración básica de ElasticSearch

En el paquete se incluye el script de inicialización elasticsearch.service para automatizar la gestión del servicio ElasticSearch, para activarlo se utilizan los siguientes comandos:

- *systemctl daemon-reload*
- *systemctl enable elasticsearch*
- *systemctl start elasticsearch*
- *systemctl status elasticsearch*

Verificación del funcionamiento de ElasticSearch

Finalmente se verifica que el servicio de ElasticSearch se encuentre funcionando correctamente mediante el comando

```
[root@uavl002 yum.repos.d]# curl --noproxy 10.204.71.247 http://10.204.71.247:92
00
{
  "status" : 200,
  "name" : "UOC",
  "cluster_name" : "logstash",
  "version" : {
    "number" : "1.5.1",
    "build_hash" : "5e38401bc4e4388537a615569ac60925788e1cf4",
    "build_timestamp" : "2015-04-09T13:41:35Z",
    "build_snapshot" : false,
    "lucene_version" : "4.10.4"
  },
  "tagline" : "You Know, for Search"
}
```

Ilustración 99: Captura de prueba de funcionamiento de ElasticSearch

10.3 Anexo 3: Instalación de logstash-forwarder

El primer paso es descargar el software desde https://www.elastic.co/downloads/logstash. Para arquitecturas de 32 bits en sistemas con kernel Linux existe la disponibilidad en binario o en código fuente, en esta ocasión se utilizará el binario.

```
[root@ualm002 ~]# curl -O https://download.elastic.co/logstash-forwarder/binarie
s/logstash-forwarder_linux_386
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 4724k  100 4724k    0     0   287k      0  0:00:16  0:00:16 --:--:--  195k
```

Ilustración 100: Captura de la descarga del archivo de instalación logstash-forwarder

A continuación se creará el ambiente de ejecución de logstash-forwarder:

- Se copiará el archivo descargado en /usr/bin y se creará un acceso directo:
  - cp logstash-forwarder_linux_386 /usr/bin
  - ln –s /usr/bin/ logstash-forwarder_linux_386 /usr/bin/logstash-forwarder
- Se creará un directorio de configuración y el correspondiente archivo:
  - mkdir /etc/logstash-forwarder
  - touch /etc/logstash-forwarder/config.json

Debido a que logstash-forwarder requiere autenticar mediante certificados al servidor LogStash al que redirecciona los logs recolectados se creará en el servidor uavl001 (LogStash Server) un certificado RSA con su correspondiente llave:

```
[root@uavl001 ~]# cd /etc/logstash/
[root@uavl001 logstash]# openssl req -x509  -batch -nodes -newkey rsa:2048 -keyo
ut uavl001.key -out uavl001.crt -subj /CN=uavl001
Generating a 2048 bit RSA private key
.........................+++
.....................................................................................
.............+++
writing new private key to 'uavl001.key'
-----
[root@uavl001 logstash]# ls
b.conf  logstash.conf  uavl001.crt  uavl001.key
```

Ilustración 101: Captura de la generación del certificado para logstash-forwarder

En el servidor LogStash, en el directorio /etc/logstash se almacena el certificado uavl001.crt y la llave uavl001.key. El certificado es necesario exportarlo al servidor que ejecuta logstash-forwarder:

```
[root@ualm002 logstash-forwarder]# scp root@10.204.71.246:/etc/logstash/uavl001.
crt .
root@10.204.71.246's password:
uavl001.crt                                   100% 1090     1.1KB/s   00:00
```

Ilustración 102: Captura de la copia del certificado para logstash-forwarder

A continuación se procederá a agregar en el archivo de configuración config.json la información correspondiente al host y puerto del servidor LogStash, y al certificado de autenticación:

```
"network": {
  # A list of downstream servers listening for our
  # logstash-forwarder will pick one at random and
  # the selected one appears to be dead or unrespo
  "servers": [ "uavl001:5516" ],

  # The path to your trusted ssl CA file. This is
  # to authenticate your downstream server.
  "ssl ca": "/etc/logstash-forwarder/uavl001.crt",

  # Network timeout in seconds. This is most impor
  # logstash-forwarder determining whether to stop
  # acknowledgement from the downstream server. If
  # logstash-forwarder will assume the connection
  # will connect to a server chosen at random from
  "timeout": 15
},
```

Ilustración 103: Captura de la configuración de certificados en logstash-forwarder

* En el servidor LogStash en la entrada lumberjack correspondiente a logstash-forwarder será necesario especificar el certificado y la clave para la autenticación.

Finalmente, para automatizar la ejecución del proceso, se procederá a crear un script de inicialización en base al script desarrollado en https://github.com/jamtur01/logstashbook-code/blob/master/code/4/logstash_forwarder_redhat_init.

- Se crea el archivo logstash_forwarder en el directorio /etc/init.d:
  - touch /etc/init.d/logstash_forwarder
- Se otorga permisos de ejecución sobre el archivo:
  - chmod 0755 /etc/init.d/ logstash_forwarder
- Se agrega el siguiente contenido al archivo:

```
#!/bin/bash
# From The Logstash Book
#     The     original     of     this     file     can     be     found     at:
http://logstashbook.com/code/index.html
#
# logstash-forwarder Start/Stop logstash-forwarder
#
# chkconfig: 345 99 99
# description: logstash-forwarder
# processname: logstash-forwarder

LOGSTASH_FORWARDER_BIN="/usr/bin/logstash-forwarder"
LOGSTASH_FORWARDER_ETC="/etc/logstash-forwarder/config.json"
```

```
find_logstash_forwarder_process () {
    PIDTEMP=`pgrep -f ${LOGSTASH_FORWARDER_BIN}`
    # Pid not found
    if [ -z "$PIDTEMP" ]; then
        PID=-1
    else
        PID=$PIDTEMP
    fi
}

start () {
    find_logstash_forwarder_process
    if [ "$PID" -ne "-1" ]; then
        echo "logstash already running: $PID"
        exit 0
    fi

    nohup ${LOGSTASH_FORWARDER_BIN} -config ${LOGSTASH_FORWARDER_ETC} &
}

stop () {
  pkill -f ${LOGSTASH_FORWARDER_BIN}
}

case $1 in
start)
        start
        ;;
stop)
        stop
        exit 0;
        ;;
restart)
        stop
        start
        ;;
status)
        find_logstash_forwarder_process
        if [ $PID -gt 0 ]; then
           echo "logstash-forwarder is running with PID $PID"
            exit 0
        else
           echo "logstash-forwarder is not running"
            exit 1
        fi
        ;;
*)
        echo $"Usage: $0 {start|stop|restart|status}"
        RETVAL=1
esac
exit 0
```

- Se agregar el servicio a los niveles correspondiente de encendido y apagado
    - chkconfig --level 2345 logstash_forwarder on
    - chkconfig --level 016 logstash_forwarder off
- El proceso se gestiona a través del comando service:
    - service logstash_forwarder start
    - service logstash_forwarder stop
    - service logstash_forwarder status

Configuración de logstash-forwarder

La exportación de los logs al servidor de LogStash requerirá de la siguiente configuración en el archivo /etc/logstash-forwarder/config.json:

```
# The list of files configurations
"files": [
  # An array of hashes. Each hash tells what
  # what fields to annotate on events from t
  {
    "paths": [
      # single paths are fine
      "/var/log/messages"
      # globs are fine too, they will be per
      # to see if any new files match the wi
      #"/var/log/*.log"
    ],
    # A dictionary of fields to annotate on
    "fields": { "type": "rhel" }
  }, {
    "paths": [
      "/var/log/httpd/*access*log"
    ],
    "fields": { "type": "apache_log" }
  }, {
    "paths": [
      "/var/log/httpd/*error*log"
    ],
    "fields": { "type": "apache_error" }
  }
]
```

Ilustración 104: Captura de la configuración de logstash-forwarder

10.4 Anexo 4: Instalación de nxlog

La aplicación se encuentra disponible para su descarga gratis desde el sitio http://nxlog.org/products/nxlog-community-edition/download.
En esta página se listan los instaladores para cada una de las arquitecturas soportadas, entre ellas Microsoft Windows.

Platform	Version	Installer
	Windows	nxlog-ce-2.9.1347.msi

Ilustración 105: Captura de la versión actual de nxlog-ce

Después de descargarse el instalador se procede a su instalación. El proceso de instalación requiere permisos de administración, y la ubicación de la instalación por defecto depende del idioma del sistema y de la arquitectura de procesador:

- C:\Program Files\nxlog

- C:\Progam Files (x86)\nxlog
- C:\Archivos de Programas\nxlog
- C:\ Archivos de Programas (nx86)\nxlog

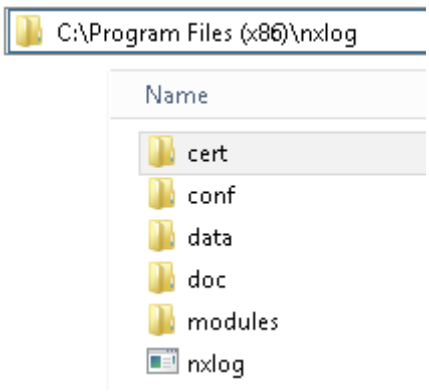


Ilustración 106: Captura del directorio de instalación de nxlog-ce

El archivo de configuración de nxlog se denomina nxlog.conf y se encuentra dentro de la carpeta conf.
La ejecución de la aplicación se gestiona como servicio de Windows a través de la consola de servicios (servicios.msc)

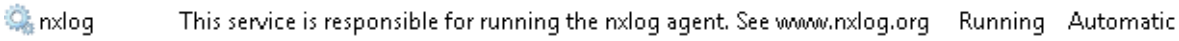


Ilustración 107: Captura del servicio nxlog

Configuración de nxlog

En esta ocasión la configuración de nxlog para recolectar los eventos generados y almacenados en el “Event Log” de servidores Microsoft Windows utiliza el módulo im_msvistalog para la lectura.
Para seleccionar y exportar los logs de Aplicaciones, del Sistema y de Seguridad en un formato JSON es necesario agregar la siguiente sección en el archivo de configuración nxlog.conf:

```
<Input EventLog>
    Module      im_msvistalog
# For windows 2003 and earlier use the following:
#   Module      im_mseventlog
    Query <QueryList>\
      <Query Id="0">\
            <Select Path="Application">*[Application/Level=3]
            </Select>\
            <Select Path="System">*[System/Level=3]</Select>\
            <Select Path="Security">*</Select>\
      </Query>\
      </QueryList>
      Exec $EventReceivedTime = integer($EventReceivedTime) /
1000000;\
     to_json();
</Input>
```

Ilustración 108: Captura de la configuración nxlog-ce

Estos eventos recolectados se enviarán al servidor LogStash, TCP puerto 514:

```
<Output EventLog_Out>
    Module      om_tcp
    Host        uavl001
    Port        514
</Output>
```

Ilustración 109: Captura de la configuración de salida en nxlog-ce

Para lo que es necesario interconectar las secciones de Entrada y Salida:

```
<Route 1>
    Path        EventLog => EventLog_Out
</Route>
```

Ilustración 110: Captura de la configuración de enrutamiento en nxlog-ce

10.5 Anexo 5: Configuración del firewall en RHEL para las entradas LogStash

La distribución de GNU/Linux RHEL 7 ejecuta por defecto un firewall (firewalld) que bloquea todo el tráfico por defecto. Por este motivo, además de configurar los módulos de entrada en LogStash, es necesario configurar el firewall en el servidor LogStash para permitir el tráfico desde/hacia estos módulos.
Para la consecución de este fin se creará un archivo en el directorio de servicios (/etc/firewalld/services) con una configuración base. Cada nuevo módulo de entrada configurado en Logstash (/etc/logstash/logstash.conf) deberá agregar una entrada correspondiente al protocolo y puerto utilizados.
Para la configuración base inicial se realizará el siguiente procedimiento:

- Crear un archivo en el directorio /etc/firewalld/services:
  - *touch /etc/firewalld/services/logstash.xml*
- Agregar el siguiente contenido al archivo creado:

```
<?xml version="1.0" encoding="utf-8"?>
<service>
 <short>logstash</short>
 <description>LogStash Ports</description>
</service>
```

- Agregar el servicio al firewall:
  - *firewall-cmd --permanent --add-service=lsyslog*
  - *firewall-cmd --reload*
  - *firewall-cmd --list-services*

```
[root@uavl001 services]# firewall-cmd --permanent --add-service=logstash
success
[root@uavl001 services]# firewall-cmd --reload
success
[root@uavl001 services]# firewall-cmd --list-services
dhcpv6-client elasticsearch http logstash ntopng redis ssh
```

Ilustración 111: Captura del servicio de firewall en los servidores RHEL 7

Cada vez que ingrese un nuevo módulo de entrada de red en la configuración logstash será necesario insertar una entrada en el archivo de servicio /etc/firewalld/services/logstash.xml con el siguiente formato:

<port protocol=”<udp o tcp>” port=”<número de puerto>” />

Por ejemplo en el caso del siguiente módulo de entrada en logstash:

```
udp {
 port => 514
 type => "3com"
}
```

Ilustración 112: Captura configuración de entrada LogStash para 3Com

La entrada correspondiente debe ser: <port protocol=”udp” port=”514” />
Y debería insertarse en el archivo de configuración del firewall:

```
[root@uavl001 services]# more /etc/firewalld/services/logstash.xml
<?xml version="1.0" encoding="utf-8"?>
<service>
 <short>logstash</short>
 <description>Logstash Ports</description>
 <port protocol="udp" port="514" />
</service>
```

Ilustración 113: Captura del archivo de configuración para un servicio de firewall

Para que la configuración del firewall entre en vigencia Luego de lo cual es necesario aplicar el comando *firewall-cmd –reload* para recargar la configuración del firewall.

10.6 Anexo 6: Instalación de Kibana

El primer paso consiste en la descarga e instalación de Kibana desde el sitio https://www.elastic.co/downloads/kibana.

```
[root@uavl002 ~]# curl -O https://download.elastic.co/kibana/kibana/kibana-4.0.2
-linux-x64.tar.gz
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 13.2M  100 13.2M    0     0   375k      0  0:00:36  0:00:36 --:--:--  448k
```

Ilustración 114: Captura de la descarga del archivo de instalación para Kibana

A continuación se procederá a la instalación en el servidor uavl002 dónde se encuentra la instancia de ElasticSearch. La carpeta de instalación será /opt:

- Se desempaqueta el archivo *tar -xvzf kibana-4.0.2-linux-x64.tar.gz –C /opt*
- Se crea un enlace lógico *ln –s /opt/ kibana-4.0.2-linux-x64 /opt/kibana.*

```
[root@uavl002 ~]# ls /opt
kibana  kibana-4.0.2-linux-x64
```

Ilustración 115: Captura del enlace de instalación Kibana

Automatización del servicio Kibana

La automatización de la ejecución de Kibana mediante el sistema de gestión de servicios systemd consiste en:

- La creación de un archivo que identificará al servicio kibana en el directorio /usr/lib/systemd/system/:
  - touch /usr/lib/system/system/kibana.service
- A continuación se agregará al archivo creado el siguiente contenido:

```
[Unit]
Description=Starts and stops a single kibana instance on this system
Documentation=http://www.elasticsearch.org
[Service]
ExecStart=/opt/kibana/bin/kibana
Restart=always
StandardOutput=syslog
StandardError=syslog
SyslogIdentifier=kibana4
User=root
Group=root
Environment=NODE_ENV=production
[Install]
WantedBy=multi-user.target
```

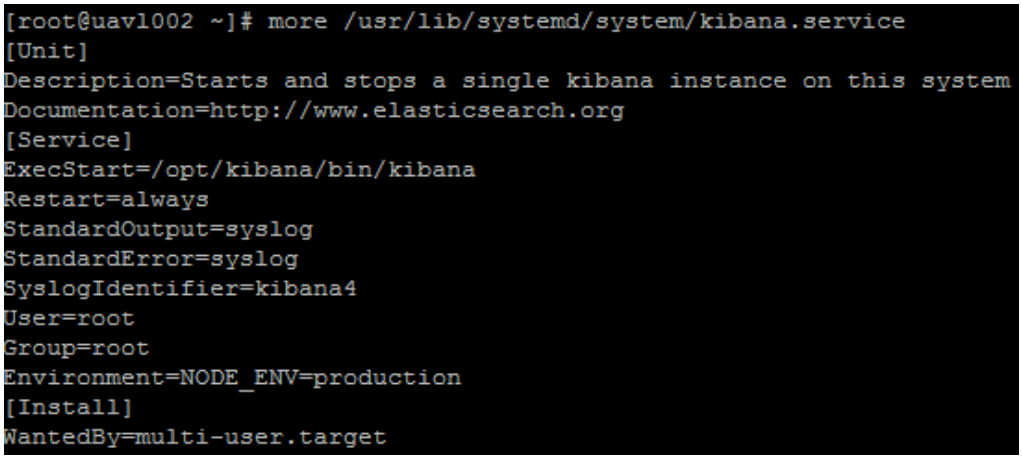


Ilustración 116: Captura del script de inicialización de Kibana

- Finalmente se prueba el servicio mediante los siguientes comandos:
  - *systemctl daemon-reload*
  - *systemctl enable kibana*
  - *systemctl start kibana*
  - *systemctl status kibana*

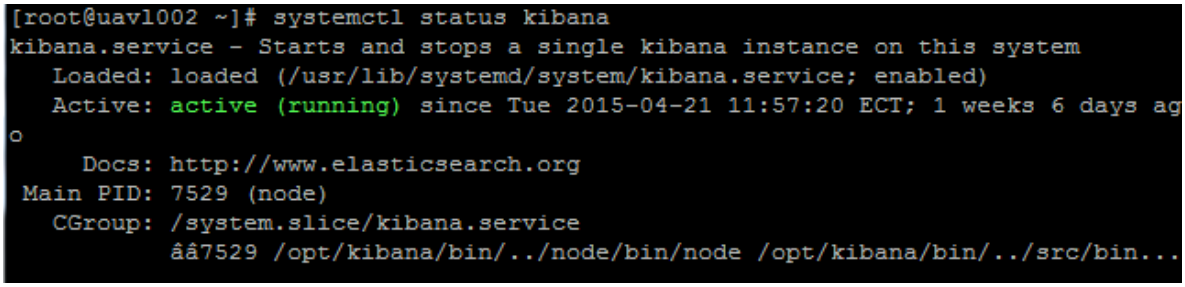


Ilustración 117: Captura del mecanismo de gestión de servicios para Kibana

Configuración del firewall en RHEL para Kibana

Antes de continuar con la configuración de los índices en Kibana se requiere abrir el puerto TCP 5601 a las conexiones externas. Para la consecución de este fin se creará un archivo en el directorio de servicios (/etc/firewalld/services) con una configuración base:

- *touch /etc/firewalld/services/kibana.xml*

Y se agregará al archive el siguiente contenido:

```
<?xml version="1.0" encoding="utf-8"?>
<service>
 <short>Kibana</short>
 <description>Kibana Explorer</description>
 <port protocol="tcp" port="5601" />
</service>
```

Finalmente se agregará el servicio y se recargará el firewall:

- *firewall-cmd --permanent --add-service=kibana*
- *firewall-cmd –reload*