Pontificia Universidad Católica del Ecuador

FACULTAD DE INGENIERÍA
COORDINACIÓN DE POSGRADO

PUCE

**PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR**

**FACULTAD DE INGENIERÍA**

Trabajo de Titulación como requisito previo para la obtención del título de Magíster en Tecnologías de Información mención Gestión y Administración de TI

# DISEÑO DE IMPLEMENTACIÓN DE UNA HERRAMIENTA OPEN SOURCE PARA MONITOREO DE SERVIDORES

**Autor**: Ing. Edwin Ricardo Sánchez Osejo

**Director**: MSc. Santiago David Silva Proaño

Quito, 2022

**PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR**

**DECLARACIÓN Y AUTORIZACIÓN**

Yo, **EDWIN RICARDO SÁNCHEZ OSEJO**, con C.I. 040112053-0, autor del trabajo de titulación titulado: **"DISEÑO DE IMPLEMENTACIÓN DE UNA HERRAMIENTA OPEN SOURCE PARA MONITOREO DE SERVIDORES"** previo para la obtención del título de Magíster en Tecnologías de Información mención Gestión y Administración de TI.

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

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

_____

Edwin Ricardo Sánchez Osejo

## APROBACIÓN DEL TUTOR

En mi carácter de Director (a) – Tutor (a) del Trabajo de Posgrado Titulado: **"DISEÑO DE IMPLEMENTACIÓN DE UNA HERRAMIENTA OPEN SOURCE PARA MONITOREO DE SERVIDORES",** presentado por el maestrante EDWIN RICARDO SÁNCHEZ OSEJO, titular de la Cédula de Identidad Nº 040112053-0 para optar al Grado de Magíster en Tecnologías de Información con mención en Gestión y Administración de TI, considero que dicho Trabajo de Investigación reúne los requisitos y méritos suficientes para ser sometido a la evaluación por parte de los Lectores – Evaluadores que se designen para tal fin por parte de las autoridades de la Facultad de Ingeniería.

En la ciudad de Quito, a los 16 días de diciembre de 2022

_____

Santiago David Silva Proaño C.I. 171481593-1

ssilva068@puce.edu.ec

NRO. TELÉFONO: 099 238 8692

**NOTA:**

Se comunica que en el servicio de análisis Turnitin, el referido trabajo de titulación alcanzó el siguiente resultado: 8 % índice de similitud con otras fuentes

**TURNITIN: INCLUIR HOJA DEL INFORME CON EL PORCENTAJE**

Diseño de Implementación de una Herramienta Open Source para Monitoreo de Servidores

ORIGINALITY REPORT

| 8% | 7% | 1% | 5% |
|---|---|---|---|
| SIMILARITY INDEX | INTERNET SOURCES | PUBLICATIONS | STUDENT PAPERS |

## DECLARACIÓN DE AUTENTICIDAD Y RESPONSABILIDAD

Yo, Edwin Ricardo Sánchez Osejo, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondientes a este trabajo, a la Pontificia Universidad Católica del Ecuador – PUCE, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

_____

**Edwin Ricardo Sánchez Osejo**

## DEDICATORIA

*A mi esposa Nuvia Esthela, mis hijos Edwin Sebastián y Esteban Isaac, mis padres Edwin Fabian, Aura Elisa, mis hermanas Erika Jazmín, Yessenia Carolina, mis amigos Jorge Humberto, Marco Vinicio que han estado pendiente y siempre apoyan mis locuras y decisiones para seguir adelante.*

.

*Edwin Ricardo Sánchez Osejo*

# ÍNDICE DE CONTENIDO

# ÍNDICE DE TABLAS

## ÍNDICE DE ILUSTRACIONES

PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR
FACULTAD DE INGENIERÍA
MAESTRÍA EN TECNOLOGÍAS DE LA INFORMACIÓN MENCIÓN GESTIÓN
Y ADMINISTRACIÓN DE TI

# DISEÑO DE IMPLEMENTACIÓN DE UNA HERRAMIENTA OPEN SOURCE PARA MONITOREO DE SERVIDORES

Autor: Ing. Edwin Ricardo Sánchez Osejo
Director -Tutor: Msc. Santiago David Silva Proaño
Fecha: 17/11/2022

## RESUMEN

Este trabajo de investigación detalla la instalación e implementación de una herramienta de monitoreo de código abierto (open source) para servidores. Fue enfocado en la recolección de métricas de infraestructura y logs en equipos con sistema operativo Linux y Windows. Los datos recolectados mediante la herramienta de monitoreo Elastic, se indexaron para realizar búsquedas personalizadas y ser presentados los resultados de una manera visual, a través de un portal web.

Para lo cual se desarrolló un entorno de pruebas, donde se presenta los pasos a seguir, como la topología e infraestructura implementada. Mediante las métricas de infraestructura recolectadas, como procesamiento, memoria y disco duro, se crearon gráficas para conocer el consumo de los recursos por servidor. Además, con los eventos de bloqueo y modificación de atributos en las cuentas de usuarios en un Directorio Activo, se crearon gráficas para detectar posibles amenazas.

Al término de este trabajo investigativo se presentan las conclusiones y recomendaciones obtenidas durante el desarrollo del trabajo.

**Palabras clave:**
Monitoreo, logs, directorio activo, métricas de infraestructura

PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR
FACULTAD DE INGENIERÍA
MAESTRÍA EN TECNOLOGÍAS DE LA INFORMACIÓN MENCIÓN GESTIÓN Y ADMINISTRACIÓN DE TI

# IMPLEMENTATION DESIGN OF AN OPEN SOURCE TOOL FOR SERVER MONITORING

Autor: Ing. Edwin Ricardo Sánchez Osejo
Director -Tutor: Msc. Santiago David Silva Proaño
Fecha: 17/11/2022

## ABSTRACT

This research studies the details about the installation and implementation of an open-source monitoring tool for servers. It is focuses on the collection of infrastructure metrics and logs on computers with Linux and Windows operating systems. The data collected through the monitoring tool “Elastic” was indexed to perform personalized searches and the results were presented in a visual way, through a web portal.

For this purpose, a test environment was developed in order to show the step by step procedure, the topology and the implemented infrastructure. Through the infrastructure metrics collected, which are processing, memory, and hard disk, based on these infrastructure metrics, graphs are created to graphically represent the consumption of resources per server. In addition, with the information obtained about blocked user accounts and the modification of their attributes in an active directory, graphs are generated to detect possible threats.

Finally, at the end of this investigation work, the conclusions and recommendations obtained during the execution of this research project are presented and discussed.

**Keywords**
Monitoring, logs, active directory, infrastructure metrics.

# CAPÍTULO I: PLANTEAMIENTO DEL PROBLEMA

## 1.1. Formulación del Problema

Durante la investigación para este trabajo de titulación se ha podido verificar que las empresas que ofertan servicios y productos, tienen equipamiento tecnológico y sistemas de información críticos, sobre los que funcionan sus diferentes servicios y aplicativos, por lo que es necesario contar con herramientas de monitoreo. Para esto existen en el mercado herramientas de código abierto y propietarias que se enfocan en monitorear los diferentes recursos del hardware de sistemas operativos Windows y Linux, así como protocolos de servicios, métricas de infraestructura tales como: CPU, memoria RAM, red, almacenamiento y registros de logs.

Mientras mayor es el número de servicios, herramientas y aplicaciones tecnológicas que manejan las empresas, crece el riesgo tecnológico. Por tal razón los administradores y personal técnico se ven en la necesidad de monitorear en tiempo real los registros de logs[1] que se generan en el equipamiento tecnológico y de esta manera evitar exponerse a factores como: suplantación de identidad, robo y alteración de información. Esto puede traer consecuencias a la empresa tales como: pérdidas económicas, desprestigio, inconsistencia de la información e incluso implicaciones legales.

Por este motivo es recomendable contar con herramientas de monitoreo, para la recolección y análisis de logs que permitan visualizar de manera oportuna posibles amenazas como inicios de sesión fallidos, cambio de atributos en cuentas de usuario de un Directorio Activo, etc. De

[1] Es un archivo que registra las actividades que se dan en el servidor o aplicación. (Lerena, 3)

esta forma se busca minimizar las tareas de búsquedas de los registros de eventos en equipos con sistemas operativo Windows y Linux.

## 1.2. Objetivos de la Investigación

### Objetivo General

Diseñar la implementación de una herramienta de monitoreo en servidores, utilizando la plataforma Elastic[2] de código abierto (open source), para aplicarla a la recolección de métricas de infraestructura, recolección de logs, etc. Con el propósito final que puedan ser visualizadas mediante un portal web.

### Objetivos Específicos

- Explorar las características de la herramienta Elasticsearch[3] para un monitoreo y analítica de logs.
- Desarrollar una guía técnica y práctica de consulta para futuras implementaciones de la herramienta de monitoreo.
- Diseñar presentaciones gráficas, numéricas y descriptivas con la herramienta Kibana[4] para que se puedan visualizar en un Panel de Control o Dashboard.
- Visualizar los logs de seguridad de Windows, referente al bloqueo y modificación de atributos de las cuentas de usuarios en un servidor de Directorio Activo.

[2] Es un grupo de herramientas que sirven para monitoreo, análisis de datos y explotación de información en tiempo real. (Sanchez Avalo, 2020)
[3] Es un motor de búsqueda de código abierto (Elastic, s.f.)
[4] Es una aplicación para visualizar datos (Elastic, s.f.)

## 1.3. Justificación de la Investigación

Este trabajo busca aprovechar las herramientas de código abierto para implementar una plataforma de monitoreo, la misma que permite hacer el seguimiento de recursos tales como: procesamiento, memoria RAM, red y almacenamiento. Además, busca agrupar al equipamiento de una manera amigable para optimizar el tiempo de búsqueda y análisis de los eventos y logs que generan los equipos. Todo esto con la finalidad de brindar claridad el uso de los recursos tecnológicos y posibles amenazas en el equipamiento, mismos que ayudan a la toma de decisiones en la optimización de los recursos tecnológicos y protección de la información.

Para escoger la plataforma de monitoreo se usó la información emitida por el cuadrante Mágico de Gartner referente a herramientas de Monitoreo en el año 2021, donde se observó que existen varias opciones propietarias y de código abierto, siendo Elastic una buena opción para el estudio, por caracterizarse en recolectar, procesar y presentar datos de una manera ágil, fácil y didáctica, para los ambientes basados en Windows y Linux.

# CAPÍTULO II: MARCO TEÓRICO

En este capítulo se presenta conceptos, definiciones de las herramientas utilizadas para el correcto manejo del monitoreo de los servidores con sistemas operativos Windows y Linux.

## 2.1 Software

El software se conoce como la parte lógica de un sistema informático, es decir toda la información procesada en los sistemas informáticos: programas y datos, que en conjunto con otros componentes lógicos pueden hacer alguna tarea específica. Este componente se ejecuta en el hardware o componentes físicos, la palabra "software" fue utilizado por primera vez por John W. Tukey en 1957. (Sánchez López, 2013)

### 2.1.1 Software Propietario

El software propietario o privativo es aquel programa que tiene un dueño o ente que dispone de sus derechos de creación, mismo que limita su uso, modificación y distribución de manera libre. Para poder utilizarlo se debe cancelar un valor, su limitación es que no se puede analizar cómo fue creado, incorporar mejoras o llevarlos a una necesidad específica que se desee. (Llamas, Software propietario, s.f.)

### 2.1.2 Software Libre.

El software libre tiene como filosofía principal respetar la libertad de los usuarios y de la comunidad. Esto hace que los usuarios tengan la libertad de ejecutar, copiar, distribuir, estudiar, adaptar y mejorar el software y hacer público el producto que han liberado. No tiene un valor comercial y tiene la opción de difundir sin restricciones su producto. (GNU, s.f.)

### 2.1.3 Software Open Source

El software Open Source (código abierto), tiene como fin tener acceso libre al código fuente del software para que los interesados pueden visualizarlo y modificar hasta distribuir el nuevo código generado. Su manejo es descentralizado y colaborativo, además puede tener revisiones por medio de la comunidad.

El "código fuente" es la parte principal del software, y para los usuarios es transparente la existencia de este, por lo que ellos solo perciben su funcionalidad; tener el código fuente hace que los programadores informáticos pueden mejorar el software, añadiéndole características o modificando partes que no cumplen sus necesidades específicas. (Red Hat, 2019)

## 2.2 Monitoreo de servidores.

Esta actividad radica en supervisar los recursos físicos y lógicos de un servidor, como el procesamiento, memoria RAM, red, almacenamiento, servicios, puertos de conexión, logs del sistema operativo y aplicaciones. Esta tarea permite comprender el uso de los recursos del servidor y planificar sus capacidades de manera óptima.

El disponer de un software de monitoreo de servidores facilita la automatización e identificación de posibles problemas relacionados al rendimiento, como la utilización elevada de recursos, tiempos de inactividad de las aplicaciones y tiempo de respuesta de los servicios o aplicaciones. (ManageEngine, s.f.)

### 2.3 Evento de registro o logs

Los eventos de registros son archivos de texto, donde se almacenan eventos y datos de procesos que han sido definidos como selectos por el programador de la aplicación o del sistema operativo. Los logs se generan de forma automática según su programación, esto depende de que la aplicación genere nuevos archivos de logs con registros propios (Digital Guide IONOS, 2016)

### 2.4 Dashboard o Tablero de Control

Es una herramienta que gestiona la información, de manera visual, la misma que facilita presentar datos monitoreados como: logs, métricas y datos de la empresa, de esta forma se facilita el seguimiento a un proceso específico. Sus principales características son: ser visual, personalizable, práctico y presentar la información en tiempo real. (Ortiz, 2021)

### 2.5 Arquitectura de Elasticsearch, Logstash y Kibana (ELK).

Son un conjunto de herramientas de código abierto que se integran para realizar la administración de registros y permitir tareas como: monitorear, consolidar y analizar la información que se obtiene de los distintos servidores. Para esto se usa herramientas como Elasticsearch, Logstash y Kibana, que se explicarán en los siguientes puntos. (Pari, 2020).

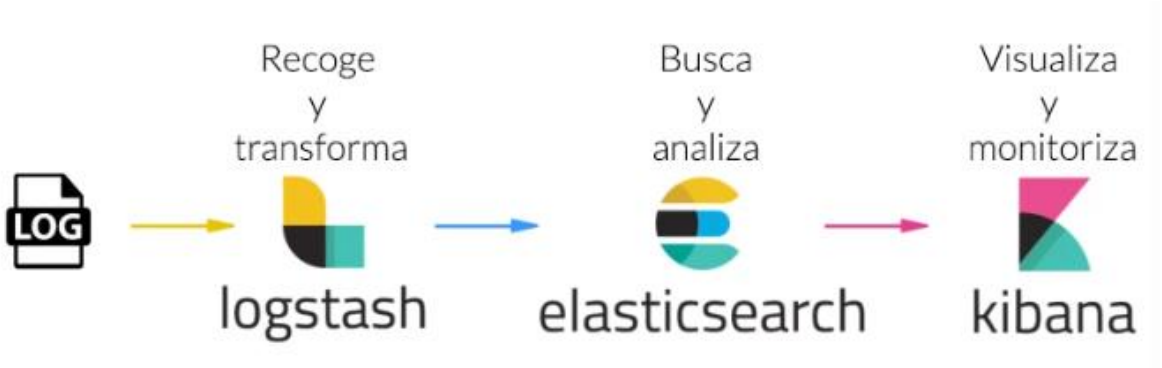


***Figura 2.1*** *Arquitectura de ELK*
*(Fuente: (Pari, 2020))*

### 2.5.1 Elasticsearch

Elasticsearch es una herramienta de código libre de la empresa Elastic, que se especializa como un motor de búsqueda y analítica de varios tipos de datos tales como: texto, números, geoespaciales, estructurados y no estructurados. Su escalabilidad y velocidad es su principal característica para simplificar el análisis de los datos (Elastic, s.f.)

### 2.5.2 Kibana

Kibana es una aplicación de frontend[5] que actúa como interfaz final del usuario, misma que permite visualizar y explorar los datos indexados en Elasticsearch o Logstash, se caracteriza por permitir la creación de Dashboard. Su principal función es la de presentar información de logs, métricas de recursos del equipamiento tecnológico, entre otros. (Elastic, s.f.).

### 2.5.3 Logstash

Logstash es una herramienta que se integra de forma nativa con Kibana y los agentes propios de Elastic. Tiene como su principal característica recolectar la información de múltiples orígenes de datos para transformarlos, normalizarlos y publicarlos mediante un Dashboard dentro de la herramienta Kibana. (Elastic, 2022)

### 2.5.4 Agente o "Beats"

Los beats se consideran como agentes ligeros o Thin Clients, estos no repercuten en el rendimiento del equipo donde se ejecuta. Se instalan en equipos Windows y Linux con la finalidad monitorear y transmitir (o enviar) la información al equipo principal, en donde se

[5] Frontend: Es el diseño frontal de un sitio web. (Bautista García, 2021)

tiene configurado el servicio de Elasticsearch. Existen varios tipos de agentes como se detalla a continuación, los mismos que se revisarán a detalle más adelante:

- Filebeat
- Heartbeat
- Metricbeat
- Winlogbeat (Elastic, 2022)

### a) Filebeat

Este es un agente que se caracteriza por leer eventos de registros o ficheros de logs solo en los equipos con sistema operativo Linux. Este agente monitorea, recopila y reenvía los eventos de registro a Elasticsearch o Logstash para su indexación[6], y mediante la herramienta Kibana crear visualizaciones con los datos recopilados.

Además, esta herramienta es capaz de gestionar el último evento enviado y continuar desde ese punto en caso de un corte en la comunicación, dispone de un control de carga para reducir el tamaño de envío de información y evitar saturación en la red. (Elastic, s.f.)

### b) Heartbeat

Es un agente que sirve para verificar de manera periódica el estado de los servicios y determinar si están accesibles en el equipo. El monitoreo se puede realizar a través de los siguientes protocolos:

- El ICMP (v4 o v6) o protocolo de control de mensajes de Internet. Ayuda a comprobar si el equipo está disponible en la red.
- TCP o protocolo de control de transmisión. Indica si un puerto está disponible.

[6] Método para ordenar datos o información de acuerdo a un criterio común a ellos. (Banco de la República, 2018)

- HTTP o Protocolo de trasferencia de hipertexto. Se configura para validar que el servicio web se encuentre operativo. (Davinci Group, 16) (Elastic, s.f.)

### c) Metricbeat

Este agente recoge de manera periódica, las métricas del servidor de parámetros como: memoria RAM, procesamiento, disco duro, tipo sistema operativo entre otros; para renviar al equipo con la herramienta Elasticsearch y posteriormente crear visualizaciones con la información recopilada. (Elastic, s.f.)

### d) Winlogbeat

Es un agente que se especializa en leer los registros de eventos o logs de equipos con sistema operativo Windows. Este agente monitorea, recopila y reenvía los eventos a la herramienta Elasticsearch o Logstash, para ser indexada y mediante la herramienta Kibana crear visualizaciones con los datos recopilados.

El agente Winlogbeat puede reenviar los datos que se almacenan, en los registros de eventos tales como:

- Eventos de aplicación
- Eventos de hardware
- Eventos de seguridad
- Eventos del sistema (Elastic, s.f.)

### 2.6 Características de Elasticsearch

Es una herramienta que está diseñada para almacenar y analizar diversos tipos de datos, que se obtienen con la ayuda de los diferentes agentes. Esta herramienta posee algunas características según manifiestan (Rodriguez Flores, 2019) y (Santos González, 2019) las cuales se detallan a continuación:

- Alta disponibilidad mediante la implementación de clústeres[7], es capaz de aislar el nodo defectuoso y reorganizarse para hacer que los datos siempre estén activos.
- Escalabilidad horizontal, se pueden agregar nodos según la necesidad.
- La visualización de resultados en tiempo real es más rápida por la indexación previa de los datos.
- Compatibilidad con la herramienta Kibana, para presentar de manera gráfica los datos procesados.
- La información almacenada no se guarda en una base de datos relacional.
- Facilidad en la búsqueda de un determinado texto o dato.
- Está orientado en JSON[8], para realizar las búsquedas.
- Permite indexar gran cantidad de datos, para utilizarlos en consultas específicas.
- Se almacena los documentos mediante el uso de índices[9].
- Integración sencilla con las APIs[10] de los distintos agentes y módulos.

[7] Conjunto de nodos, que gestionan mismos servicios y distribuyen la carga para mantener disponible el servicio. (IBM, 2022)
[8] (JavaScript Object Notation) es un archivo que contiene una serie de datos estructurados en formato de texto y se usa para transferir información entre sistemas (MDN Plus, 2022)
[9] Referencia de ordenamiento de información, para una localización y búsqueda más rápida. (Palma, 2005)
[10] Facilitan la interacción entre 2 aplicaciones mediante una serie de reglas. (Red Hat, 2017)

# CAPÍTULO III: IMPLEMENTACIÓN Y DISEÑO

## 3.1 Determinación de la Herramienta de implementación.

Al existir varias herramientas de monitoreo de tipo propietarias, libres y de código abierto, se realizó la validación usando el Cuadrante de Gartner, referente a motores de analítica de información (Insight Engines) del año 2021. Dentro de las herramientas y soluciones que ahí se explican, y tomando como característica principal que debe ser Open Source, se ha escogido la solución Elastic misma que es considerada como retador (Challenger), como se presenta en la Figura 3.1.

Esta herramienta ofrece una solución de búsqueda moderna, poderosa, abierta y gratuita para ser implementada en cualquier empresa, organización o instituciones de carácter gubernamental, privado, mixto o internacional.


***Figura 3.1*** *El Cuadrante de Gartner 2021 sobre analítica de información*
*Fuente: (Riley, 2021))*

Entre algunas características destacadas que la herramienta Elastic posee, según lo que manifiestan (Gartner, 2022) y (Stephen Emmott, 2021), se detallan a continuación:

- Modelo de negocio. No existe impedimento para realizar pruebas de la herramienta, se encuentra disponible de una manera libre.
- Innovación. Fácil y personalizable en el modelo de administración.
- Múltiples idiomas. Permite tener varios idiomas.
- Facilita el análisis de datos mediante la indexación.
- Monitoreo de aplicaciones y sistemas operativos.
- Visualización de los eventos en tiempo real,
- Escalable y fácil en el crecimiento para tener alta disponibilidad
- Gestiona los datos históricos en informes.

### 3.2 Diseño de entorno de trabajo.

Para la elaboración de este proyecto se optó en tener un ambiente heterogéneo de servidores físicos y virtuales. En el ambiente físico se usó los equipos que se indican en la Tabla 3.1.

***Tabla 3.1*** *Descripción de servidores físicos*

| Nombre del Equipo | RAM (GB) | CPU | Disco Duro (GB) | Sistema Operativo | Rol |
|---|---|---|---|---|---|
| SESX18 | 96 | 2 | 600 | VMware 6.5 | Virtualización |
| SADD03 | 96 | 2 | 600 | Windows Server 2016 | Directorio Activo |

*(Fuente: Autor)*

Para el ambiente virtual, mediante la plataforma de virtualización VMware[11] se crearon máquinas virtuales para aprovechar los recursos del equipo físico, con las siguientes características, según se presenta en la Tabla 3.2.

**Tabla 3.2** Descripción de servidores virtuales

| Nombre del Equipo | RAM (GB) | vCPU | Disco Duro (GB) | Sistema Operativo | Rol |
|---|---|---|---|---|---|
| PMON04 | 32 | 8 | 600 | Red Hat 8 | Elasticsearch |
| DAPJ55 | 6 | 4 | 60 | Red Hat 8.4 | App |
| DWEB01 | 2 | 4 | 90 | Red Hat 7 | App |
| SADD06 | 16 | 4 | 150 | Windows Server 2016 | Directorio Activo |
| SADD07 | 16 | 6 | 150 | Windows Server 2016 | Directorio Activo |
| PADD02 | 4 | 2 | 80 | Windows Server 2016 | Directorio Activo |
| DSQL17 | 12 | 6 | 138 | Windows Server 2016 | Base Datos |

*(Fuente: Autor)*

### 3.3 Arquitectura del entorno de trabajo

Para el entorno de trabajo, se plantea una arquitectura heterogénea de 8 equipos, en cada uno de ellos se instalan las aplicaciones mismas que se analizan más adelante para equipos con sistemas operativos Linux y Windows. En la Figura 3.2 se muestra la arquitectura del ambiente propuesto.

[11] Es un software utilizado para imitar las características del hardware y crear equipos de manera virtual. (Pure Storage, s.f.)

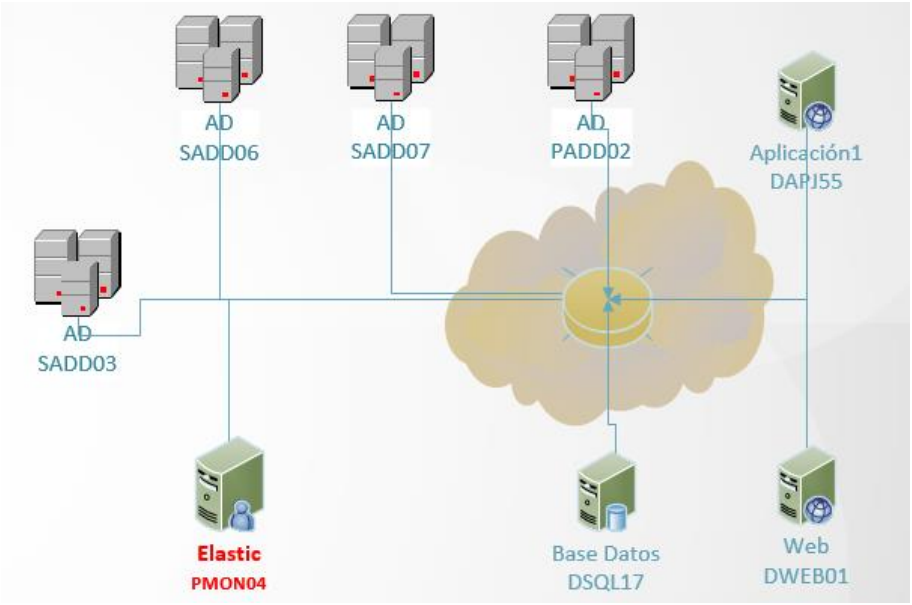


***Figura 3.2*** *Arquitectura del entorno de trabajo*
*(Fuente: Autor)*

Para el servidor principal (Elastic – PMON04) se realizó la instalación de los componentes Elasticsearch, Kibana y Logstash como aplicaciones bases. Además, se instalaron los agentes Metricbeat y Filebeat, como módulos de monitoreo del equipo e interconexión con la herramienta Kibana, tal como se muestra en la Figura 3.3.

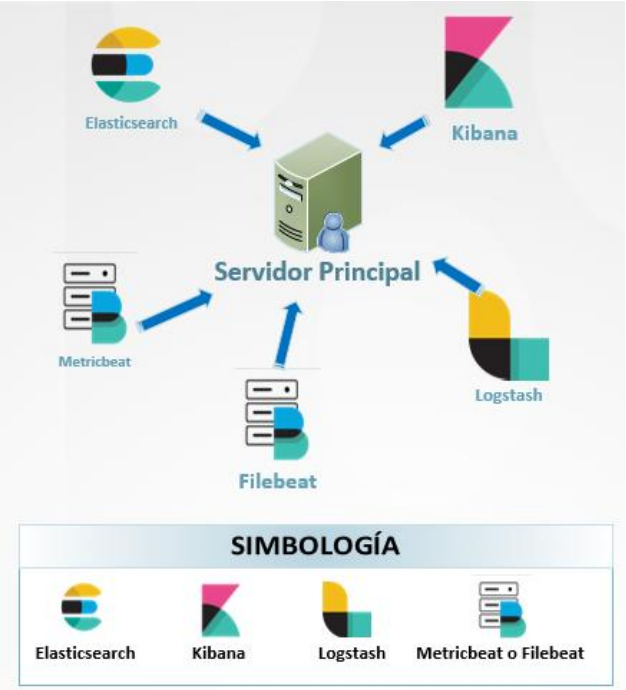


***Figura 3.3*** *Herramientas hacer instaladas en el servidor principal*
*(Fuente: Autor)*

En los equipos clientes del entorno de trabajo, se instalaron los agentes o “beat”, tal como se describe en la Tabla 3.4.

***Tabla 3.3*** *Agentes instalados en cada servidor*

| Nombre del Equipo | Sistema Operativo | Heartbeat | Metricbeat | Filebeat | Winlogbeat |
|---|---|---|---|---|---|
| DWEB01 | Red Hat 8 | X | X | X | |
| DAPJ55 | Red Hat 8.4 | X | X | X | |
| SADD03 | Windows Server 2016 | X | X | | X |
| SADD06 | Windows Server 2016 | X | X | | X |
| SADD07 | Windows Server 2016 | X | X | | X |
| PADD02 | Windows Server 2016 | X | X | | X |
| DSQL17 | Windows Server 2016 | X | X | | |

*(Fuente: Autor)*

La Herramienta Elastic se caracteriza por ser flexible en la implementación de la solución. De esa manera se facilita la adaptación de la arquitectura planteada con todos los componentes, tal como se indica en la Figura 3.4

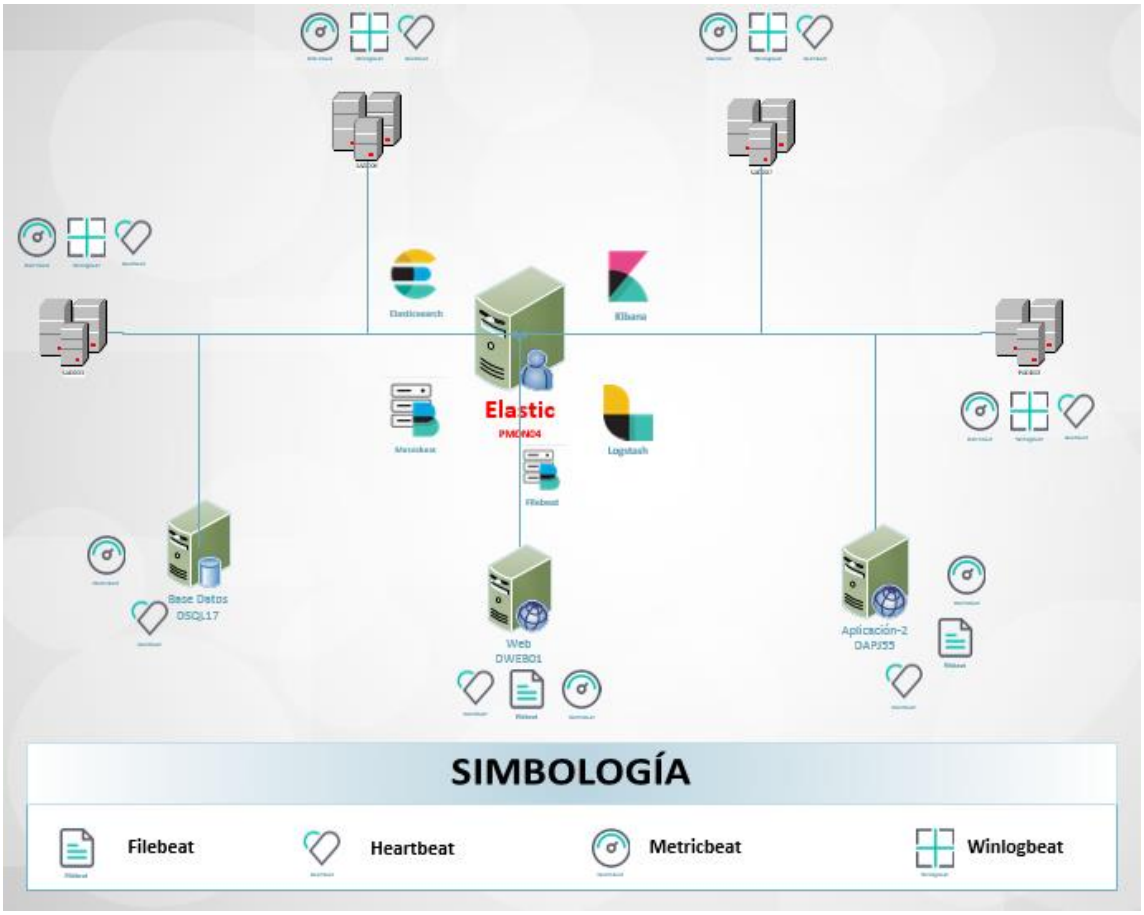


***Figura 3.4*** *Arquitectura con los componentes en cada equipo*
*(Fuente: Autor)*

### 3.4 Implementación y configuración de la herramienta de monitoreo.

Como parte de este trabajo se presenta una guía de implementación y configuración de la herramienta de monitoreo Elastic con sus aplicaciones y agentes, los mismos que se detallan a continuación.

El servidor principal Elastic, se instaló bajo la distribución de Red Hat Enterprise Linux 8.5. La configuración debe ser realizada con el usuario root, y ejecutar los comandos como se indica en la Figura 3.5.

```
yum update
yum install vim
```

***Figura 3.5*** *Comandos de instalación como prerrequisitos*
*(Fuente: Autor)*

Como pre-requisito esencial es necesario la instalación del paquete Java openjdk[12], así como se indica en la Figura 3.6:

*dnf install java-1.8.0-openjdk*

```
[root@pchquit01pmon04 ~]#
[root@pchquit01pmon04 ~]# dnf install java-1.8.0-openjdk
Updating Subscription Management repositories.
Red Hat Satellite Tools 6.5 for RHEL 8 x86_64 (RPMs)                33 kB/s | 2.1 kB     00:00
Red Hat Enterprise Linux 8 for x86_64 - Supplementary (RPMs)        34 kB/s | 2.1 kB     00:00
Red Hat Enterprise Linux 8 for x86_64 - AppStream (RPMs)            38 kB/s | 2.8 kB     00:00
Red Hat Enterprise Linux 8 for x86_64 - BaseOS (RPMs)               38 kB/s | 2.4 kB     00:00
Dependencies resolved.
================================================================================================================================
 Package                        Arch       Version                                  Repository                             Size
================================================================================================================================
Installing:
 java-1.8.0-openjdk             x86_64     1:1.8.0.312.b07-2.el8_5                  rhel-8-for-x86_64-appstream-rpms      341 k
Installing dependencies:
 copy-jdk-configs               noarch     4.0-2.el8                                rhel-8-for-x86_64-appstream-rpms       31 k
 java-1.8.0-openjdk-headless    x86_64     1:1.8.0.312.b07-2.el8_5                  rhel-8-for-x86_64-appstream-rpms       34 M
 javapackages-filesystem        noarch     5.3.0-1.module+el8+2447+6f56d9a6         rhel-8-for-x86_64-appstream-rpms       30 k
 lksctp-tools                   x86_64     1.0.18-3.el8                             rhel-8-for-x86_64-baseos-rpms         100 k
 ttmkfdir                       x86_64     3.0.9-54.el8                             rhel-8-for-x86_64-appstream-rpms       62 k
 tzdata-java                    noarch     2021e-1.el8                              rhel-8-for-x86_64-appstream-rpms      191 k
 xorg-x11-fonts-Type1           noarch     7.5-19.el8                               rhel-8-for-x86_64-appstream-rpms      522 k
Enabling module streams:
 javapackages-runtime                      201801

Transaction Summary
================================================================================================================================
Install  8 Packages

Total download size: 35 M
Installed size: 120 M
Is this ok [y/N]:
```

***Figura 3.6*** *Instalación del paquete Java openjdk*
*(Fuente: Consola de servidor principal)*

### 3.4.1 Elasticsearch instalación y configuración en el servidor.

Para la instalación de la herramienta Elasticsearch, se debe ingresar mediante una conexión SSH al servidor Elastic (PMON04), usando el usuario *root* y seguir los siguientes pasos:

**Paso 1.** Descargar el instalador

Para la instalación de la herramienta Elasticsearch, se debe descargar el instalador, desde la página oficial de Elastic, para esto se ejecuta el comando de la Figura 3.7.

[12] Es la versión libre para el lenguaje orientado a objetos de la plataforma de desarrollo Java. (SG, 2012)

*wget https://artifacts.elastic.co/downloads/elasticsearch/elasticsearch-8.2.0-x86_64.rpm*


***Figura 3.7*** *Descarga del instalador de la herramienta Elasticsearch*
*(Fuente: Consola de servidor principal)*

**Paso 2.** Creación del repositorio

En la Figura 3.8 se presenta la sintaxis utilizada, para la creación del archivo de configuración como repositorio para la instalación de la herramienta Elasticsearch.

*vim /etc/yum.repos.d/elasticsearch.repo*

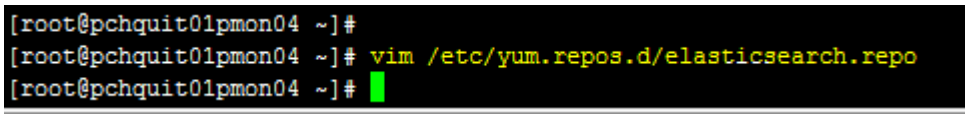


***Figura 3.8*** *Creación del archivo para repositorio en descargas*
*(Fuente: Consola de servidor principal)*

**Paso 3.** Edición del repositorio de instalación

Este repositorio tiene la funcionalidad de garantizar que los instaladores vengan desde la página oficial de Elastic, con esto se busca para evitar vulnerabilidades de la aplicación. Para esto se coloca las siguientes líneas de código:

```
[elasticsearch]
name=Elasticsearch repository for 8.x packages
baseurl=https://artifacts.elastic.co/packages/8.x/yum
gpgcheck=1
gpgkey=https://artifacts.elastic.co/GPG-KEY-elasticsearch
enabled=0
autorefresh=1
type=rpm-md
```

**Paso 4.** Instalación de la herramienta Elasticsearch

Para proceder con la instalación de la herramienta Elasticsearch, es necesario ejecutar la sintaxis presentada en la Figura 3.9.

```
yum localinstall elasticsearch-8.2.0-x86_64.rpm
```

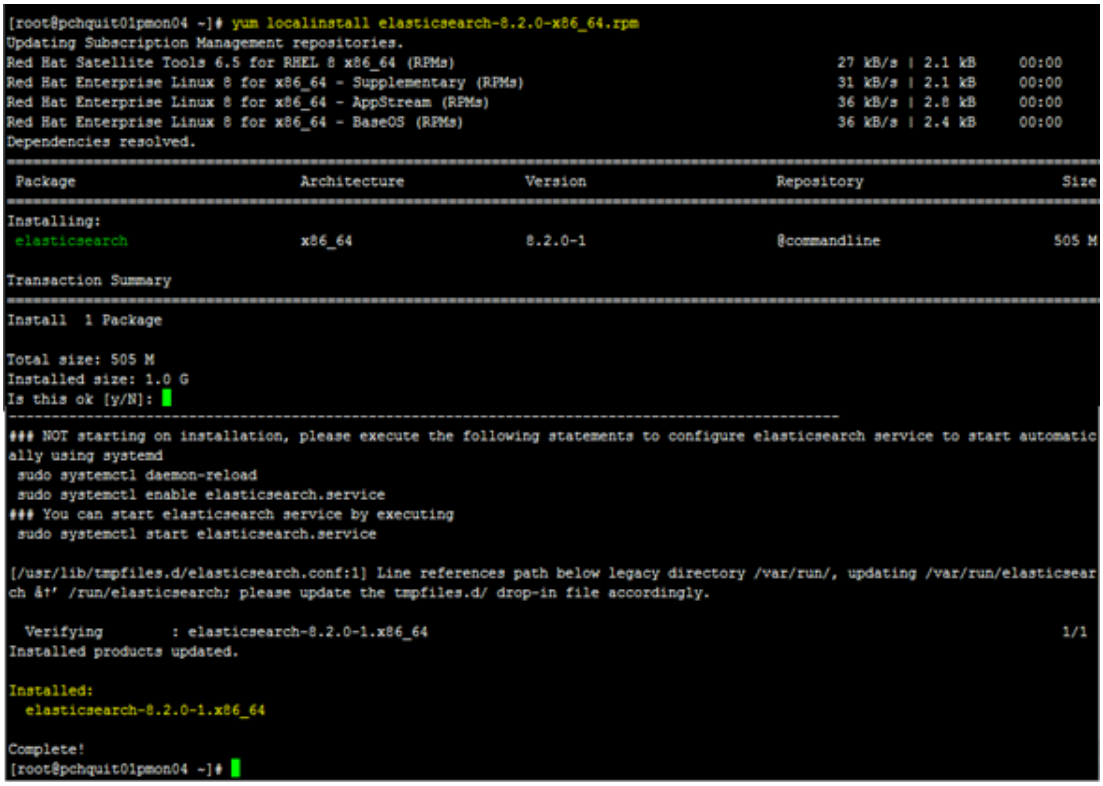


***Figura 3.9*** *Instalación de la herramienta Elasticsearch*
*(Fuente: Consola de servidor principal)*

**Paso 5.** Edición del archivo de configuración de Elasticsearch

Es necesario modificar el archivo de configuración de la herramienta Elasticsearch, con el editor *vim*, el mismo que se ubica en el directorio */etc/elasticsearch/elasticsearch.yml*, y realizar los cambios indicados en las Figuras 3.10 y 3.11.

```
vim /etc/elasticsearch/elasticsearch.yml
```

Continuando con la edición en el archivo de configuración, es necesario modificar los parámetros de red en la sección *"Network"*, en los campos: *network.host* y *http.port* como se presenta en la Figura 3.10, para configurar el puerto del equipo.

```
# ---------------------------------- Network -----------------------------------
#
# By default Elasticsearch is only accessible on localhost. Set a different
# address here to expose this node on the network:
#
network.host: 0.0.0.0
#
# By default Elasticsearch listens for HTTP traffic on the first free port it
# finds starting at 9200. Set a specific HTTP port here:
#
http.port: 9200
```

***Figura 3.10*** *Editar el archivo de configuración de Elasticsearch*
*(Fuente: Autor)*

Además, se debe editar las configuraciones del certificado SSL, en nuestro escenario al no tener una entidad certificadora se pone *"false"* en los campos que se indican en la Figura 3.11

```
# Enable security features
xpack.security.enabled: false

xpack.security.enrollment.enabled: false

# Enable encryption for HTTP API client connections, such as Kibana, Logstash, and Agents
xpack.security.http.ssl:
  enabled: false
  keystore.path: certs/http.p12

# Enable encryption and mutual authentication between cluster nodes
xpack.security.transport.ssl:
  enabled: false
  verification_mode: certificate
  keystore.path: certs/transport.p12
  truststore.path: certs/transport.p12
# Create a new cluster with the current node only
# Additional nodes can still join the cluster later
cluster.initial_master_nodes: ["pchquit01pmon04.fj.local"]

# Allow HTTP API connections from anywhere
# Connections are encrypted and require user authentication
http.host: 0.0.0.0
```

***Figura 3.11*** *Editar la sección del certificado del servicio Elasticsearch*
*(Fuente: Autor)*

**Paso 6.** Habilitar el servicio Elasticsearch

Es necesario configurar que el servicio de la herramienta Elasticsearch inicie de manera automática cuando el servidor se esté encendiendo, es preciso ejecutar la sintaxis presentada en la Figura 3.12.

*systemctl enable elasticsearch*

```
[root@pchquit01pmon04 ~]#
[root@pchquit01pmon04 ~]# systemctl enable elasticsearch
Created symlink /etc/systemd/system/multi-user.target.wants/elasticsearch.service
vice.
```

***Figura 3.12*** *Habilitar el servicio Elasticsearch*
*(Fuente: Consola de servidor principal)*

**Paso 7.** Activación de puerto en firewall local

En la Figura 3.13 se puede observar la sintaxis para la activación del puerto de la herramienta Elasticsearch, dicho puerto garantiza la aceptación de peticiones de los equipos clientes.

*firewall-cmd --add-port=9200/tcp --permanent*

```
[root@pchquit01pmon04 ~]#
[root@pchquit01pmon04 ~]# firewall-cmd --add-port=9200/tcp --permanent
success
```

***Figura 3.13*** *Agregar el puerto 9200 al Firewall*
*(Fuente: Consola de servidor principal)*

**Paso 8.** Validación del servicio Elasticsearch

Es necesario reiniciar el equipo, para validar el correcto funcionamiento del servicio de Elasticsearch, tal como se indica en la Figura 3.14.

*systemctl status elasticsearch.service*

```
[root@pchquit01pmon04 ~]# systemctl status elasticsearch.service
â elasticsearch.service - Elasticsearch
   Loaded: loaded (/usr/lib/systemd/system/elasticsearch.service; enabled; vendor preset: disabled)
   Active: active (running) since Wed 2022-08-10 11:55:09 -05; 2 months 4 days ago
     Docs: https://www.elastic.co
 Main PID: 1482 (java)
    Tasks: 208 (limit: 139671)
   Memory: 15.9G
   CGroup: /system.slice/elasticsearch.service
```

***Figura 3.14*** *Validar el servicio de Elasticsearch*
*(Fuente: Consola de servidor principal)*

Además, es indispensable realizar la validación del servicio de Elasticsearch, utilizando un Navegador o Browser para corroborar la disponibilidad del aplicativo en la red, tal como se indica en la Figura 3.15.

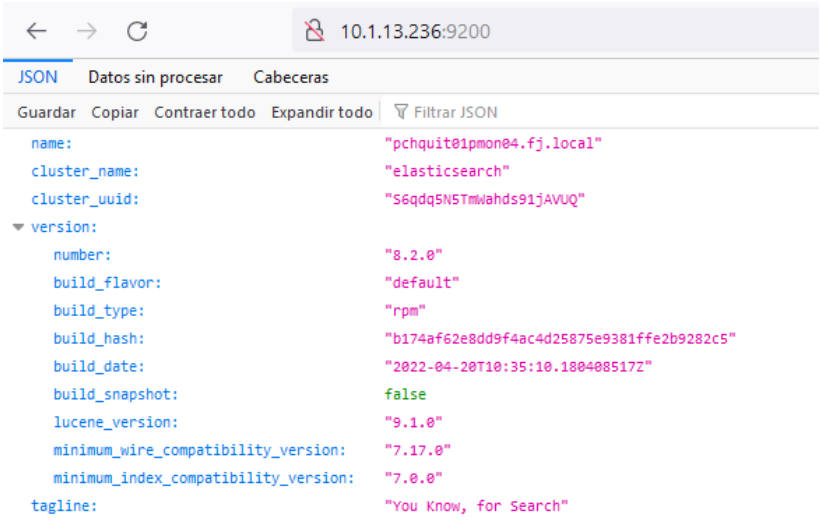


***Figura 3.15*** *Validación de Elasticsearch mediante un navegador*
*(Fuente: Consola de servidor principal)*

### 3.4.2 Kibana instalación y configuración en el servidor.

Para la instalación de la herramienta Kibana, se debe ingresar mediante una conexión SSH al servidor Elastic (PMON04), usando el usuario *root* y seguir los siguientes pasos:

**Paso 1.** Creación del repositorio

En la Figura 3.16 se presenta la sintaxis utilizada, para la creación del archivo de configuración como repositorio para la instalación de la herramienta Kibana.

*vim /etc/yum.repos.d/kibana.repo*

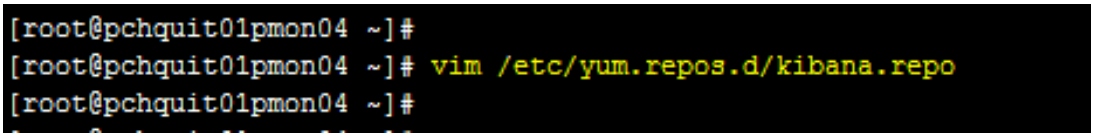


***Figura 3.16*** *Creación del archivo como repositorio para Kibana*
*(Fuente: Consola de servidor principal)*

**Paso 2.** Edición del repositorio de instalación

Este repositorio tiene la funcionalidad de certificar que los instaladores provienen de la página oficial de Elastic, con esto se busca evitar vulnerabilidades de la aplicación. Para lo cual se coloca las siguientes líneas de código:

```
 [kibana-8.x]
name=Kibana repository for 8.x packages
baseurl=https://artifacts.elastic.co/packages/8.x/yum
gpgcheck=1
gpgkey=https://artifacts.elastic.co/GPG-KEY-elasticsearch
enabled=1
autorefresh=1
type=rpm-md
```

**Paso 3.** Instalación de la herramienta Kibana

Para proceder con la instalación de la herramienta Kibana, es necesario ejecutar la sintaxis presentada en la Figura 3.17.

```
yum install Kibana
```

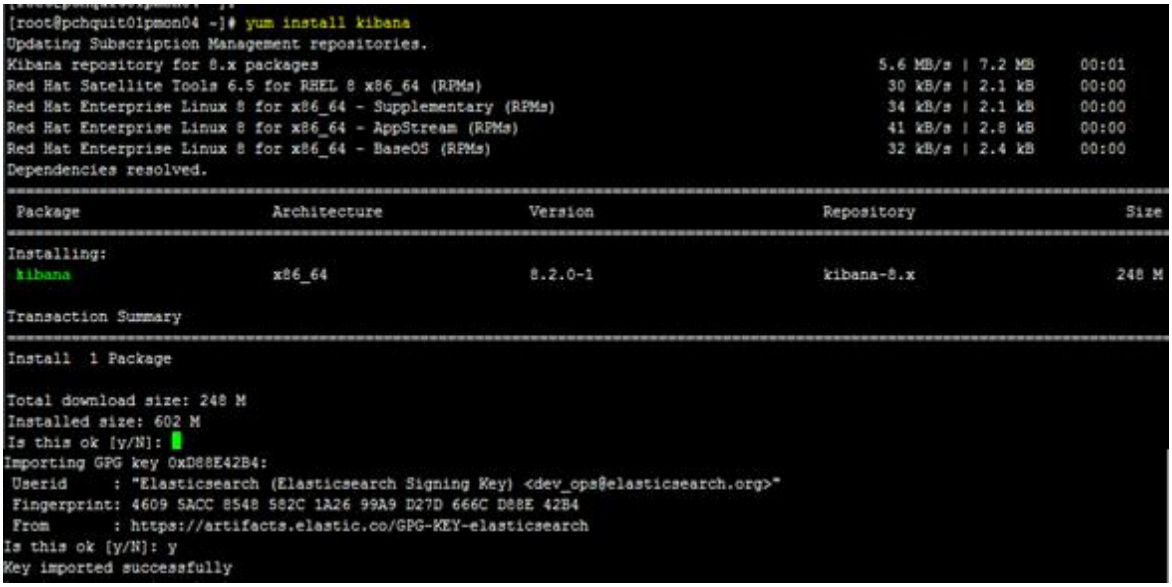


***Figura 3.17*** *Instalación de la herramienta Kibana*
*(Fuente: Consola de servidor principal)*

**Paso 4.** Edición del archivo de configuración de Kibana

Es recomienda modificar el archivo de configuración de la herramienta Kibana, con el editor *vim*, el mismo que se ubica en el directorio */etc/kibana/kibana.yml.*

*vim /etc/kibana/kibana.yml*

Continuando con la edición en el archivo de configuración, es necesario modificar los parámetros de la sección *"System: Kibana Server"*, en los campos: *network.host* y *http.port* como se presenta en la Figura 3.18, para configurar el puerto del servicio.

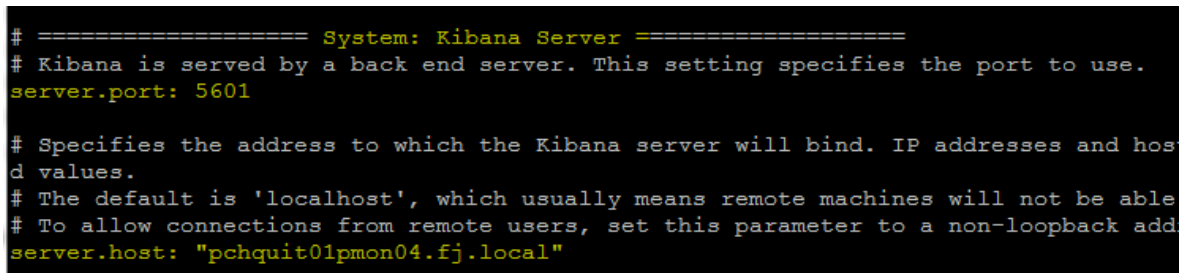


***Figura 3.18*** *Configuración del puerto para la herramienta Kibana*
*(Fuente: Autor)*

Además, se debe configurar la conexión con el servicio de Elasticsearch, para lo cual se debe editar en la sección *"System: Elasticsearch"* el campo: *elasticsearch.hosts*, como se indican en la Figura 3.19.

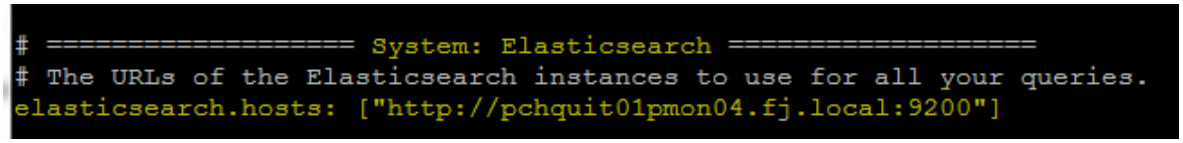


***Figura 3.19*** *Configuración de comunicación entre Kibana y Elasticsearch*
*(Fuente: Autor)*

**Paso 5.** Habilitar el servicio de Kibana

Es necesario configurar que el servicio de la herramienta Kibana inicie de manera automática cuando el servidor se esté encendiendo, para lo cual se debe ejecutar la sintaxis presentada en la Figura 3.20.

*systemctl enable kibana*

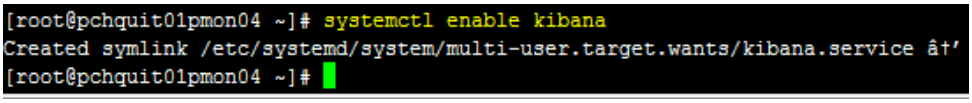


***Figura 3.20*** *Habilitar el servicio de la herramienta Kibana*
*(Fuente: Consola de servidor principal)*

**Paso 6.** Activación de puerto de la herramienta Kibana en firewall local

En la Figura 3.21 se puede observar la sintaxis para activar el puerto de la herramienta Kibana, este puerto garantiza la aceptación de peticiones de los equipos clientes.

*firewall-cmd --add-port=5601/tcp --permanent*

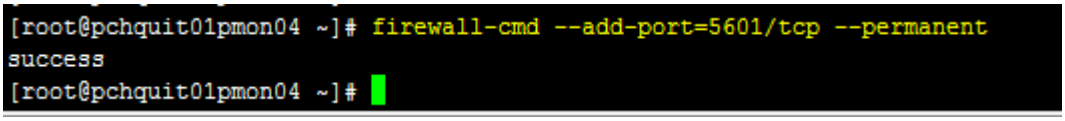


***Figura 3.21*** *Agregar el puerto 5601 al Firewall*
*(Fuente: Consola de servidor principal)*

**Paso 7.** Validación del servicio Kibana

Es necesario reiniciar el equipo, para validar el correcto funcionamiento del servicio de Kibana, tal como se indica en la Figura 3.22.

*systemctl status Kibana*

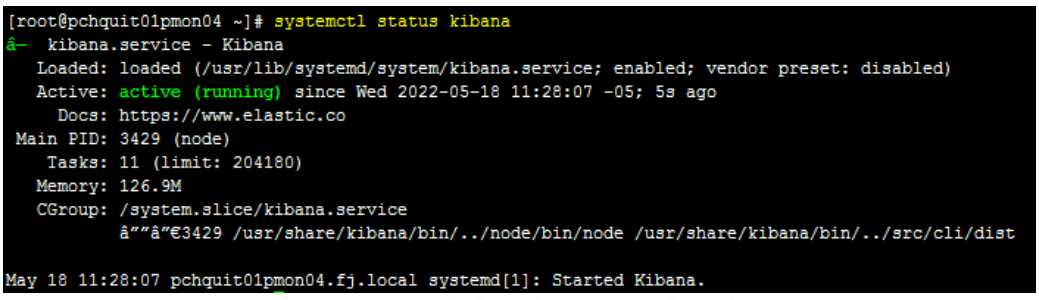


***Figura 3.22*** *Validar el servicio de Kibana*
*(Fuente: Consola de servidor principal)*

Además, es indispensable realizar la validación del servicio Kibana, utilizando un Navegador o Browser para corroborar la disponibilidad del aplicativo en la red, tal como se indica en la Figura 3.23.

*http://pchquit01pmon04.fj.local:5601*

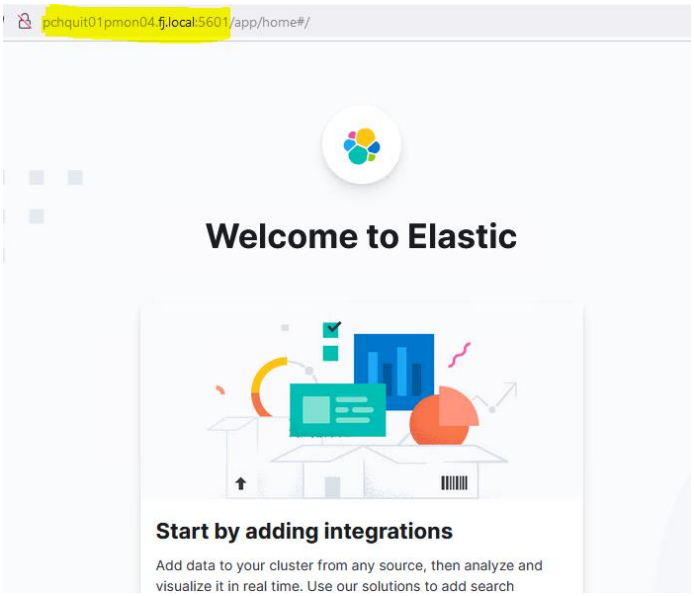


***Figura 3.23*** *Validación de la herramienta Kibana mediante un browser (Fuente: Consola de servidor principal)*

### 3.4.3 Logstash instalación y configuración en el servidor.

Para la instalación de la herramienta Logstash, se debe ingresar mediante una conexión SSH al servidor Elastic (PMON04), usando el usuario *root* y continuar con los siguientes pasos:

**Paso 1.** Instalación de la herramienta Logstash

Para proceder con la instalación de la herramienta Logstash, es necesario ejecutar la sintaxis presentada en la Figura 3.24.

```
yum install logstash
```

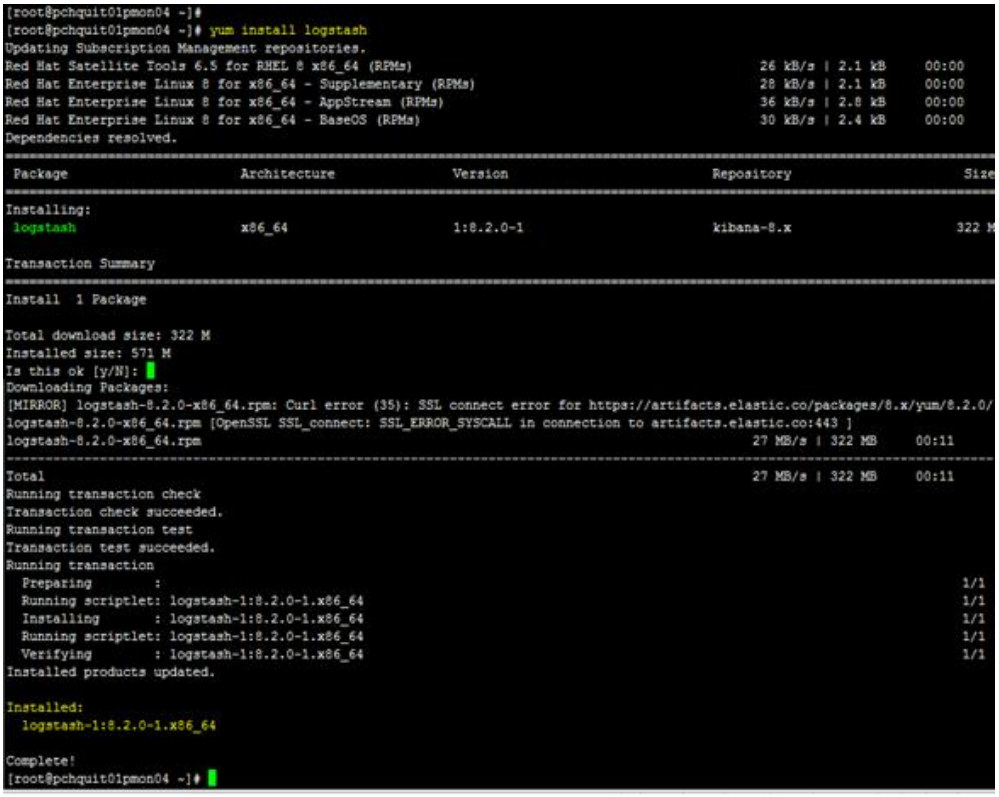


***Figura 3.24*** *Instalación de la herramienta Logstash*
*(Fuente: Consola de servidor principal)*

**Paso 2.** Edición del archivo de configuración de Logstash

Es necesario modificar el archivo de configuración de la herramienta Logstash, con el editor *vim*, el mismo que se ubica en el directorio */etc/logstash/logstash.yml*.

```
vim /etc/logstash/logstash.yml
```

Continuando con la edición en el archivo de configuración, es necesario modificar los parámetros de la sección *"API Settings"* en los campos: *api.http.host, api.http.port* como se presenta en la Figura 3.25, para configurar el puerto del servicio.

```
# ------------ API Settings -------------
# Define settings related to the HTTP API here.
#
# The HTTP API is enabled by default. It can be disab
# on it will not work as intended.
#
# api.enabled: true
#
# By default, the HTTP API is not secured and is ther
# host's loopback interface, ensuring that it is not
# the network.
# When secured with SSL and Basic Auth, the API is bo
# unless configured otherwise.
#
api.http.host: 0.0.0.0
#
# The HTTP API web server will listen on an available
# Values can be specified as a single port (e.g., `96
# of ports (e.g., `9600-9700`).
#
api.http.port: 9600-9700
#
```

***Figura 3.25*** *Edición de la configuración del servicio Logstash*
*(Fuente: Autor)*

**Paso 3.** Habilitar el servicio de Logstash

Es necesario configurar que el servicio de la herramienta Logstash inicie de manera automática cuando el servidor se esté encendiendo, para lo cual se debe ejecutar la sintaxis presentada en la Figura 3.26.

*systemctl enable logstash*

```
[root@pchquit01pmon04 ~]#
[root@pchquit01pmon04 ~]# systemctl enable logstash
Created symlink /etc/systemd/system/multi-user.target.wants/logstash.service â†'
[root@pchquit01pmon04 ~]#
```

***Figura 3.26*** *Habilitar el servicio Logstash*
*(Fuente: Consola de servidor principal)*

**Paso 4.** Activación de puerto de la herramienta Logstash

En la Figura 3.27 se puede observar la sintaxis para activar el puerto de la herramienta Logstash, este puerto garantiza la aceptación de peticiones de los equipos clientes.

*firewall-cmd --add-port=5044/tcp --permanent*

```
[root@pchquit01pmon04 ~]#
[root@pchquit01pmon04 ~]# firewall-cmd --add-port=5044/tcp --permanent
success
[root@pchquit01pmon04 ~]#
```

***Figura 3.27*** *Agregar el puerto 5044 al Firewall*
*(Fuente: Consola de servidor principal)*

**Paso 5.** Validación del servicio de Logstash

Es necesario reiniciar el equipo, para validar el correcto funcionamiento del servicio de Logstash, tal como se presenta en la Figura 3.28.

*systemctl status logstash*

```
[root@pchquit01pmon04 ~]# systemctl status logstash
â logstash.service - logstash
   Loaded: loaded (/usr/lib/systemd/system/logstash.service; enabled; vendor preset: disabled)
   Active: active (running) since Wed 2022-08-10 11:53:42 -05; 2 months 4 days ago
 Main PID: 1091 (java)
    Tasks: 60 (limit: 139671)
   Memory: 933.6M
   CGroup: /system.slice/logstash.service
           ââ1091 /usr/share/logstash/jdk/bin/java -Xms1g -Xmx1g -XX:+UseConcMarkSweepGC -XX:CMSInitia>

Oct 14 09:52:59 pchquit01pmon04.fj.local logstash[1091]: [2022-10-14T09:52:59,489][INFO ][logstash.out>
Oct 14 09:52:59 pchquit01pmon04.fj.local logstash[1091]: [2022-10-14T09:52:59,489][INFO ][logstash.out>
Oct 14 09:52:59 pchquit01pmon04.fj.local logstash[1091]: [2022-10-14T09:52:59,489][INFO ][logstash.out>
Oct 14 09:52:59 pchquit01pmon04.fj.local logstash[1091]: [2022-10-14T09:52:59,490][INFO ][logstash.out>
Oct 14 09:53:57 pchquit01pmon04.fj.local logstash[1091]: [2022-10-14T09:53:57,265][INFO ][logstash.out>
Oct 14 09:53:57 pchquit01pmon04.fj.local logstash[1091]: [2022-10-14T09:53:57,265][INFO ][logstash.out>
Oct 14 09:54:01 pchquit01pmon04.fj.local logstash[1091]: [2022-10-14T09:54:01,330][INFO ][logstash.out>
Oct 14 09:54:01 pchquit01pmon04.fj.local logstash[1091]: [2022-10-14T09:54:01,330][INFO ][logstash.out>
Oct 14 09:54:03 pchquit01pmon04.fj.local logstash[1091]: [2022-10-14T09:54:03,903][INFO ][logstash.out>
Oct 14 09:54:03 pchquit01pmon04.fj.local logstash[1091]: [2022-10-14T09:54:03,904][INFO ][logstash.out>
lines 1-19/19 (END)
```

***Figura 3.28*** *Validar el servicio de Logstash*
*(Fuente: Consola de servidor principal)*

**Paso 6.** Edición del archivo de conexión

En la Figura 3.29 se presenta la sintaxis utilizada, para abrir del archivo de conexión de la herramienta Logstash.

*vim /etc/logstash/conf.d/logstash.conf*

```
[root@pchquit01pmon04 ~]#
[root@pchquit01pmon04 ~]# vim /etc/logstash/conf.d/logstash.conf
[root@pchquit01pmon04 ~]#
```

***Figura 3.29*** *Abrir el archivo logstash.conf*
*(Fuente: Consola de servidor principal)*

Este repositorio tiene la funcionalidad de garantizar la conexión del servicio Logstash con la herramienta Elasticsearch. Para esto se coloca las siguientes líneas de código:

```
input {
 beats {
 port => 5044
 }
}
output {
 elasticsearch {
 hosts => ["pchquit01pmon04.fj.local:9200"]
 index => "%{[@metadata][beat]}-%{[@metadata][version]}-%{+YYYY.MM.dd}"
 }
}
```

**Paso 7.** Configuración de filtros

En la Figura 3.30 se presenta la sintaxis utilizada, para la edición del archivo de filtros, donde interactúa con la herramienta Elasticsearch.

```
vim /etc/logstash/conf.d/logstash-filter.conf
```

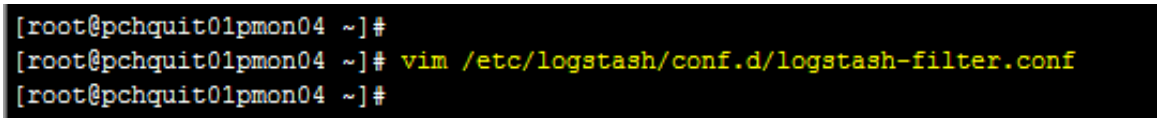


***Figura 3.30*** *Abrir el archivo Logstash-filter.conf*
*(Fuente: Consola de servidor principal)*

Este repositorio tiene la funcionalidad de garantizar la recepción de los datos de logs, que vienen de los equipos clientes y son emitidos a la herramienta Elasticsearch. Para esto se coloca las siguientes líneas de código:

```
input { stdin { } }
filter {
 grok {
 match => { "message" => "%{COMBINEDAPACHELOG}" }
 }
 date {
 match => [ "timestamp" , "dd/MMM/yyyy:HH:mm:ss Z" ]
 }
}
output {
 elasticsearch { hosts => ["pchquit01pmon04.fj.local:9200"] }
 stdout { codec => rubydebug }
}
```

**Paso 8.** Validación del servicio Logstash

Es necesario realizar la validación de conexiones al servicio de Logstash mediante el puerto, tal como se indica en la Figura 3.31.

*netstat -ant | grep 5044*

```
[root@pchquit01pmon04 ~]# netstat -ant | grep 5044
tcp        0      0 10.1.13.236:50448       10.1.13.236:9200        ESTABLISHED
tcp6       0      0 :::5044                 :::*                    LISTEN
tcp6       0      0 10.1.13.236:9200        10.1.13.236:50448       ESTABLISHED
tcp6       0      0 10.1.13.236:5044        10.1.13.204:36526       ESTABLISHED
```

***Figura 3.31*** *Comprobación de actividad del puerto en el servicio Logstash*
*(Fuente: Consola de servidor principal)*

### 3.4.4 Metricbeat instalación y configuración en el servidor.

Para la instalación del agente Metricbeat, se debe ingresar mediante una conexión SSH al servidor Elastic (PMON04), usando el usuario *root* y continuar con los siguientes pasos:

**Paso 1.** Descargar el instalador

Para la instalación del agente Metricbeat, se debe descargar el instalador, desde la página oficial de Elastic, como se indica en la Figura 3.32.

*curl -L -O https://artifacts.elastic.co/downloads/beats/metricbeat/metricbeat-8.2.0-x86_64.rpm*

```
[root@pchquit01pmon04 ~]# curl -L -O https://artifacts.elastic.co/downloads/beats/metricbeat/metricbeat-8.2.0-x86_64.rpm
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 69.8M  100 69.8M    0     0  24.8M      0  0:00:02  0:00:02 --:--:-- 24.8M
[root@pchquit01pmon04 ~]#
```

***Figura 3.32*** *Descarga del archivo Metricbeat en rpm*
*(Fuente: Consola de servidor principal)*

**Paso 2.** Instalación del agente Metricbeat

Para proceder con la instalación del agente Metricbeat, es necesario ejecutar la sintaxis presentada en la Figura 3.33.

*yum localinstall metricbeat-8.2.0-x86_64.rpm*

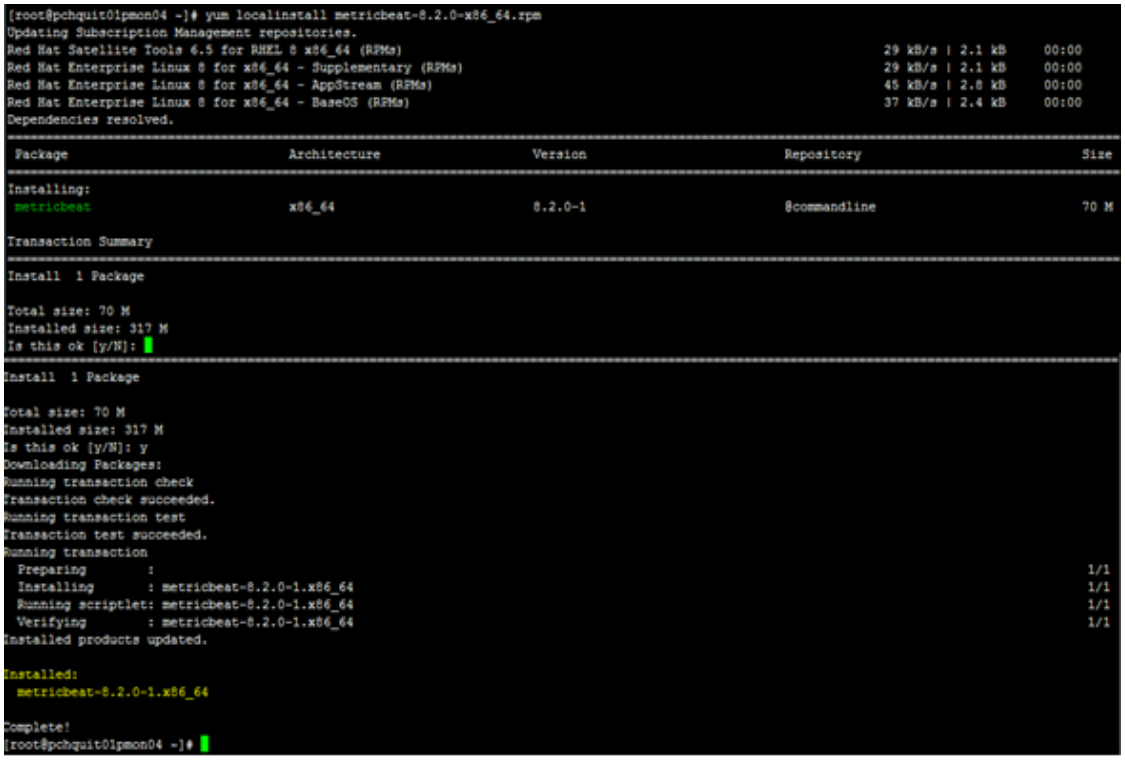


***Figura 3.33*** *Instalación del agente Metricbeat*
*(Fuente: Consola de servidor principal)*

**Paso 3.** Edición del archivo de configuración del agente Metricbeat

Es necesario modificar el archivo de configuración del agente Metricbeat, con el editor *vim*, el mismo que se encuentra en: */etc/metricbeat/metricbeat.yml*.

*vim /etc/elasticsearch/elasticsearch.yml*

En la Figura 3.34 se presenta los campos a modificarse en la sección *"Kibana"*, como son: *network.host* y *http.port*, de esta manera se garantiza la correcta comunicación con la herramienta Kibana.

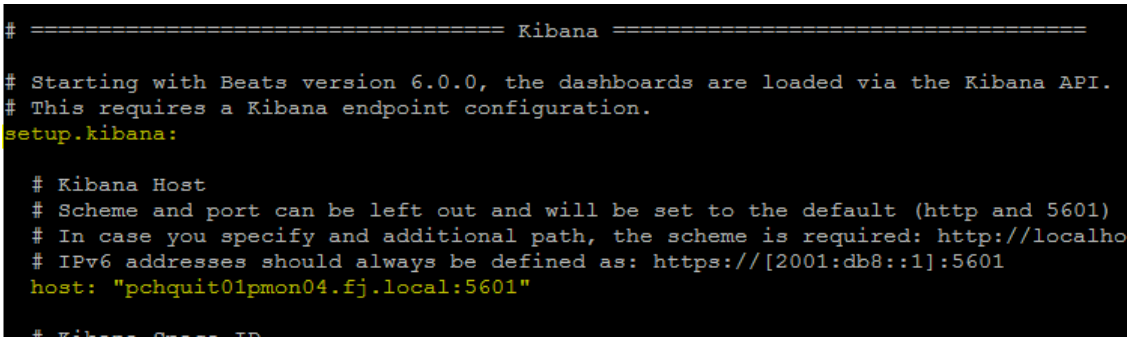


***Figura 3.34*** *Configuración del agente Metricbeat*
*(Fuente: Autor)*

**Paso 4.** Habilitación de módulos del agente Metricbeat.

Los beats o agentes tienen módulos propios para interactuar con la herramienta Kibana, para un mejor desempeño es necesario habilitar varios módulos, los mismos que se detallan más adelante.

**Paso 4.1** Habilitación del módulo *Elasticsearch*

En la Figura 3.35 se presenta la activación del módulo *elasticsearch*, mismo que permite tener una conexión de los datos con la herramienta Kibana.

*metricbeat modules enable Elasticsearch*

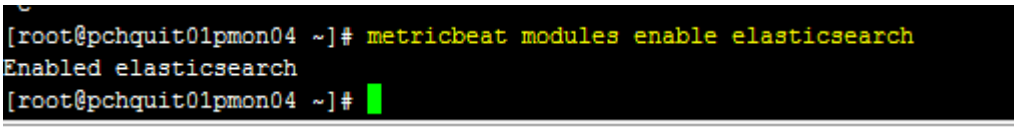


***Figura 3.35*** *Habilitar el módulo Elasticsearch del agente Metricbeat*
*(Fuente: Consola de servidor principal)*

**Paso 4.2** Habilitación del módulo *Windows*

En la Figura 3.36 se presenta la activación del módulo *Windows*, mismo que sirve para conexiones con equipos con sistema operativo Windows.

*metricbeat modules enable windows*

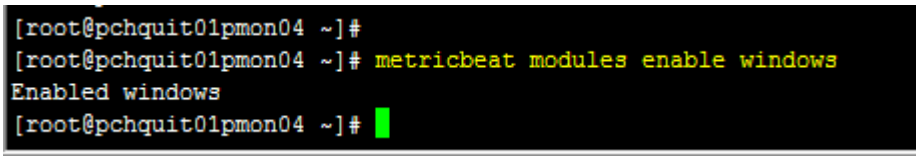


***Figura 3.36*** *Habilitar el módulo Windows del agente Metricbeat*
*(Fuente: Consola de servidor principal)*

**Paso 4.3** Habilitación del módulo elasticsearch-xpack

En la Figura 3.37 se presenta la activación del módulo *elasticsearch-xpack*, mismo que sirve para monitorear el mismo equipo.

*metricbeat modules enable elasticsearch-xpack*

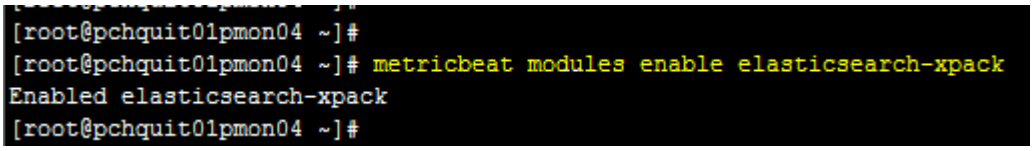


***Figura 3.37*** *Habilitar el módulo elasticsearch-xpack del agente Metricbeat*
*(Fuente: Consola de servidor principal)*

**Paso 4.4** Listado de módulos

En la Figura 3.41 se presenta los módulos habilitados del agente Metricbeat, con la siguiente sintaxis:

*metricbeat modules list*

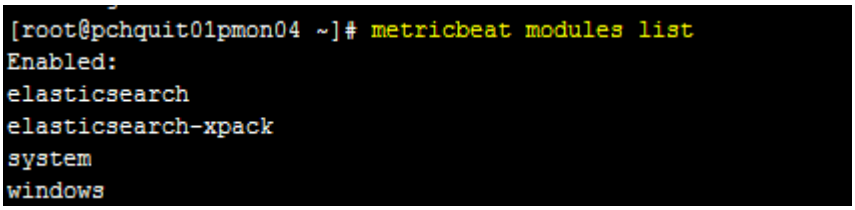


***Figura 3.38*** *Se listan los módulos activados del agente Metricbeat*
*(Fuente: Consola de servidor principal)*

**Paso 5.** Habilitar el servicio del agente Metricbeat

Es necesario configurar que el servicio del agente Metricbeat, inicie de manera automática cuando el servidor se esté encendiendo, para lo cual se debe ejecutar la sintaxis presentada en la Figura 3.39.

*systemctl enable metricbeat.service*

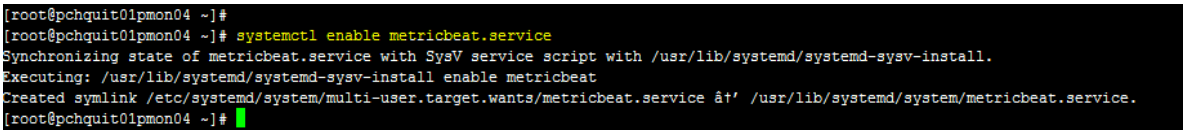


***Figura 3.39*** *Habilitar el servicio del agente Metricbeat*
*(Fuente: Consola de servidor principal)*

**Paso 6.** Validación del agente Metricbeat

Es necesario reiniciar el equipo, para validar el correcto funcionamiento del servicio de Metricbeat, tal como se indica en la Figura 3.40

*metricbeat setup –e*

```
[root@pchquit01pmon04 ~]# metricbeat setup -e
{"log.level":"info","@timestamp":"2022-10-14T09:56:27.419-0500","log.origin":
t.go","file.line":685},"message":"Home path: [/usr/share/metricbeat] Config p
a path: [/var/lib/metricbeat] Logs path: [/var/log/metricbeat]","service.name
n":"1.6.0"}
{"log.level":"info","@timestamp":"2022-10-14T09:56:27.419-0500","log.origin":
t.go","file.line":693},"message":"Beat ID: 283341dc-36c9-4559-b5d9-fcbf94cf83
beat","ecs.version":"1.6.0"}
```

***Figura 3.40*** *Validar el correcto funcionamiento del agente Metricbeat*
*(Fuente: Consola de servidor principal)*

### 3.4.5 Filebeat instalación y configuración en el servidor.

Para la instalación del agente Filebeat, se debe ingresar mediante una conexión SSH al servidor Elastic (PMON04), usando el usuario *root* y continuar con los siguientes pasos:

**Paso 1.** Descargar el instalador

Para la instalación del agente Filebeat, se debe descargar el instalador, desde la página oficial de Elastic, como se indica en la Figura 3.41.

*curl -L -O https://artifacts.elastic.co/downloads/beats/filebeat/filebeat-8.2.0-x86_64.rpm*

```
[root@pchquit01pmon04 ~]#
[root@pchquit01pmon04 ~]# curl -L -O https://artifacts.elastic.co/downloads/beats/filebeat/filebeat-8.2.0-x86_64.rpm
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 56.4M  100 56.4M    0     0  24.1M      0  0:00:02  0:00:02 --:--:-- 24.1M
[root@pchquit01pmon04 ~]# ll
```

***Figura 3.41*** *Descarga el instalador Filebeat en formato rpm*
*(Fuente: Consola de servidor principal)*

**Paso 2.** Instalación del agente Filebeat

Para proceder con la instalación del agente Filebeat, es necesario ejecutar la sintaxis presentada en la Figura 3.42.

*yum localinstall filebeat-8.2.0-x86_64.rpm*

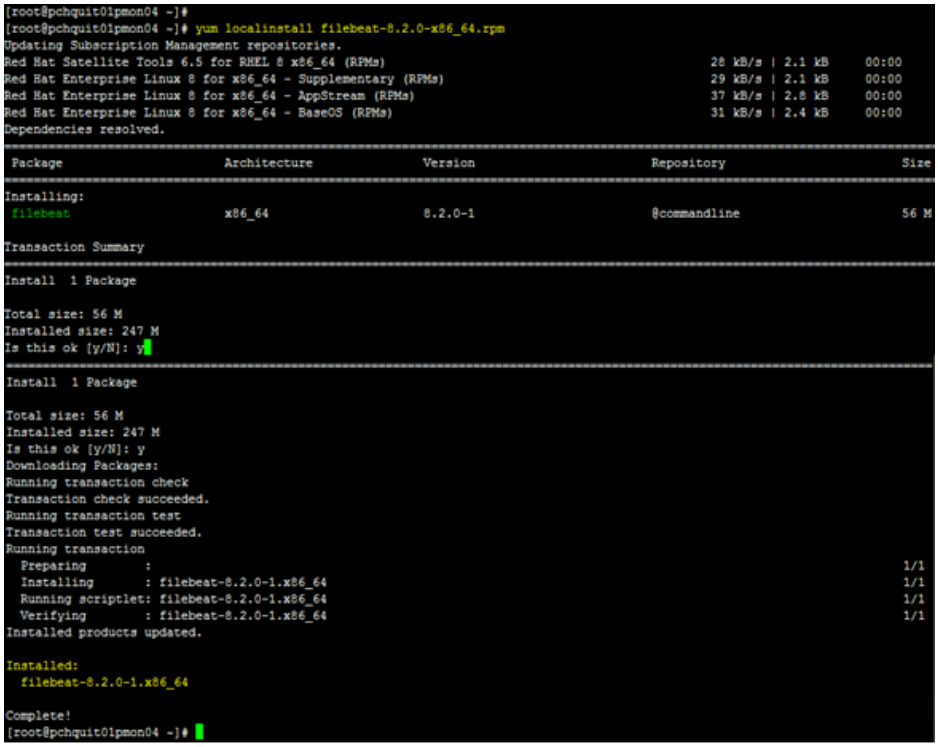


***Figura 3.42*** *Instalación del agente Filebeat*
*(Fuente: Consola de servidor principal)*

**Paso 3.** Edición del archivo de configuración del agente

Es necesario modificar el archivo de configuración del agente Filebeat, con el editor *vim*, el mismo que se ubica en el directorio */etc/metricbeat/metricbeat.yml*.

```
vim /etc/filebeat/filebeat.yml
```

En la Figura 3.43 se presenta los campos a modificarse en la sección *"Kibana"*, como son: *setup.kibana y host*, para una comunicación con la herramienta Kibana. Además, en la sección de *"Elasticsearch Output"* se edita el campo *"output.elasticsearch"* el que permite el reenvió de datos a la herramienta Elasticsearch.

```
# ================================== Kibana ===================================

# Starting with Beats version 6.0.0, the dashboards are loaded via the Kibana API.
# This requires a Kibana endpoint configuration.
setup.kibana:

  # Kibana Host
  # Scheme and port can be left out and will be set to the default (http and 5601)
  # In case you specify and additional path, the scheme is required: http://localho
  # IPv6 addresses should always be defined as: https://[2001:db8::1]:5601
  host: "pchquit01pmon04.fj.local:5601"



# ---------------------------- Elasticsearch Output ----------------------------
output.elasticsearch:
  # Array of hosts to connect to.
  hosts: ["pchquit01pmon04.fj.local:9200"]

  # Protocol - either `http` (default) or `https`.
  #protocol: "https"
```

***Figura 3.43*** *Edición en el módulo Filebeat para comunicación con las herramientas Kibana y Elasticsearch*
*(Fuente: Autor)*

**Paso 4.** Habilitación de módulos del agente Filebeat

Para obtener mejores prestaciones con la herramienta Kibana, es necesario habilitar varios módulos, los mismos que se detallan más adelante.

**Paso 4.1** Habilitación del módulo ***Elasticsearch***

En la Figura 3.44 se presenta la activación del módulo *elasticsearch*, mismo que permite tener una conexión de los datos con la herramienta Kibana.

*filebeat modules enable elasticsearch*

```
[root@pchquit01pmon04 ~]# filebeat modules enable elasticsearch
Enabled elasticsearch
[root@pchquit01pmon04 ~]#
```

***Figura 3.44*** *Habilitar el módulo Elasticsearch del agente Filebeat*
*(Fuente: Consola de servidor principal)*

**Paso 4.2** Habilitación del módulo *system*

En la Figura 3.45 se presenta la activación del módulo *system*, el que permite gestionar la recopilación de registros basados en Unix/Linux.

*filebeat modules enable system*

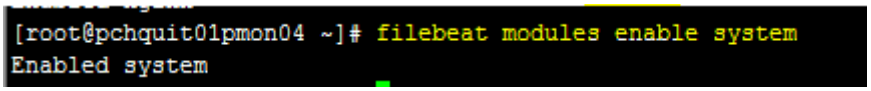


***Figura 3.45*** *Habilitar el módulo system del agente Filebeat*
*(Fuente: Consola de servidor principal)*

**Paso 4.3** Listado de módulos

En la Figura 3.46 se presenta los módulos habilitados del agente Filebeat, con la siguiente sintaxis:

*filebeat modules list*

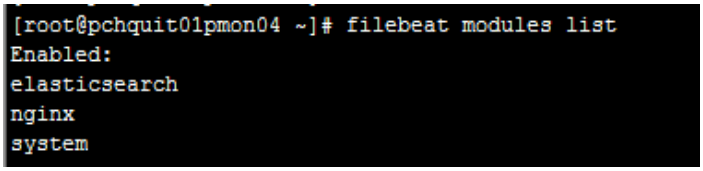


***Figura 3.46*** *Se listan los módulos activados del agente Filebeat*
*(Fuente: Consola de servidor principal)*

**Paso 5.** Habilitar el servicio del agente Filebeat

Es necesario configurar que el servicio del agente Filebeat, inicie de manera automática cuando el servidor se esté encendiendo, para lo cual se debe ejecutar la sintaxis presentada en la Figura 3.47.

*systemctl enable filebeat.service*

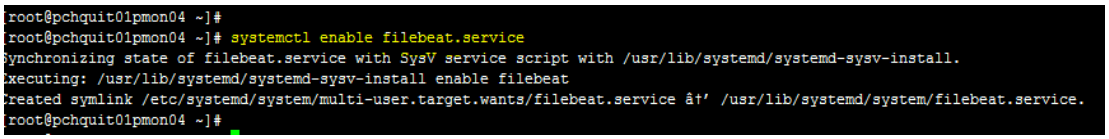


***Figura 3.47*** *Habilitar el servicio del agente Filebeat*
*(Fuente: Consola de servidor principal)*

**Paso 6.** Editar el archivo del módulo Elasticsearch

Es necesario modificar el archivo de configuración del módulo Elasticsearch, con el editor *vim*, el mismo que se ubica en el directorio */etc/filebeat/modules.d/elasticsearch.yml*,

*vim /etc/filebeat/modules.d/elasticsearch.yml*

Continuando con la edición del archivo, es necesario modificar los parámetros que se indican en la Figura 3.48.

```
- module: elasticsearch
  # Server log
  server:
    enabled: true

    var.paths:
      - /var/log/elasticsearch/*.log
      - /var/log/elasticsearch/*_server.json
```

***Figura 3.48*** *Configurar los paths donde se emitirán los eventos de logs (Fuente: Autor)*

**Paso 7.** Validación del agente Filebeat

Es necesario reiniciar el equipo, para validar el correcto funcionamiento del servicio de Filebeat, tal como se indica en la Figura 3.49

*filebeat setup –e*

```
[root@pchquit01pmon04 ~]#
[root@pchquit01pmon04 ~]# filebeat setup -e
{"log.level":"info","@timestamp":"2022-05-18T12:29:55.911-0500","log.origin":{"file.name":"instance/beat.go","file.line":685},"message":"Home path
: [/usr/share/filebeat] Config path: [/etc/filebeat] Data path: [/var/lib/filebeat] Logs path: [/var/log/filebeat]","service.name":"filebeat","ecs
.version":"1.6.0"}
{"log.level":"info","@timestamp":"2022-05-18T12:29:55.911-0500","log.origin":{"file.name":"instance/beat.go","file.line":693},"message":"Beat ID:
6e1836a2-958f-4e6e-ab59-03305cf6c613","service.name":"filebeat","ecs.version":"1.6.0"}
{"log.level":"warn","@timestamp":"2022-05-18T12:29:58.914-0500","log.logger":"add_cloud_metadata","log.origin":{"file.name":"add_cloud_metadata/pr
ovider_aws_ec2.go","file.line":80},"message":"read token request for getting IMDSv2 token returns empty: Put \"http://169.254.169.254/latest/api/t
oken\": context deadline exceeded (Client.Timeout exceeded while awaiting headers). No token in the metadata request will be used.","service.name"
:"filebeat","ecs.version":"1.6.0"}
```

***Figura 3.49*** *Validar el correcto funcionamiento del agente Filebeat (Fuente: Consola de servidor principal)*

## 3.5 Agentes para equipos clientes en Windows

Para instalar los agentes en equipos con sistema operativo Windows, se debe ingresar mediante una conexión de escritorio remoto, usando el usuario administrador local y seguir los pasos que se describen a continuación para cada agente.

### 3.5.1 Instalación y configuración del agente Heartbeat.

Para instalar el agente Heartbeat en un equipo con sistema operativo Windows, se debe realizar los siguientes pasos:

**Paso 1.** Descargar el instalador

Se debe descargar el instalador del agente Heartbeat, desde la página oficial de Elastic, como se indica en la Figura 3.50.

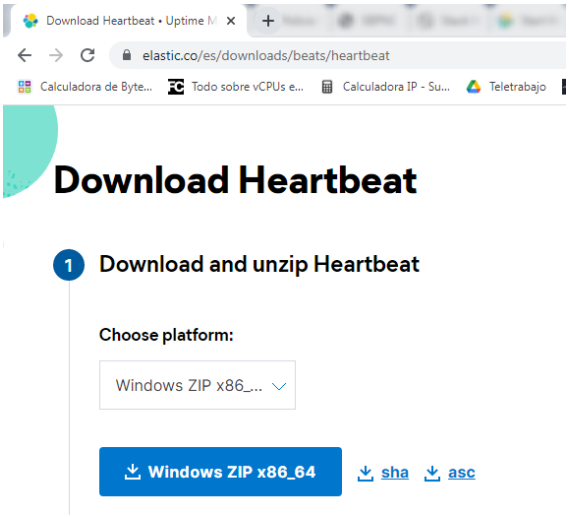


***Figura 3.50*** *Descargar el instalador del agente Heartbeat en formato ZIP*
*(Fuente: Sitio Web de Elastic)*

**Paso 2.** Descomprimir el instalador

Con el instalador descargado, se procede a descomprimir el archivo ZIP, para colocar la información en el directorio: *"C:\Program Files"*, y renombrar la carpeta y con: "***Heartbeat",*** tal como se indica en la Figura 3.51

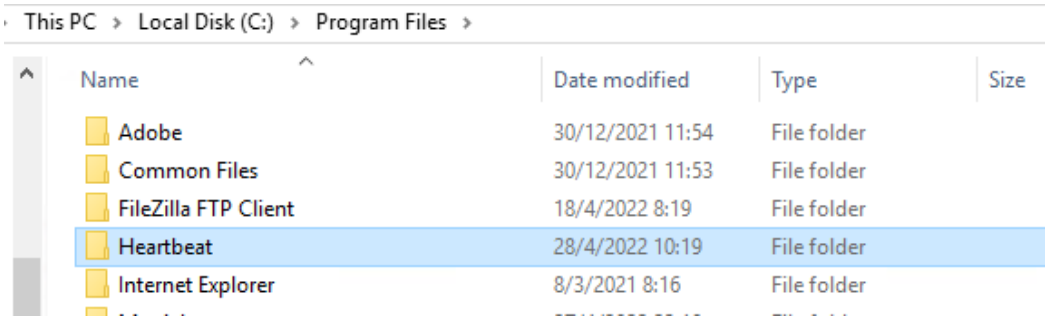


***Figura 3.51*** *Colocar los archivos instaladores del agente Heartbeat en la carpeta archivos de programa*
*(Fuente: Autor)*

**Paso 3.** Instalación del agente Heartbeat

Para proceder con la instalación del agente Heartbeat, es necesario ingresar en la consola del PowerShell[13], y ejecutar la sintaxis tal como se presenta en la Figura 3.52.

```
.\install-service-heartbeat.ps1
```

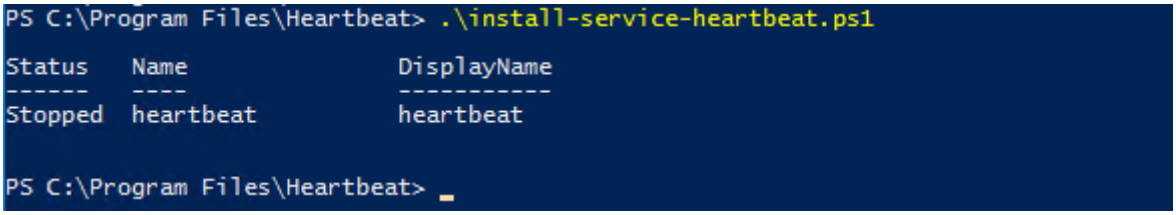


***Figura 3.52*** *Instalación del agente Heartbeat en Windows*
*(Fuente: Consola de un equipo cliente Windows)*

**Paso 4.** Edición del archivo de configuración del agente

Es necesario modificar el archivo de configuración *"heartbeat.yml"*, ubicado en el directorio *"C:\Program Files\Heartbeat"*. Al abrir el archivo se debe ubicar en la sección *"Hearbeat"* donde se podrán configurar los protocolos hacer monitoreados, tales como: icmp, tcp y http, como se presenta a continuación:

```
############################ Heartbeat ######################################

- type: icmp
  id: ping-myhost
  name: My Host Ping
  hosts: ["myhost"]
  schedule: '*/5 * * * * * *'

- type: tcp
  id: my-host-services
  name: My Host Services
  hosts: ["myhost"]
  ports: [80, 9200, 5044]
  schedule: '@every 5s'

- type: http
  id: myhost
  name: My HTTP Host
  schedule: '@every 5s'
  hosts: ["http://myhost:80"]
```

[13] Es una interfaz de comando, que facilita la ejecución de scripts o sentencias. (Soto, 2020)

Continuando con la edición en el archivo de configuración, es necesario modificar los parámetros de la sección *"Elasticsearch Output"*, en los campos: *output.elasticsearch* y *hosts,* tal como se presenta en la Figura 3.53, de esta manera se garantiza la conexión con la herramienta Elasticsearch.

```
# ================================ Outputs ================================

# Configure what output to use when sending the data collected by the beat.

# ---------------------------- Elasticsearch Output ----------------------------
output.elasticsearch:
  # Array of hosts to connect to.
  hosts: ["pchquit01pmon04.fj.local:9200"]

  # Protocol - either `http` (default) or `https`.
  #protocol: "https"
```

***Figura 3.53*** *Edición de la configuración para la salida de datos del agente Heartbeat (Fuente: Autor)*

**Paso 5.** Inicialización del servicio del agente Heartbeat

Es necesario iniciar el servicio del agente Heartbeat, para que empiece a monitorear el equipo, con sintaxis presentada en la Figura 3.54.

*Start-Service heartbeat*

```
PS C:\Program Files\Heartbeat> Start-Service heartbeat
PS C:\Program Files\Heartbeat>
```

***Figura 3.54*** *Iniciar el servicio del agente Heartbeat (Fuente: Consola de un equipo cliente Windows)*

**Paso 6.** Validación del agente Heartbeat

De manera previa se debe reiniciar el equipo, para proceder a validar el correcto funcionamiento del servicio Heartbeat en el panel de *"Servicios"* de Windows, tal como se indica en la Figura 3.55

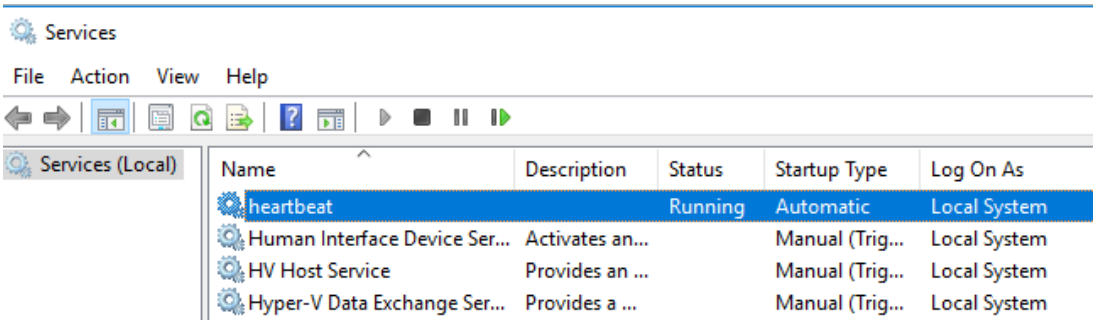


***Figura 3.55*** *Validar que el servicio Heartbeat este inicializado*
*(Fuente: Consola de un equipo cliente Windows)*

### 3.5.2 Instalación y configuración del agente Metricbeat

Para instalar el agente Metricbeat en un equipo con sistema operativo Windows, se debe realizar los siguientes pasos:

**Paso 1.** Descargar el instalador

Se debe descargar el instalador del agente Metricbeat, desde la página oficial de Elastic, como se indica en la Figura 3.56.

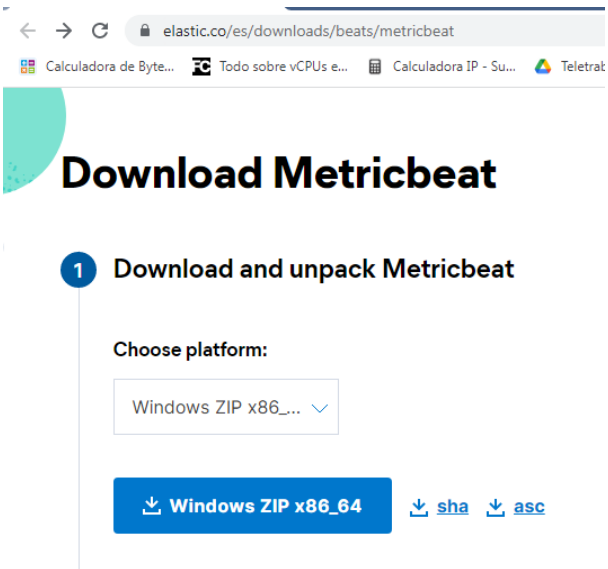


***Figura 3.56*** *Descargar el instalador del agente Metricbeat en formato ZIP*
*(Fuente: Sitio Web de Elastic)*

**Paso 2.** Descomprimir el instalador

Con el instalador descargado, se procede a descomprimir el archivo ZIP, para colocar la información en el directorio: *"C:\Program Files"*, y renombrar la carpeta con: "***Metricbeat",*** tal como se indica en la Figura 3.57

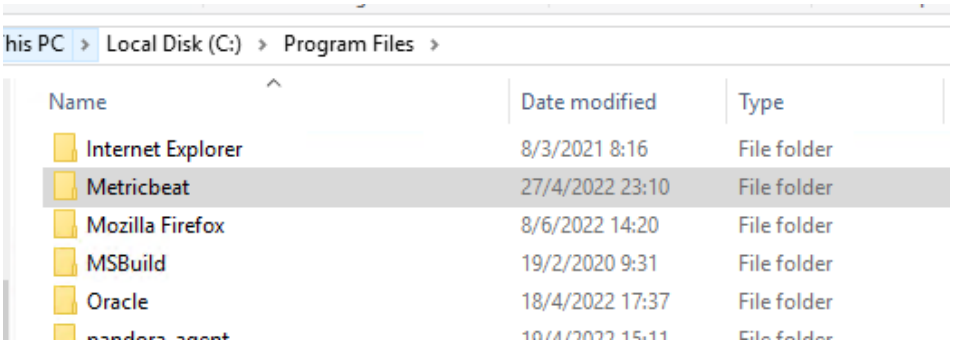


***Figura 3.57*** *Colocar los archivos instaladores del agente Metricbeat en la carpeta archivos de programa*
*(Fuente: Autor)*

**Paso 3.** Instalación del agente Metricbeat

Para proceder con la instalación del agente, es necesario ingresar en la consola del PowerShell, y ejecutar la sintaxis tal como se presenta en la Figura 3.58

```
.\install-service-metricbeat.ps1
```

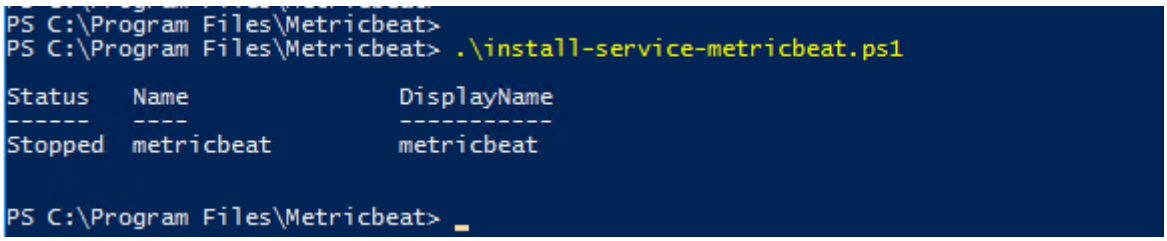


***Figura 3.58*** *Instalación del agente Metricbeat en Windows*
*(Fuente: Consola de un equipo cliente Windows)*

**Paso 4.** Edición del archivo de configuración del agente

Es necesario modificar el archivo de configuración *"metricbeat.yml"*, ubicado en el directorio *"C:\Program Files\Metricbeat"*. Al abrir el archivo se debe ubicar en la sección *"Kibana" y* editar los campos *setup.kibana y host*, para la comunicación con la herramienta Kibana, tal como se presenta en la Figura 3.59

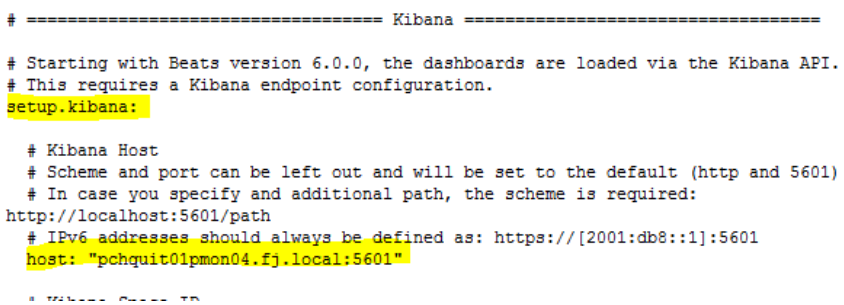


***Figura 3.59*** *Edición en la comunicación de Metricbeat con la herramienta Kibana (Fuente: Autor)*

Continuando con la edición en el archivo de configuración, es necesario modificar los parámetros de la sección *"Elasticsearch Output"*, en los campos: *output.elasticsearch* y *hosts,* tal como se presenta en la Figura 3.60, de esta manera se garantiza la conexión con la herramienta Elasticsearch.

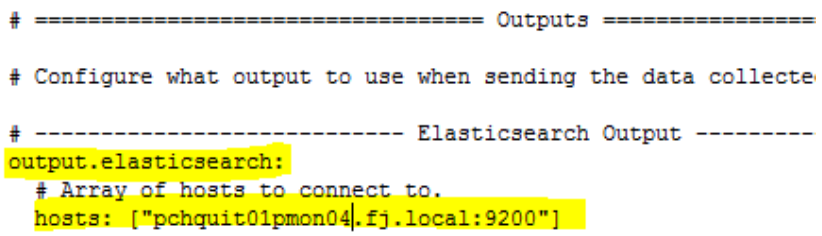


***Figura 3.60*** *Edición de la configuración para la salida de datos del agente Metricbeat (Fuente: Autor)*

**Paso 6** Inicialización del servicio

Es necesario iniciar el servicio del agente Heartbeat, para que empiece a monitorear el equipo, con sintaxis presentada en la Figura 3.61.

*Start-Service metricbeat*

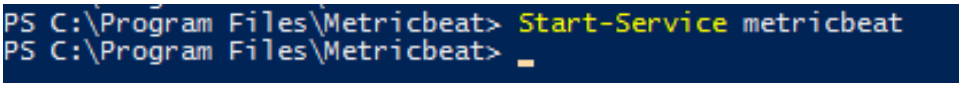


***Figura 3.61*** *Iniciar el servicio Metricbeat (Fuente: Consola de un equipo cliente Windows)*

**Paso 4.** Validación del agente Metricbeat

De manera previa se debe reiniciar el equipo, para validar el correcto funcionamiento del servicio Metricbeat en el panel de *"Servicios"* de Windows, tal como se indica en la Figura 3.62

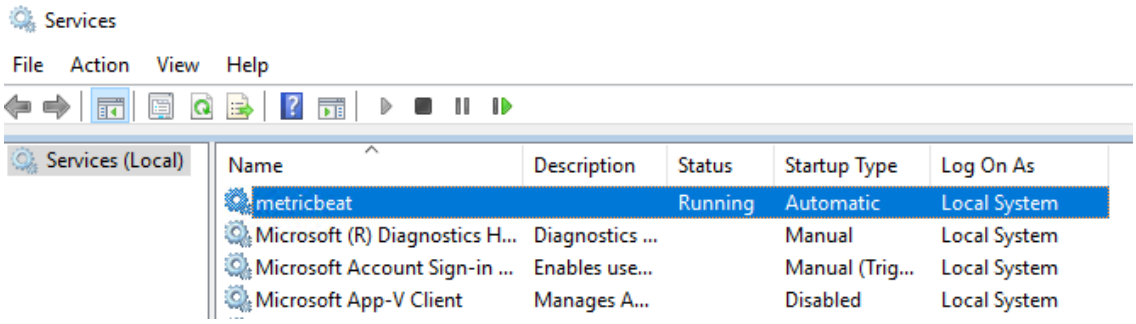


***Figura 3.62*** *Validar que el servicio Metricbeat este inicializado*
*(Fuente: Consola de un equipo cliente Windows)*

### 3.5.3 Instalación y configuración del agente Winlogbeat

Para instalar el agente Winlogbeat en un equipo con sistema operativo Windows, se debe realizar los siguientes pasos:

**Paso 1.** Descargar el instalador

Se debe descargar el instalador del agente Winlogbeat, desde la página oficial de Elastic, como se indica en la Figura 3.63.

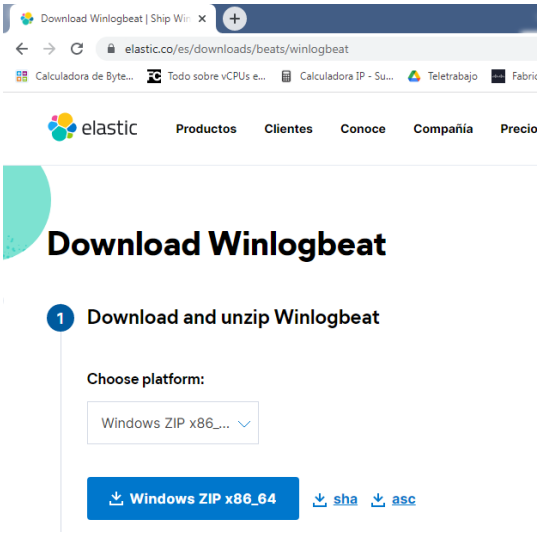


***Figura 3.63*** *Descargar el instalador del agente Winlogbeat en formato ZIP*
*(Fuente: Sitio Web de Elastic)*

**Paso 2.** Descomprimir el instalador

Con el instalador descargado, se procede a descomprimir el archivo ZIP, para colocar la información en el directorio: *"C:\Program Files"*, y renombrar la carpeta con: "***Winlogbeat",*** tal como se indica en la Figura 3.64.

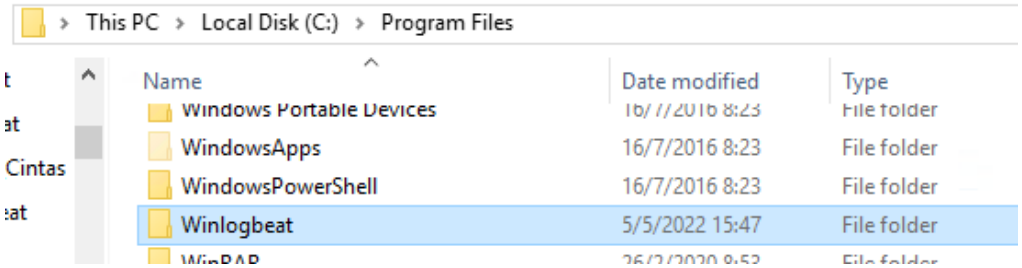


***Figura 3.64*** *Colocar los archivos instaladores del agente Winlogbeat en la carpeta archivos de programa*
*(Fuente: Autor)*

**Paso 3.** Instalación del agente

Para proceder con la instalación del agente, es necesario ingresar en la consola del PowerShell, y ejecutar la sintaxis tal como se presenta en la Figura 3.65

```
.\install-service-winlogbeat.ps1
```

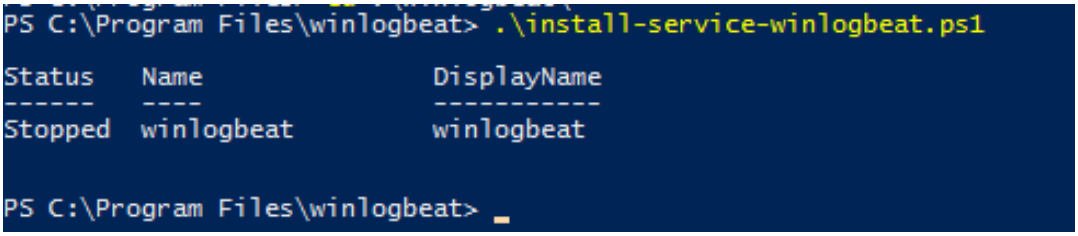


***Figura 3.65*** *Instalación del agente Winlogbeat en Windows*
*(Fuente: Consola de un equipo cliente Windows)*

**Paso 4.** Edición del archivo de configuración del agente

Es necesario modificar el archivo de configuración *"winlogbeat.yml"*, ubicado en el directorio *"C:\Program Files\ Winlogbeat"*. Al abrir el archivo ubicarse en la sección *""Winlogbeat.event_logs", y* configurar los parámetros hacer monitoreados referente a los eventos de Windows, tal como se presenta a continuación:

```
winlogbeat.event_logs:
 - name: Application
 - name: System
 - name: Security
 - name: Microsoft-Windows-Sysmon/Operational
 - name: Windows PowerShell
event_id: 400, 403, 600, 800
 - name: Microsoft-Windows-PowerShell/Operational
event_id: 4103, 4104, 4105, 4106
 - name: ForwardedEvents
```

Continuando con la edición en el archivo de configuración, es necesario modificar en la sesión de *"Kibana"* los campos *setup.kibana* y *host*, para la comunicación con la herramienta Kibana, tal como se presenta en la Figura 3.71

```
# ================================== Kibana ===================================

# Starting with Beats version 6.0.0, the dashboards are loaded via the Kibana API.
# This requires a Kibana endpoint configuration.
setup.kibana:

  # Kibana Host
  # Scheme and port can be left out and will be set to the default (http and 5601)
  # In case you specify and additional path, the scheme is required:
http://localhost:5601/path
  # IPv6 addresses should always be defined as: https://[2001:db8::1]:5601
  host: "pchquit01pmon04.fj.local:5601"
```

***Figura 3.66*** *Edición de la comunicación con la herramienta Kibana*
*(Fuente: Autor)*

Además en la sección *"Elasticsearch Output"*, editar los campos *output.elasticsearch* y *hosts*, tal como se presenta en la Figura 3.67, de esta manera se garantiza la conexión con la herramienta Elasticsearch.

```
# ================================== Outputs ===================

# Configure what output to use when sending the data collecte

# ------------------------------ Elasticsearch Output ----------
output.elasticsearch:
  # Array of hosts to connect to.
  hosts: ["pchquit01pmon04.fj.local:9200"]
```

***Figura 3.67*** *Edición de la configuración para la salida de datos del agente Winlogbeat (Fuente: Autor)*

**Paso 6.** Inicialización del servicio

Es necesario iniciar el servicio del agente Winlogbeat, para que empiece a monitorear el equipo, con sintaxis presentada en la Figura 3.68

*Start-Service winlogbeat*

```
PS C:\Program Files\Winlogbeat> Start-Service winlogbeat
PS C:\Program Files\Winlogbeat> _
```

***Figura 3.68*** *Iniciar el servicio del agente Winlogbeat (Fuente: Consola de un equipo cliente Windows)*

**Paso 4.** Validar servicio Winlogbeat

Es necesario de manera previa reiniciar el equipo, para validar que servicio Winlogbeat este ejecutándose en el panel de *"Servicios"* de Windows, tal como se indica en la Figura 3.69

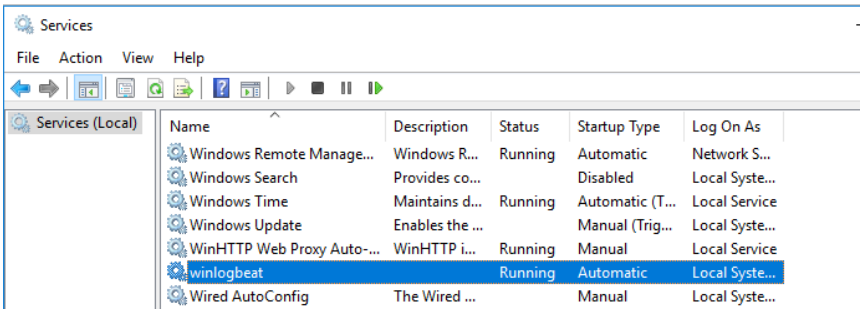


***Figura 3.69*** *Validar que el servicio Winlogbeat este inicializado (Fuente: Consola de un equipo cliente Windows)*

### 3.6 Agentes para equipos clientes en Linux

Para los equipos con sistema operativo Linux, se debe acceder mediante conexión SSH con el usuario root, para instalar los agentes, como se describe a continuación.

Para instalar los agentes en equipos con sistema operativo Linux, se debe acceder mediante conexión SSH con el usuario root y seguir los pasos que se describen a continuación por cada agente.

#### 3.6.1 Instalación y configuración del agente Heartbeat.

Para instalar el agente Heartbeat en un equipo con sistema operativo Linux, se debe realizar los siguientes pasos:

**Paso 1.** Descargar el instalador

Se debe descargar el instalador del agente Heartbeat, desde la página oficial de Elastic, para esto se ejecuta el comando de la Figura 3.70.

```
curl -L -O https://artifacts.elastic.co/downloads/beats/heartbeat/heartbeat-8.2.2-x86_64.rpm
```

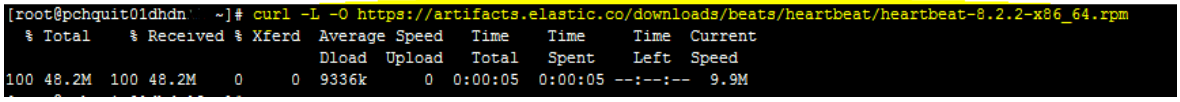


***Figura 3.70*** *Descargar el instalador del agente Heartbeat para Linux*
*(Fuente: Consola del equipo cliente)*

**Paso 2.** Instalación del agente Heartbeat

Para proceder con la instalación del agente Heartbeat, es necesario ejecutar la sintaxis presentada en la Figura 3.71.

```
yum localinstall heartbeat-8.2.2-x86_64.rpm
```

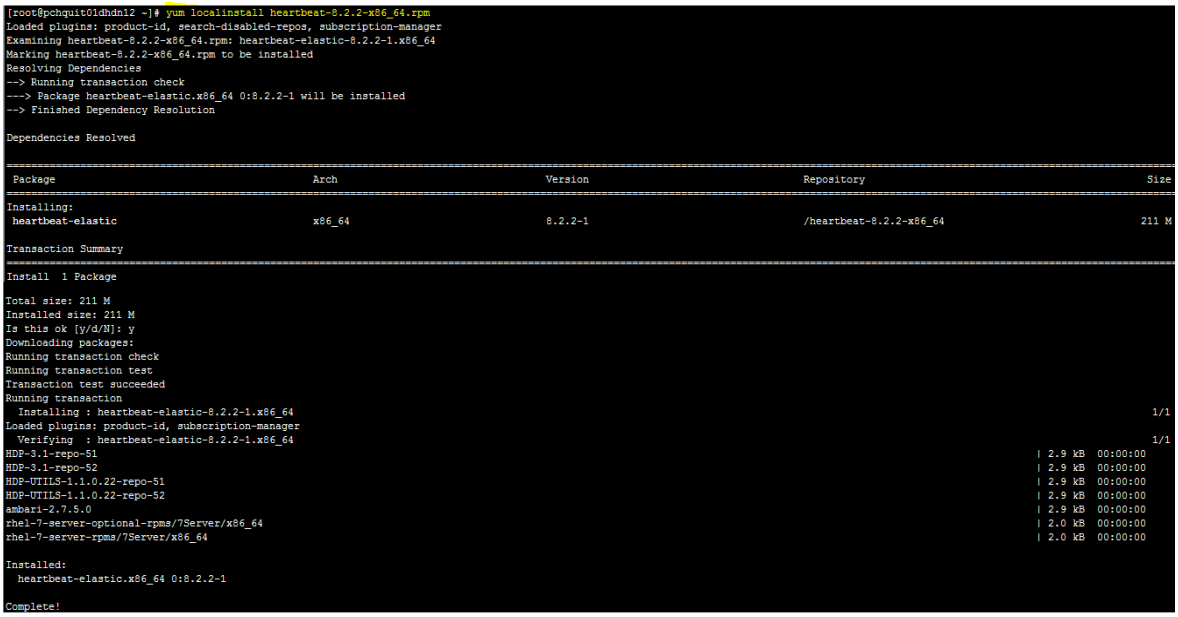


***Figura 3.71*** *Instalar el agente Heartbeat en Linux*
*(Fuente: Consola del equipo cliente)*

**Paso 3.** Edición del archivo de configuración del agente

Es necesario abrir el archivo de configuración del agente Heartbeat, con el editor *vim*, tal como se indica en la Figura 3.72

```
vim /etc/heartbeat/heartbeat.yml
```

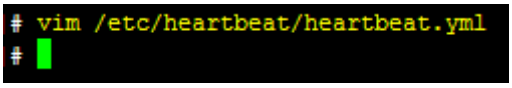


***Figura 3.72*** *Abrir el archivo de configuración Heartbeat.yml*
*(Fuente: Consola del equipo cliente)*

Continuando con la edición del archivo, ubicare en la sección *"Heartbeat"*, de esta manera editar los parámetros hacer monitoreados en referencia a los protocolos tales como: icmp, tcp y http, como se presenta a continuación:

```
############################## Heartbeat ##################################

- type: icmp
 id: ping-myhost
 name: My Host Ping
 hosts: ["myhost"]
 schedule: '*/5 * * * * * *'

- type: tcp
```

```
id: my-host-services
name: My Host Services
hosts: ["myhost"]
ports: [80, 9200, 5044]
schedule: '@every 5s'

- type: http
id: myhost
name: My HTTP Host
schedule: '@every 5s'
hosts: ["http://myhost:80"]
```

Además, en la sección *"Kibana"*, editar los campos: *setup.kibana* y *host*, tal como se presenta en la Figura 3.73, de esta manera se garantiza la conexión con la herramienta Kibana.

```
# ================================== Kibana ===================================

# Starting with Beats version 6.0.0, the dashboards are loaded via the Kibana API.
# This requires a Kibana endpoint configuration.
setup.kibana:

  # Kibana Host
  # Scheme and port can be left out and will be set to the default (http and 5601)
  # In case you specify and additional path, the scheme is required: http://localhost
  # IPv6 addresses should always be defined as: https://[2001:db8::1]:5601
  host: "pchquit01pmon04.fj.local:5601"
```

***Figura 3.73*** *Editar la configuración del agente Heartbeat con la herramienta Kibana*
*(Fuente: Autor)*

Es necesario modificar los parámetros de la sección *"Elasticsearch Output"*, en los campos: *output.elasticsearch* y *hosts*, tal como se presenta en la Figura 3.74, de esta manera se garantiza la conexión con la herramienta Elasticsearch.

```
# ---------------------------- Elasticsearch Output ----------------------------
output.elasticsearch:
  # Array of hosts to connect to.
  hosts: ["pchquit01pmon04.fj.local:9200"]

  # Protocol - either `http` (default) or `https`.
  #protocol: "https"
```

***Figura 3.74*** *Configuración del agente Heartbeat para comunicarse con la herramienta Elasticsearch*
*(Fuente: Autor)*

**Paso 4.** Inicialización del servicio del agente Heartbeat

Es necesario iniciar el servicio del agente Heartbeat, para que empiece a monitorear el equipo, con sintaxis presentada en la Figura 3.75.

*systemctl start heartbeat-elastic.service*

```
dweb01 ~]# systemctl start heartbeat-elastic.service
dweb01 ~]#
```

***Figura 3.75*** *Iniciar el servicio del agente Heartbeat*
*(Fuente: Consola del equipo cliente)*

### 3.6.2 Instalación y configuración del agente Metricbeat

Para instalar el agente Metricbeat en un equipo con sistema operativo Linux, se debe realizar los siguientes pasos:

**Paso 1.** Descargar el instalador

Se debe descargar el instalador del agente Metricbeat, desde la página oficial de Elastic, para esto se ejecuta el comando de la Figura 3.76.

*curl -L -O https://artifacts.elastic.co/downloads/beats/metricbeat/metricbeat-8.2.0-x86_64.rpm*

```
[root@pchquit01dhdn  ~]# curl -L -O https://artifacts.elastic.co/downloads/beats/metricbeat/metricbeat-8.2.0-x86_64.rpm
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 69.8M  100 69.8M    0     0  22.4M      0  0:00:03  0:00:03 --:--:-- 22.4M
```

***Figura 3.76*** *Descargar el instalador del agente Metricbeat para Linux*
*(Fuente: Consola del equipo cliente)*

**Paso 2.** Instalación del agente Metricbeat

Para proceder con la instalación del agente Metricbeat, es necesario ejecutar la sintaxis presentada en la Figura 3.77.

*yum localinstall metricbeat-8.2.0-x86_64.rpm*

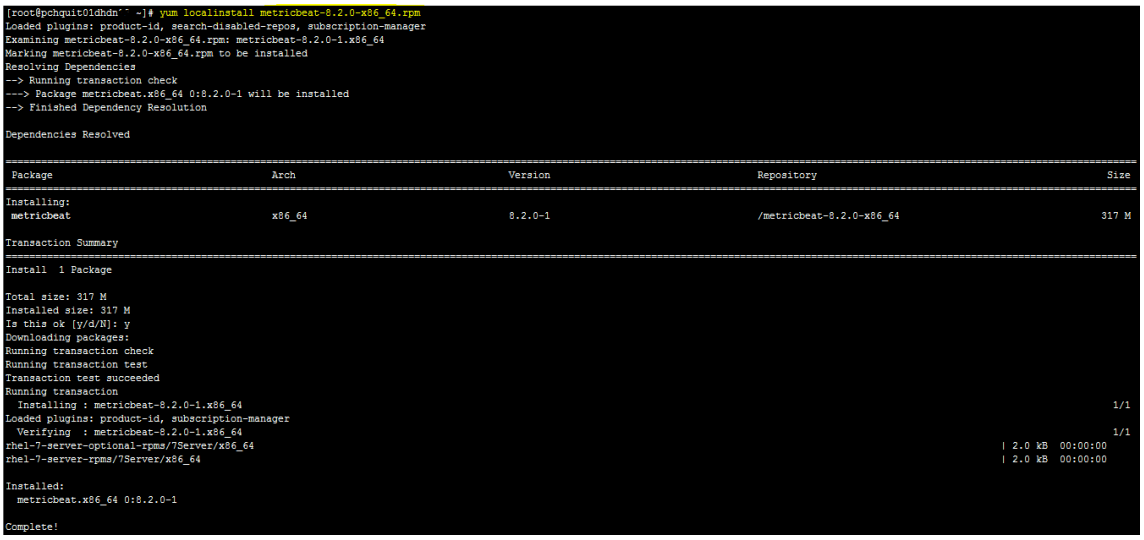


***Figura 3.77*** *Instalar el agente Metricbeat en Linux*
*(Fuente: Consola del equipo cliente)*

**Paso 3.** Edición del archivo de configuración del agente

Es necesario abrir el archivo de configuración del agente Metricbeat, con el editor *vim*, tal como se indica en la Figura 3.78

```
vim /etc/metricbeat/metricbeat.yml
```

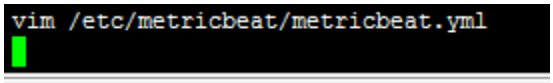


***Figura 3.78*** *Abrir el archivo de configuración Metricbeat.yml*
*(Fuente: Consola del equipo cliente)*

Además, en la sección *"Kibana"*, editar los campos: *setup.kibana* y *host*, tal como se presenta en la Figura 3.79, de esta manera se garantiza la conexión con la herramienta Kibana.

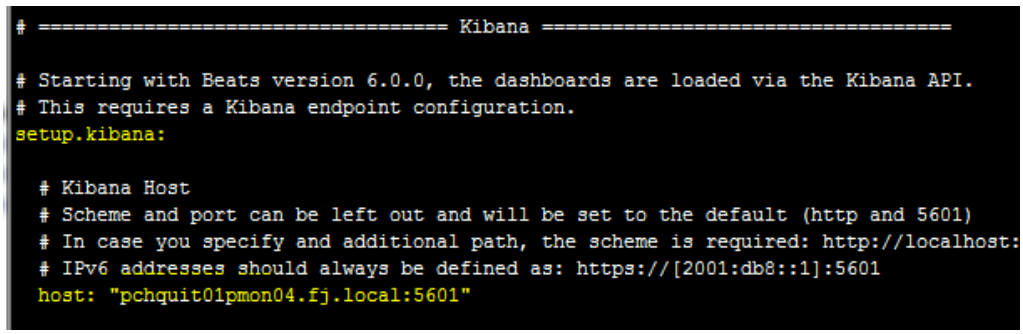


***Figura 3.79*** *Editar la configuración del agente Metricbeat con la herramienta Kibana*
*(Fuente: Autor)*

Además, modificar los parámetros de la sección *"Elasticsearch Output"*, en los campos: *output.elasticsearch* y *hosts*, tal como se presenta en la Figura 3.80, de esta manera se garantiza la conexión con la herramienta Elasticsearch.

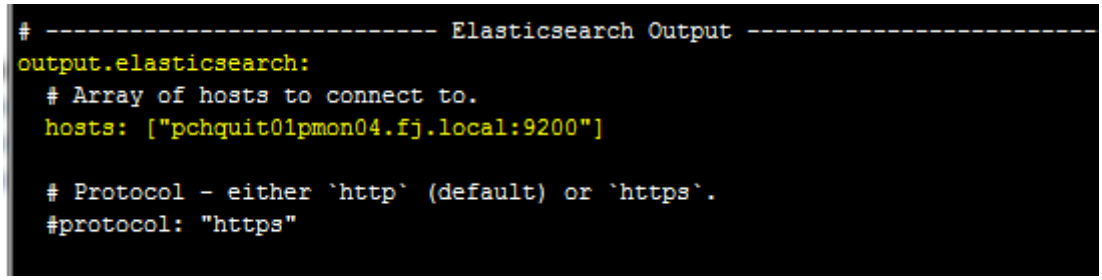


***Figura 3.80*** *Configuración del agente Metricbeat con la herramienta Elasticsearch (Fuente: Autor)*

**Paso 4.** Inicialización del servicio del agente

Es necesario iniciar el servicio del agente Metricbeat, para que empiece a monitorear el equipo, con sintaxis presentada en la Figura 3.81.

*systemctl start metricbeat.servicebeat*

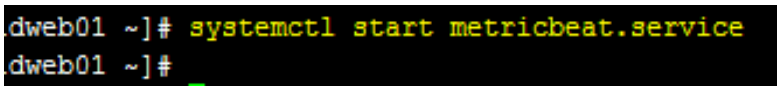


***Figura 3.81*** *Iniciar el servicio del agente Metricbeat (Fuente: Consola del equipo cliente)*

### 3.6.3 Instalación y configuración del agente Filebeat

Para instalar el agente Filebeat en un equipo con sistema operativo Linux, se debe realizar los siguientes pasos:

**Paso 1.** Descargar el instalador

Se debe descargar el instalador del agente Filebeat, desde la página oficial de Elastic, para esto se ejecuta el comando de la Figura 3.82.

*curl -L -O https://artifacts.elastic.co/downloads/beats/filebeat/filebeat-8.2.0-x86_64.rpm*


***Figura 3.82*** *Descarga el instalador del agente Filebeat para Linux*
*(Fuente: Consola del equipo cliente)*

**Paso 2.** Instalación del agente Filebeat

Para proceder con la instalación del agente Filebeat, es necesario ejecutar la sintaxis que se indica en la Figura 3.83.

*yum localinstall filebeat-8.2.0-x86_64.rpm*

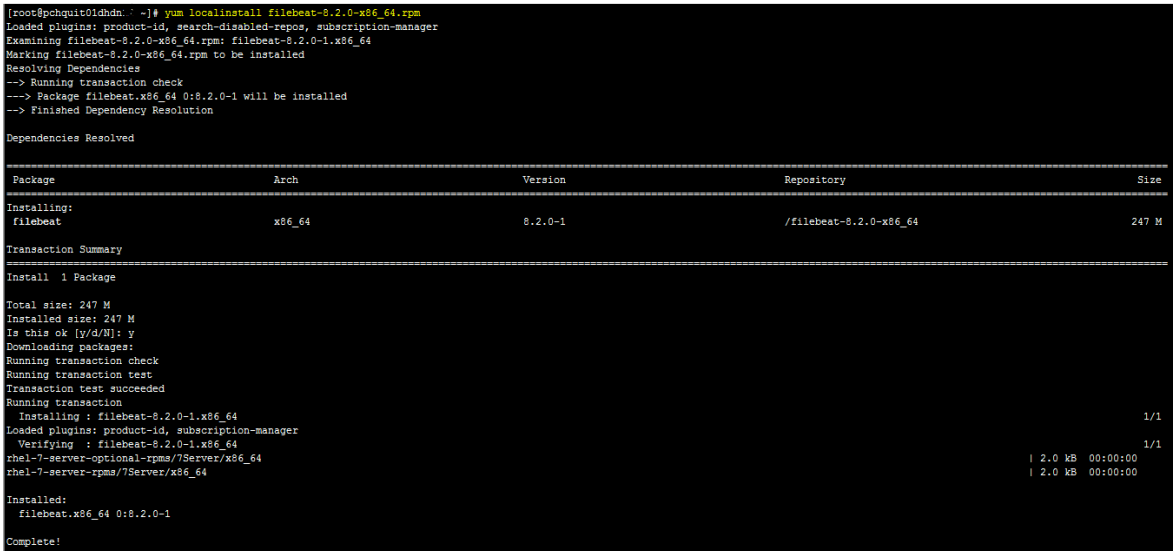


***Figura 3.83*** *Instalar el agente Filebeat en Linux*
*(Fuente: Consola del equipo cliente)*

**Paso 3.** Edición del archivo de configuración del agente

Es necesario abrir el archivo de configuración del agente Filebeat, con el editor *vim*, tal como se indica en la Figura 3.84

*vim /etc/filebeat/filebeat.yml*

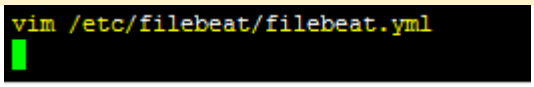


***Figura 3.84*** *Abrir el archivo de configuración filebeat.yml*
*(Fuente: Consola del equipo cliente)*

Es necesario modificar en la sección *"type:log"* los paths hacer de los logs que van hacer monitoreados, como se presenta a continuación:

```
- type: log
 id: my-filestream-idcnj
 enabled: true
paths:
 - /var/log/messages
 - /var/log/secure
```

Continuando con la edición en el archivo de configuración, es necesario modificar los parámetros en la sección *"System: Kibana Server"*, en los campos: *network.host* y *http.port*, para la comunicación con la herramienta Kibana, tal como se presenta en la Figura 3.85.

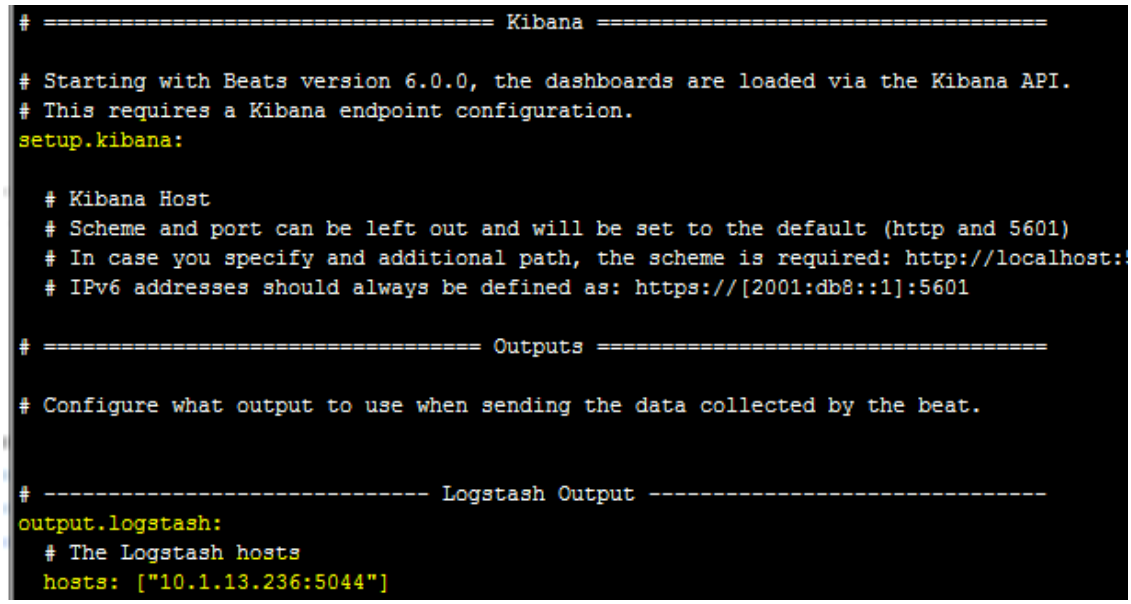


***Figura 3.85*** *Editar la configuración del agente Filebeat con la herramienta Logstash (Fuente: Autor)*

**Paso 4.** Inicialización del servicio del agente

Es necesario iniciar el servicio del agente Winlogbeat, para que empiece a monitorear el equipo, con sintaxis presentada en la Figura 3.86.

```
systemctl start filebeat.service
```

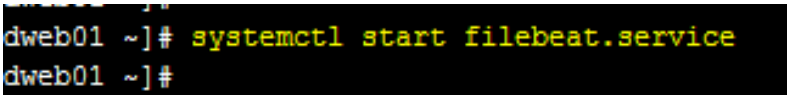


***Figura 3.86*** *Iniciar el servicio del agente Filebeat (Fuente: Consola del equipo cliente)*

# CAPÍTULO IV: RESULTADOS

En este capítulo se presenta y se analizan los datos obtenidos en la implementación del escenario de pruebas con la herramienta de monitoreo Elastic. Se procesa la información recolectada por cada uno de los diferentes agentes referente a: métricas de recursos, disponibilidad de servicios y eventos de logs en equipos Windows y Linux. Con la creación de diferentes Dashboard o Tableros de Visualización, para presentar de manera gráfica los resultados de monitoreo y faciliten la toma de decisiones en la optimización de los recursos tecnológicos y protección de datos.

## 4.1 Monitoreo Recursos del servidor.

Para el monitoreo de los recursos en equipos clientes con sistema operativo Windows y Linux, se instaló el agente Metricbeat, según se describe en las secciones 3.5.2 y 3.6.2 de este trabajo. Este agente permite obtener métricas de información referente a: procesamiento, memoria RAM, red de los equipos monitoreados.

En la Figura 4.1 se presenta un listado de equipos monitoreados, con la métrica de memoria RAM.

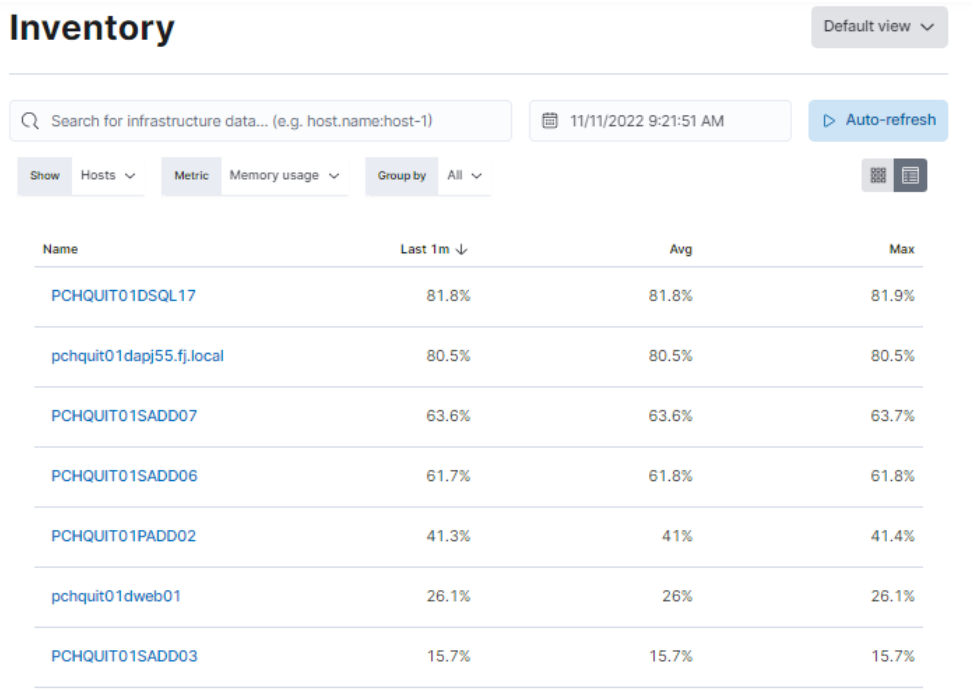


***Figura 4.1*** *Inventario de los equipos monitoreados con el agente Metricbeat (Fuente: Captura de la consola Elastic implementada)*

### 4.1.1 Visualización de recursos monitoreados.

Con la información que emite el agente Metricbeat a la herramienta Elasticsearch, se crean visualizaciones gráficas utilizando la herramienta Kibana referente a: uso del procesamiento, uso de memoria RAM, consumo del disco duro, identificación de procesos en el consumo del procesamiento, para unificarlas en un Dashboard, según la necesidad del administrador. En la Figura 4.2 se presenta la visualización referente al consumo en porcentaje de la memoria RAM de un equipo monitoreado, de forma de un medidor.

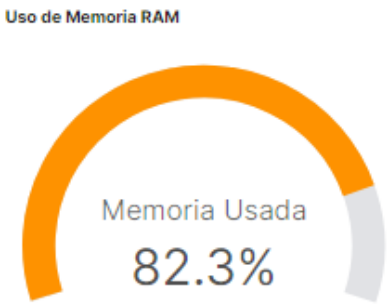


**Figura 4.2** Visualización uso en porcentaje de Memoria RAM. *(Fuente: Captura de la consola Elastic implementada)*

En la Figura 4.3(a) se presenta un Dashboard de monitoreo en métricas de recursos, sobre el consumo de recursos asignado a un equipo. Esto facilita al administrador de tecnología a tener un reporte gráfico sobre el comportamiento del equipo, el mismo que permite tomar correctivos para evitar perdida de servicio o saturación, las visualizaciones utilizadas para este Dashboard son las siguientes:

- Porcentaje del uso de CPU, mediante un medidor.
- Porcentaje del uso de memoria RAM, a través de un medidor.
- Porcentaje del uso de disco, en forma de un medidor.
- Consumo del CPU e identificación de procesos, mediante un gráfico de área.

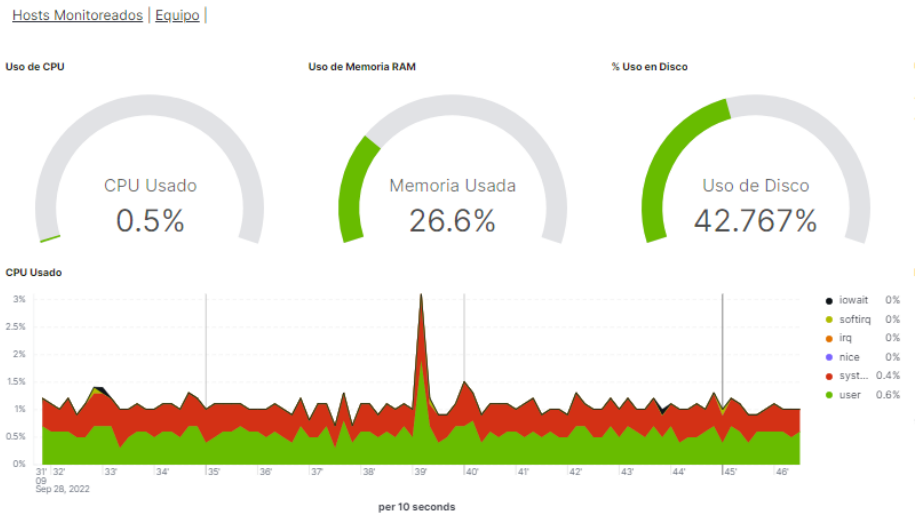


***Figura 4.3(a)*** *Dashboard de monitoreo de métricas del equipamiento tecnológico en Elastic (Fuente: Captura de la consola Elastic implementada)*

Además, en la Figura 4.4(b) se continua con las visualizaciones que tiene el Dashboard de monitoreo en métricas de recursos, con las siguientes visualizaciones:

- Porcentaje de consumo de disco duro por partición, con un gráfico de barras.
- Memoria RAM usada y libre, a través de un gráfico de barras.

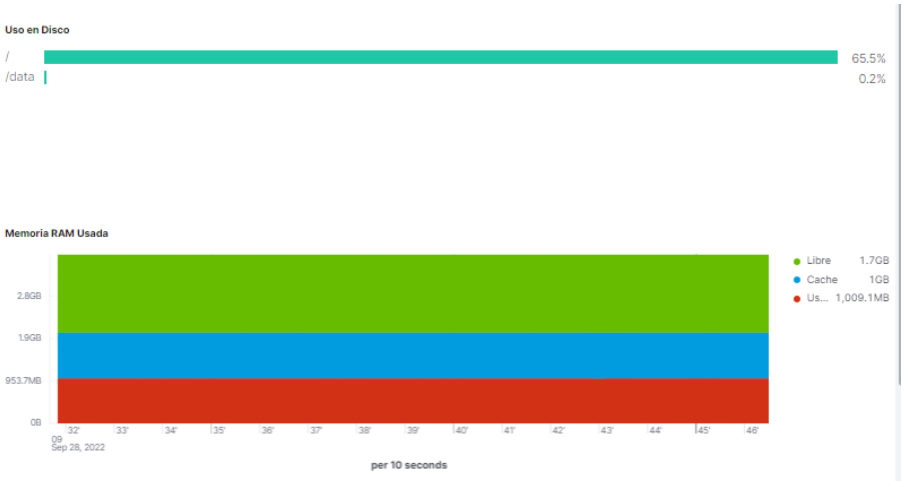


***Figura 4.4(b)****Dashboard de un equipo Linux monitoreado.*
*(Fuente: Captura de la consola Elastic implementada)*

## 4.2 Monitoreo de servicios.

Para el monitoreo de servicios en equipos clientes con sistema operativo Windows y Linux, se instaló el agente Heartbeat, según lo descrito en las secciones 3.5.1 y 3.6.1. Este agente permite verificar de manera periódica el estado de servicios como: ICMP, puertos o URL. La información recolectada puede ser visualizada en la herramienta Kibana. (Hershkovitch, 2018)

### 4.2.1 Monitoreo mediante ICMP.

El protocolo de control de mensajes de Internet o ICMP, se utiliza para diagnosticar errores en la comunicación de la red y alertar sobre un fallo de disponibilidad de servicio, en el equipo.

En la Figura 4.5 se presenta el módulo de monitoreo del servicio ICMP, en el cual se puede visualizar la disponibilidad del servicio y determinar si el mismo está disponible.

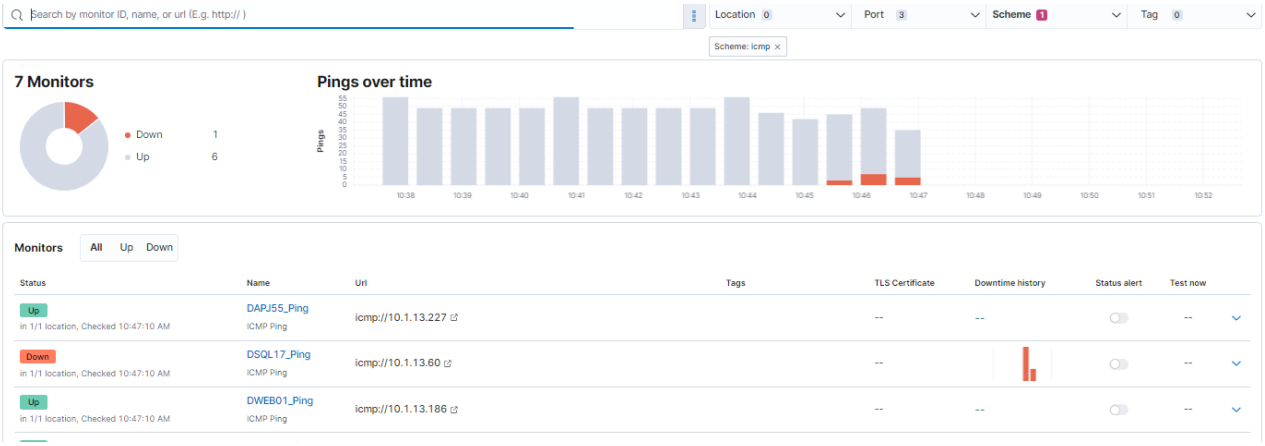


***Figura 4.5*** *Monitoreo del servicios ICMP en los equipos.*
*(Fuente: Captura de la consola Elastic implementada)*

### 4.2.2 Monitoreo de Puertos TCP/IP

El monitoreo de puertos, se realiza mediante el protocolo de control de transmisión o TCP, el cual informa la disponibilidad de conexiones en la red. En la Figura 4.6 se presenta el monitoreo de puertos, esto facilita al administrador de tecnología a identificar posibles bloqueos o indisponibilidad del servicio en el equipo.

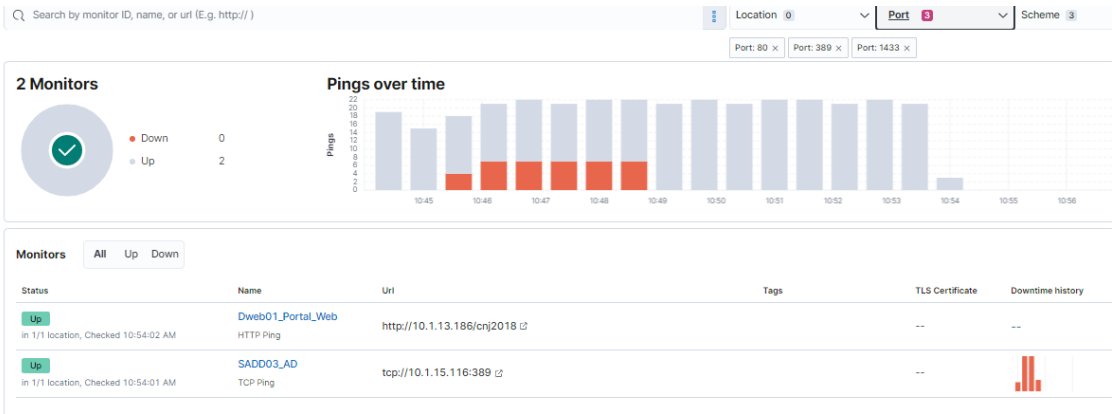


***Figura 4.6*** *Monitoreo de puertos.*
*(Fuente: Captura de la consola Elastic implementada)*

### 4.2.3 Dirección URL.

El monitoreo de una dirección específica o URL[14] (Localizador uniforme de recursos), es un mecanismo que permite identificar si los servicios a nivel de aplicación (basado en el modelo OSI[15], Capa 7) se encuentran disponibles. En la Figura 4.7 se presenta el monitoreo de la dirección URL.

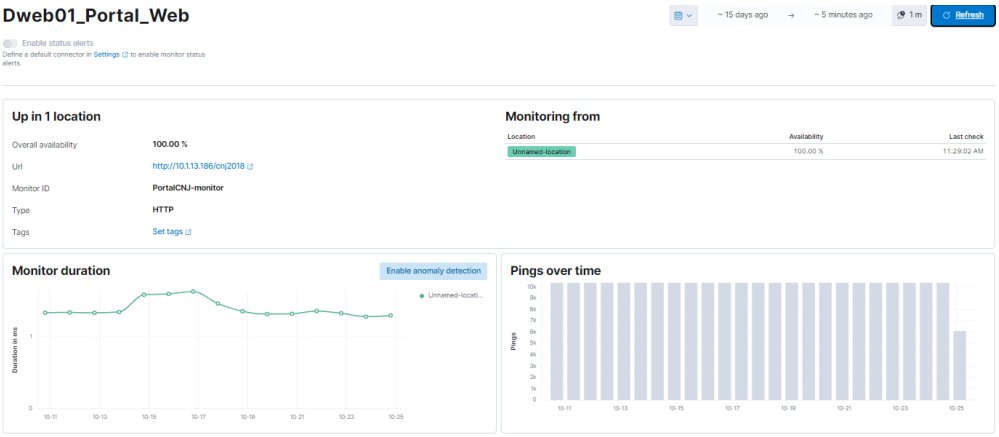


***Figura 4.7*** *Monitoreo de una dirección URL.*
*(Fuente: Captura de la consola Elastic implementada)*

## 4.3 Monitoreo de logs en equipos Linux

Los eventos y logs en equipos Linux, se almacenan de manera predeterminada en el directorio */var/log*, aquí se guarda la información de eventos propios del sistema operativo y aplicativos. Su limitación se basa en la visualización vía texto. Esto impide tener un análisis por la cantidad de registros que existen en dicho repositorio.

[14] URL Es una dirección única en la red para que los usuarios la localicen. (EDIX, 2022)
[15] OSI – Capa 7 El Modelo OSI divide las funciones del sistema de red, la capa 7 hace referencia a las funcionalidades y servicios de la aplicación (CloudFlare, s.f.)

En la Figura 4.8 se observa los eventos de logs del archivo: *"/var/log/messages"*. Al ser una visualización de manera textual, hace que el administrador tarde en determinar el problema y plantear una solución.

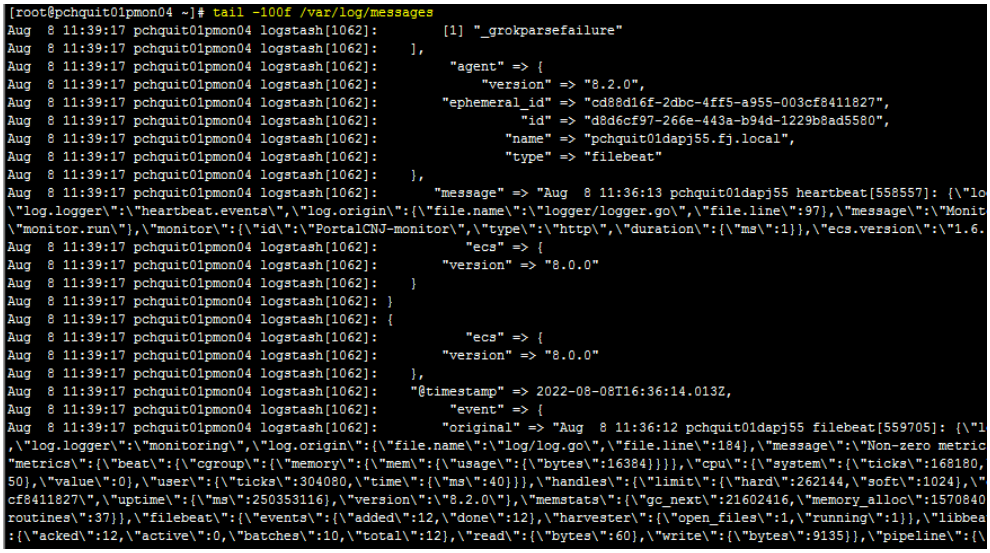


***Figura 4.8*** *Visualizar de logs en un equipo Linux.*
*(Fuente: Logs de un equipo cliente)*

### 4.3.1 Visualización de logs en equipos Linux

Para el monitoreo de logs en equipos con sistema operativo Linux, se instaló el agente Filebeat según lo descrito en la sección 3.6.3. Este agente permite centralizar los logs que se generan en los distintos equipos en la herramienta Elasticsearch, la que permite optimizar el tiempo de búsquedas de un determinado evento. Así el administrador puede analizar y visibilizar los eventos con mayor rapidez.

En la Figura 4.9 se observa el panel de eventos con el agente Filebeat en la consola de la herramienta Kibana, mismo que permite ejecutar búsquedas con palabras claves, enteras o cortas, para facilitar al administrador en el análisis y búsqueda de determinado evento.

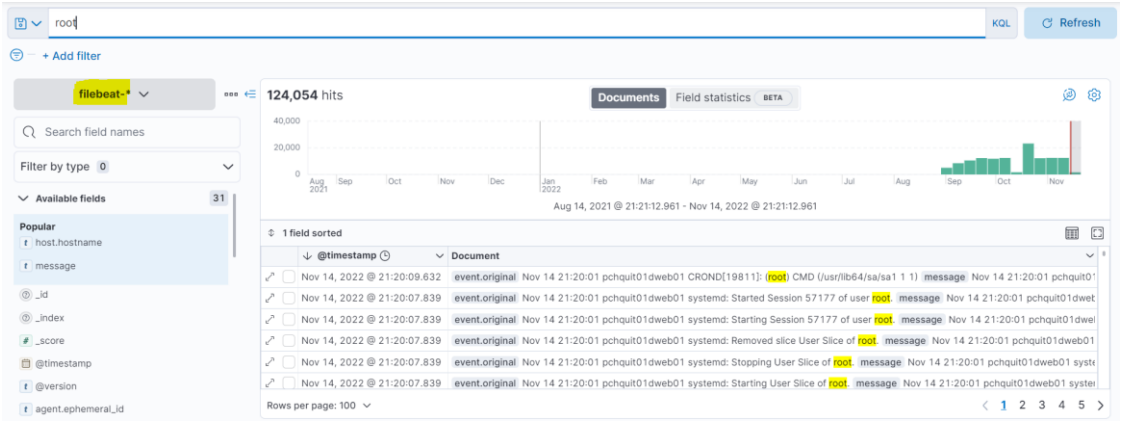


***Figura 4.9*** *Panel de logs del agente Filebeat en la herramienta Kibana.*
*(Fuente: Captura de la consola Elastic implementada)*

En la Figura 4.10 se presenta un Dashboard de monitoreo de logs para equipos con sistema operativo Linux, referente a: número de logs, logueos fallidos, creación y eliminación de usuarios, lo que permite al administrador a tomar correctivos sobre la seguridad en los equipos. Para esto se crearon las siguientes visualizaciones:

- Número de eventos, mediante un contador.
- Porcentaje de usuarios creados en los diferentes equipos, a través de un pastel.
- Cantidad de usuarios creados, en forma numérica.
- Cantidad de usuarios eliminados, mediante representación numérica.
- Logueos fallidos por dispositivo, mediante un diagrama de barras.
- Logueos fallidos, con una descripción en forma de tabla.

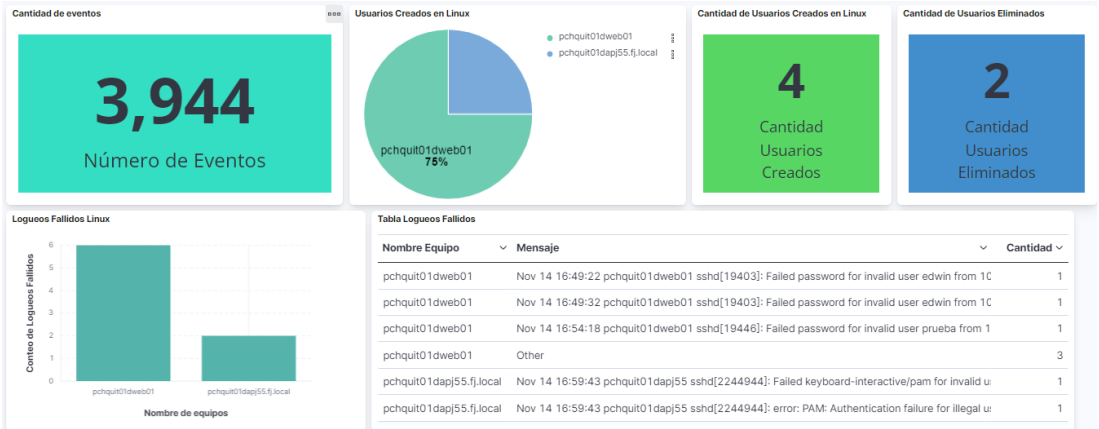


***Figura 4.10*** *Dashboard de Logs a nivel de Linux.*
*(Fuente: Captura de la consola Elastic implementada)*

### 4.4 Monitoreo de logs en equipos Windows

En los equipos con sistema operativo Windows, los logs se almacenan de forma predeterminada en el directorio *"C:\Windows\System32\winevt\Logs"*, aquí se guarda la información propia del sistema operativo y aplicativos.

En la Figura 4.11 se presenta la consola *"visor de eventos"*, propia de equipos con sistema operativo Windows. La limitación de esta herramienta es el tiempo de búsqueda en encontrar un evento o realizar un filtro por un determinado valor.

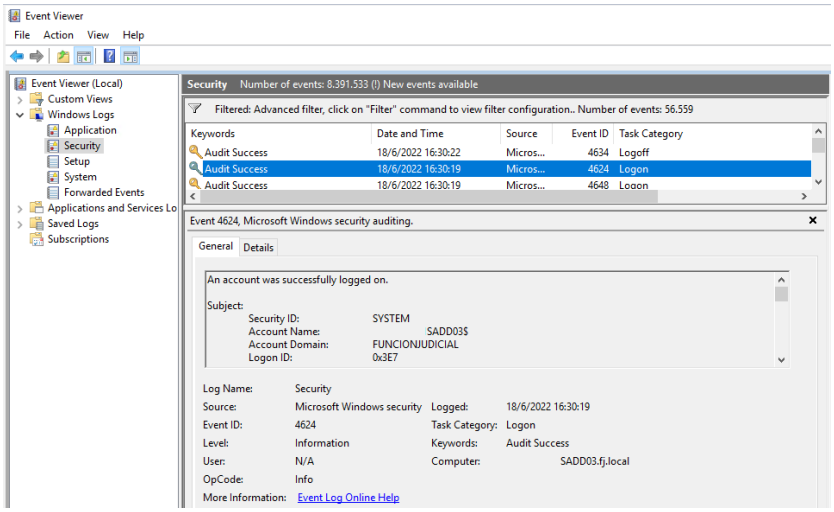


***Figura 4.11*** *Visor de eventos de Windows.*
*(Fuente: Consola de Visor de eventos de un equipo cliente)*

#### 4.4.1 Visualización de logs en equipos Windows.

Para el monitoreo de logs en equipos con sistema operativo Windows, se instaló el agente Winlogbeat según lo descrito en la sección 3.5.3. Este agente permite centralizar los logs que se generan en los distintos equipos en la herramienta Elasticsearch, el cual permite optimizar el tiempo de búsquedas de un determinado evento, agrupar los tipos de eventos o identificadores. Así el administrador puede analizar y visibilizar los eventos con mayor rapidez.

En la Figura 4.12 se presenta el panel de eventos filtrados con el agente Winlogbeat en la consola de la herramienta Kibana, mismo que permite realizar búsquedas con palabras claves, enteras o cortas, de esta forma se obtiene resultados que facilitan al administrador en la optimización del análisis y búsqueda de un evento.

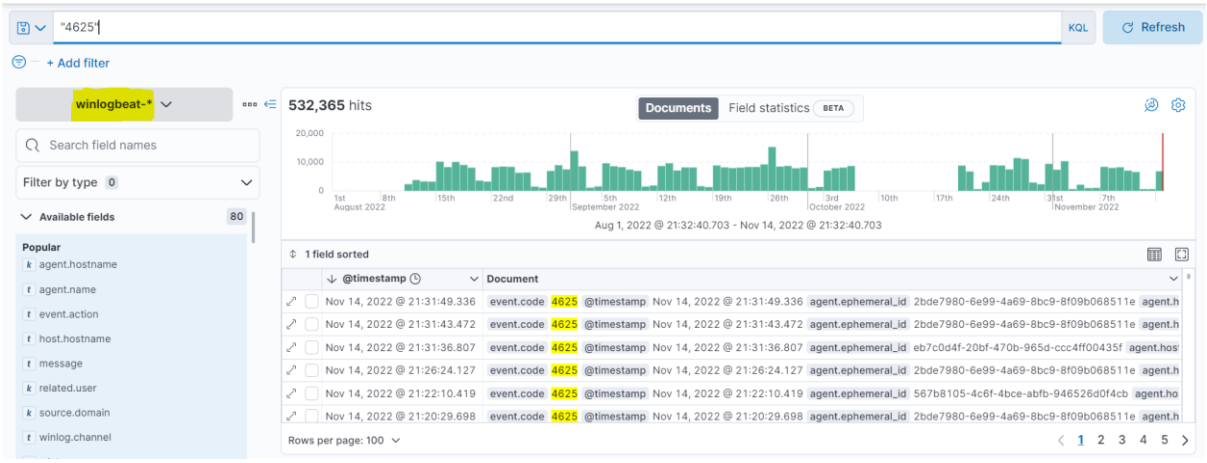


***Figura 4.12*** *Panel de logs del agente Winlogbeat en la herramienta Kibana.*
*(Fuente: Captura de la consola Elastic implementada)*

En la Figura 4.13 se muestra un Dashboard de monitoreo de logs para equipos con sistema operativo Windows, referente a: cantidad de eventos generados, tipos de eventos por ID o clasificación. Esto facilita al administrador a tomar correctivos sobre la seguridad en los equipos, se usaron las siguientes visualizaciones:

- Número de eventos, mediante un contador
- Número de eventos a lo largo del tiempo, mediante un gráfico de barras
- Top de eventos por identificador, a través de una lista
- Niveles de eventos, a través de una tabla

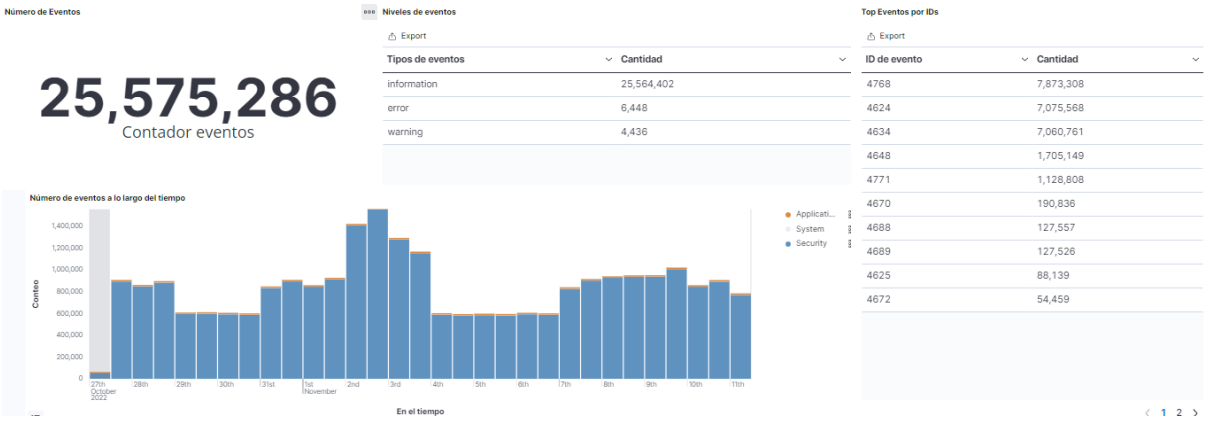


***Figura 4.13*** *Dashboard de logs de seguridad para equipos Windows.*
*(Fuente: Captura de la consola Elastic implementada)*

## 4.5 Presentaciones en la herramienta Kibana.

La herramienta de monitoreo Elastic, tiene el aplicativo Kibana, misma que permite visualizar de manera gráfica los datos que se recolectan con los diferentes agentes como: Heartbeat, Metricbeat, Filebeat y Winlogbeat, de una manera visual. Para esto se crean visualizaciones gráficas como: círculos, anillos, barras verticales u horizontales, lineales, tablas entre otras (Cabrera, 2022). Las que permite mostrar el consumo de recursos, disponibilidad de servicios, como la visualización de eventos, en un determinado tiempo. De esta manera el administrador de tecnología puede interpretar la información y así puede tomar decisiones para corregir posibles incidentes u problemas que se estén en la infraestructura.

### 4.5.1 Visualización de logueos fallidos y cuentas de usuario bloqueadas en el Directorio Activo.

Como parte de los objetivos de este trabajo se planteó visualizar mediante Dashboard los logs de seguridad de Windows, referente al bloqueo de cuentas de usuarios de un Directorio Activo. En la Figura 4.11 se presenta el Dashboard donde las cuentas bloqueadas y logueos fallidos. Cabe indicar para que una cuenta sea bloquee, debe existir de manera previa un número de logueos fallidos consecutivos, esta regla se configura con anticipación en el servidor de

Directorio Activo. Está información brinda al administrador de tecnología las herramientas para identificar posibles ataques o denegación de acceso a un determinado usuario, para esto se usaron las siguientes visualizaciones:

- Número de cuentas de usuarios bloqueadas, a través de etiqueta.
- Cantidad de cuentas con logueos fallidos, mediante una etiqueta.
- Presentación de cuentas de usuario bloqueadas, a través de una tabla.
- Información de los logueos satisfactorios vs fallidos, en forma de pastel.

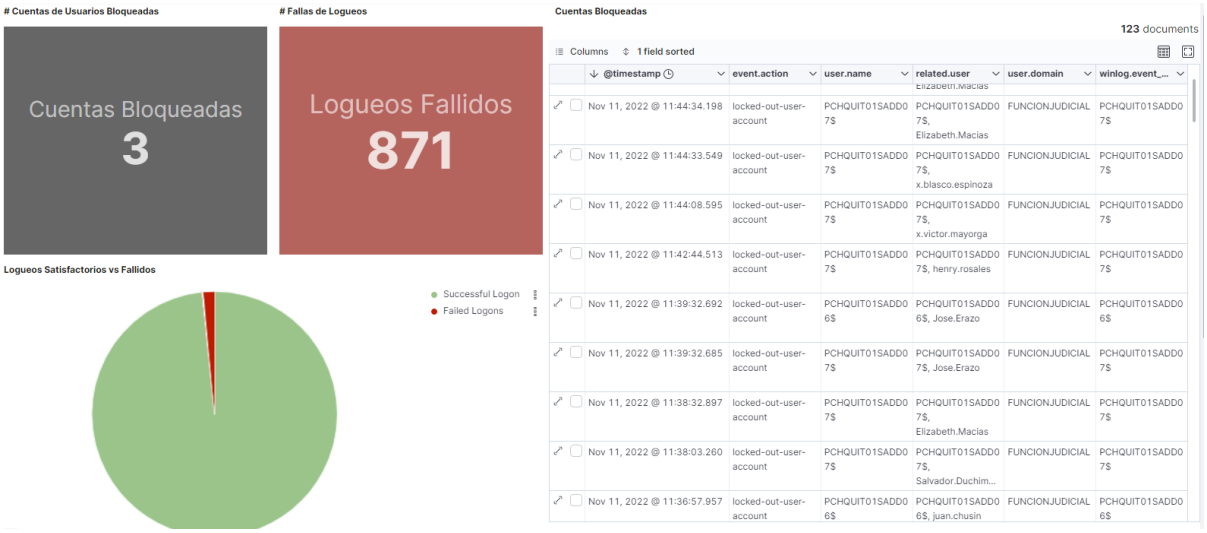


***Figura 4.14*** *Dashboard logueos fallidos y cuentas de usuario bloqueadas en un Directorio (Fuente: Captura de la consola Elastic implementada)*

### 4.5.2 Visualización de la administración de eventos de cuentas de usuario del Directorio Activo.

Al ser uno de los objetivos del presente trabajo, la visualización mediante Dashboard de logs de seguridad de Windows, con respecto a modificaciones de atributos realizadas a nivel de las cuentas de usuarios de un Directorio Activo. En la Figura 4.12 se presenta el Tablero de Visualización de administración de cuentas de usuario en el Directorio Activo, referente a eventos de administración como: creación, modificación, bloqueos, y eliminación de cuentas de usuario. Está información facilita al administrador del Directorio Activo a identificar

modificaciones de información en las propiedades del objeto usuario, así también como desactivación de cuentas y posible suplantación de identidad. En el Dashboard se usaron las siguientes visualizaciones:

- Administración de usuarios por eventos, en forma de pastel.
- Listado de los eventos de usuario con su identificador, mediante una tabla
- Distribución de eventos en el tiempo, mediante un diagrama de columnas
- Conteo de usuarios creados, en forma numérica.
- Conteo de usuarios deshabilitados, de representación numérica.
- Conteo de usuarios bloqueados, mediante representación numérica
- Conteo de usuarios eliminados, de manera numérica.
- Lista de usuarios creados, a través de una tabla.
- Lista de usuarios deshabilitados, mediante tabla.
- Lista de usuarios bloqueados, a través de una tabla
- Lista de usuarios eliminados, mediante tabla.

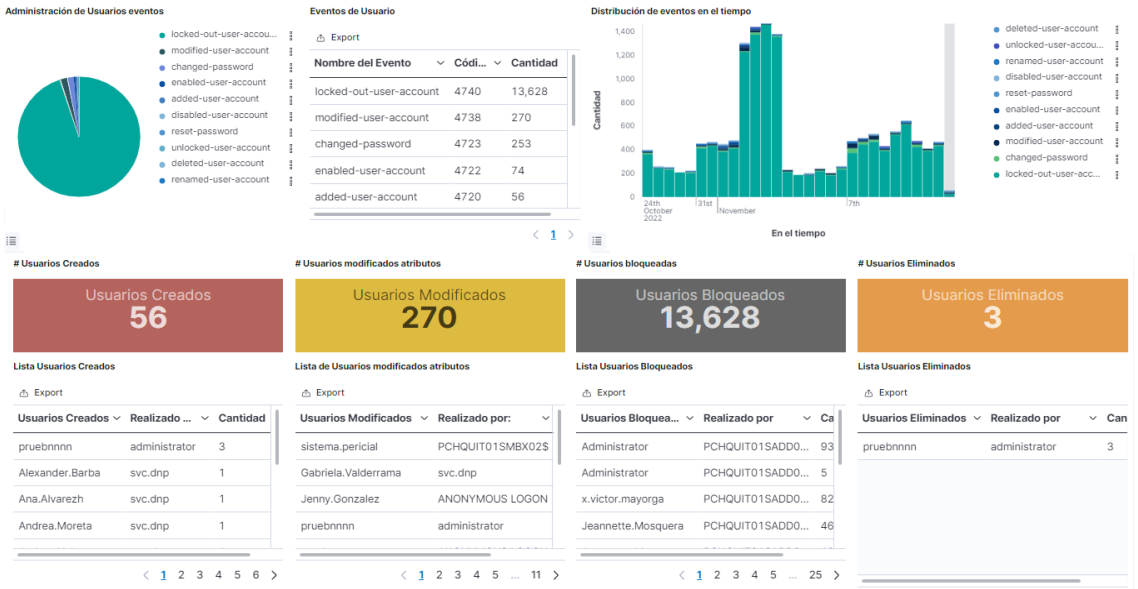


***Figura 4.15*** *Dashboard administración de eventos en cuentas de usuario del Directorio Activo.*
*(Fuente: Captura de la consola Elastic implementada)*

# CAPÍTULO V: CONCLUSIONES Y RECOMENDACIONES

## 5.1 Conclusiones

- La herramienta de monitoreo usada en la presente investigación funciona como un motor de búsqueda con capacidades analíticas, la misma que procesan grandes volúmenes de datos en tiempo real, tal es el caso de la información de los archivos de *logs* de servidores y aplicaciones. Además, puede ser utilizada para el monitoreo de recursos tecnológicos como: procesamiento, memoria RAM, red, capacidad de almacenamiento, disponibilidad de servicios mediante los puertos de los aplicativos; al ser una tecnología de código abierto permite adaptarse con facilidad a las necesidades del administrador de las plataformas tecnológicas en los diversos proyectos de tecnologías de la información.

- La herramienta Elastic permite la generación de tableros de control o Dashboards para presentar información e indicadores de manera gráfica e intuitiva de tal forma que se pueda entregar al personal operativo y gerencial información en tiempo real de la infraestructura tecnológica para la toma de decisiones sobre eventos que se puedan presentar en la misma.

- A través de la interfaz de usuario y la capa de visualización altamente personalizable que la herramienta Elastic brinda mediante su módulo Kibana, se puede crear de manera gráfica la información de los eventos y errores de los equipos monitoreados. De esta manera el operador de la infraestructura tecnológica puede detectar incidentes y problemas que provoquen indisponibilidad de los servicios tecnológicos.

- Durante el desarrollo del componente práctico de este trabajo se pudo verificar que la herramienta Elastic facilita la presentación de resultados en diferentes formas como son: gráfica, numérica y estadística. Esto permite tener un mayor control de los diferentes servicios tecnológicos que ingresan a la herramienta de monitoreo, como por ejemplo los eventos de seguridad en el servicio de Directorio Activo, que proporcionan información a los administradores para facilitar la implementación de medidas correctivas y proactivas ante posibles eventos de seguridad que se pueden detectar tales como la cantidad de intentos de logueos fallidos de un usuario determinado.

- Una de las características que más se analizó de la herramienta Elastic, es su capacidad de almacenar de manera independiente los eventos o logs que se generan en cada equipo monitoreado. Esta función permite generar visualizaciones independientes de los eventos generados, como por ejemplo la visualización gráfica de la cantidad de eventos en cuentas de usuario referente a: creación, modificación y eliminación dentro del servicio de Directorio Activo, mismas que permiten tener auditorias de los diferentes eventos.

- En el escenario de los eventos de logs generados a nivel de seguridad en Windows, se analizó en un equipo que cuenta con el rol de Directorio Activo, en el que es posible visualizar la información de logs mediante el visor de eventos propio de Windows, cada evento se genera cronológicamente y con un identificador del evento; buscar un incidente en particular de la manera tradicional acarrea demasiado tiempo a un administrador, además que no permite mostrarlos e interpretarlos de una manera gráfica. Haciendo uso de la herramienta Elastic se facilita al administrador la tarea de

procesar y buscar incidentes en los datos de logs de eventos de manera rápida y con una visualización gráfica fácil de interpretar.

- Referente a los logs de eventos de seguridad que se generan en un servidor Linux, estos se almacenan en archivos planos, de manera cronológica según las actividades que se generan en el sistema operativo y aplicativos, esto hace que la búsqueda y visualización sea poco amigable, es decir línea por línea y no de una manera gráfica. Mediante el uso de la herramienta Elastic se facilita al administrador tener búsquedas personalizadas y presentar resultados de manera visual, con lo que el administrador de la plataforma optimiza el tiempo de análisis de incidentes en los logs de eventos.

- Para conocer la disponibilidad o pérdida de un determinado servicio en servidores y aplicativos se realiza un monitoreo mediante el uso del protocolo ICMP, puertos propios de la aplicación y verificación de direcciones URL. Mediante la herramienta Elastic se pudo verificar que se pueden realizar estos monitoreos, y se brindó al administrador de la plataforma la opción de identificar posibles indisponibilidades de servicios.

- La integración de los equipos monitoreados con sistemas operativos Windows y Linux, es muy ágil y sencilla, ya que los agentes instalados se conectan de una manera nativa con el servidor principal de la herramienta Elastic my mediante la obtención de los datos a través de los agentes, es posible mostrar mediante Dashboards la información recolectada en tiempo real. Estas funcionalidades permiten tomar decisiones sobre eventos que se puedan presentar en la infraestructura tecnológica.

## 5.2 Recomendaciones y Trabajos futuros

- Para ambientes productivos se recomienda obtener un respaldo periódico de los archivos de configuraciones de todos los servicios configurados de la herramienta, esto es útil en caso de realizar auditorías y análisis más exhaustivo sobre la información recolectada de los clientes, con lo cual se contará con un contingente en caso de existir algún incidente en la plataforma.

- Es necesario como trabajo previo analizar la estructura de los eventos de logs que se van a recolectar, a fin de localizar de manera óptima la información clave que se necesita monitorear, de esa manera al momento de realizar las búsquedas y generar visualizaciones gráficas se puede acoplar la información a las necesidades del usuario final.

- Para futuras implementaciones, se recomienda trabajar en una arquitectura que brinde alta disponibilidad para la recolección y análisis de los datos, es decir en un clúster de al menos dos nodos y de esa manera garantizar la disponibilidad del servicio de la Herramienta Elastic, para presentar las visualizaciones de los eventos que se generan en los equipos monitoreados.

- Con respecto a la continuidad del presente trabajo, se recomienda extender el análisis a los distintos eventos que genera en un servidor de Directorio Activo, como por ejemplo la administración de cuentas de equipo y auditorias de seguridad. Con el propósito de actuar de manera proactiva ante posibles ataques cibernéticos que en el tiempo presente se encuentran con frecuencia.

## REFERENCIAS


Banco de la República. (2018). *¿Qué es indexación y cuáles son los mecanismos de indexación que existen?* Obtenido de https://www.banrep.gov.co/es/indexacion-y-cuales-son-mecanismos-indexacion-existen

Bautista García, I. J. (30 de Marzo de 2021). *Backend y Frontend, ¿Qué es y cómo funcionan en la programación?* Obtenido de https://www.servnet.mx/blog/backend-y-frontend-partes-fundamentales-de-la-programacion-de-una-aplicacion-web

Cabrera, G. (05 de Febrero de 2022). *¿Que son las visualizaciones en Kibana?* Obtenido de https://somospnt.com/blog/242-que-son-las-visualizaciones-en-kibana#:~:text=Para%20crear%20una%20visualizaci%C3%B3n%20en,datos%20cargados%20en%20el%20Elasticsearch.

CloudFlare. (s.f.). *¿Qué es la capa 7 de Internet?* Obtenido de https://www.cloudflare.com/es-es/learning/ddos/what-is-layer-7/#:~:text=La%20capa%207%20hace%20referencia,las%20que%20interact%C3%BAan%20los%20usuarios.

Davinci Group. (2020 de Julio de 16). *¿Qué son los Beats de elastic? + Ejemplo de uso de Filebeat*. Obtenido de https://www.davincigroup.es/beats-elastic-ejemplo-filebeat/

Digital Guide IONOS. (2016). *Ficheros log: Toda la información de registro en un archivo*. Obtenido de https://www.ionos.es/digitalguide/online-marketing/analisis-web/el-log-el-archivo-de-registro-de-procesos-informaticos/

EDIX. (11 de Julio de 2022). *URL*. Obtenido de https://www.edix.com/es/instituto/que-es-url/

Elastic. (s.f.). *¿Qué es Elasticsearch?* Obtenido de https://www.elastic.co/es/what-is/elasticsearch

Elastic. (s.f.). *¿Qué es Kibana?* Obtenido de https://www.elastic.co/es/what-is/kibana

Elastic. (2021). *2021 Gartner Magic Quadrant for SIEM*. Obtenido de https://www.elastic.co/es/campaigns/2021-gartner-magic-quadrant-siem

Elastic. (2022). *Agentes de datos ligeros*. Obtenido de https://www.elastic.co/es/beats/

Elastic. (2022). *Centraliza, transforma y almacena tus datos*. Obtenido de Elastic: https://www.elastic.co/es/logstash/

Elastic. (s.f.). *Filebeat overview*. Obtenido de https://www.elastic.co/guide/en/beats/filebeat/current/filebeat-overview.html

Elastic. (s.f.). *Heartbeat overview*. Obtenido de https://www.elastic.co/guide/en/beats/heartbeat/current/heartbeat-overview.html

Elastic. (s.f.). *Metricbeat overview*. Obtenido de https://www.elastic.co/guide/en/beats/metricbeat/current/metricbeat-overview.html

Elastic. (s.f.). *Winlogbeat Overview*. Obtenido de Winlogbeat Overview: https://www.elastic.co/guide/en/beats/winlogbeat/current/_winlogbeat_overview.html

Gartner. (02 de Marzo de 2022). *Elastic (ELK) Stack Reviews*. Obtenido de https://www.gartner.com/reviews/market/security-information-event-management/vendor/elasticsearch/product/elastic-elk-stack

GNU. (s.f.). *¿Qué es el Software Libre?* Obtenido de https://www.gnu.org/philosophy/free-sw.es.html

Hershkovitch, D. (19 de Noviembre de 2018). *Monitoreo de tiempo de actividad con Heartbeat y el Elastic Stack*. Obtenido de https://www.elastic.co/es/blog/uptime-monitoring-with-heartbeat-and-the-elastic-stack

IBM. (21 de Octubre de 2022). *Introducción: Clústeres*. Obtenido de https://www.ibm.com/docs/es/was-zos/9.0.5?topic=servers-introduction-clusters

Lerena, S. (2020 de Diciembre de 3). *Logs: qué son y por qué monitorizarlos*. Obtenido de Qué es un log, para qué sirve y algunos de los principales tipos de log

Llamas, J. (s.f.). *Software libre*. Obtenido de economipedia: https://economipedia.com/definiciones/software-libre.html

Llamas, J. (s.f.). *Software propietario*. Obtenido de Economipedia: https://economipedia.com/definiciones/software-propietario.html

ManageEngine. (s.f.). *¿Qué es el monitoreo de servidores?* Obtenido de https://www.manageengine.com/latam/network-monitoring/software-monitoreo-de-servidores.html#:~:text=El%20monitoreo%20de%20servidores%20consiste,disco%2C%20el%20procesador%2C%20etc.

MDN Plus. (04 de Agosto de 2022). *Trabajando con JSON*. Obtenido de https://developer.mozilla.org/es/docs/Learn/JavaScript/Objects/JSON

Opensource.com. (s.f.). *What is open source?* Obtenido de Open Source: https://opensource.com/resources/what-open-source

Ortiz, D. (20 de Octubre de 2021). *¿Qué es un dashboard y para qué se usa?* Obtenido de https://www.cyberclick.es/numerical-blog/que-es-un-dashboard

Palma, W. (2005). *Almacenamiento y Recuperación de la Información*. Obtenido de https://www.inf.utfsm.cl/~wpalma/ari/indices.pdf

Pari, J. (26 de Abril de 2020). *ELK – ElasticSearch, Logstash y Kibana*. Obtenido de Arquitectura Cloud: https://arquitecturacloud.com/elk-elasticsearch-logstash-y-kibana/

Pure Storage. (s.f.). *¿Qué es VMware?* Obtenido de https://www.purestorage.com/la/knowledge/what-is-vmware.html

Red Hat. (31 de Octubre de 2017). *¿Qué es una API?* Obtenido de https://www.redhat.com/es/topics/api/what-are-application-programming-interfaces

Red Hat. (Octubre de 2019). *¿Qué es el open source?* Obtenido de https://www.redhat.com/es/topics/open-source/what-is-open-source

Riley, M. (2021). *Elastic reconocido como Retador en el Cuadrante Mágico de Gartner 2021 de motores de información*. Obtenido de Elastic: https://www.elastic.co/es/blog/elastic-recognized-as-a-challenger-in-the-2021-gartner-magic-quadrant-for-insight-engines

Rodriguez Flores, J. (07 de Marzo de 2019). *¿Que es ELK? Elasticsearch, Logstash y Kibana*. Obtenido de https://usuarioperu.com/2019/03/07/que-es-elk-elasticsearch-logstash-y-kibana/#:~:text=Principales%20Caracter%C3%ADsticas,-Recolecta%20logs%20de&text=ELK%20tiene%20m%C3%B3dulos%20de%20seguridad,a%20cada%20tipo%20de%20usuario.&text=Presenta%20esta%20inform

Sanchez Avalo, S. (17 de Septiembre de 2020). *¿Qué es Elastic Stack? Aprendé a monitorear tus aplicaciones*. Obtenido de https://somospnt.com/blog/175-que-es-elastic-stack

Sánchez López, J. (Noviembre de 2013). SOFTWARE 1. Sistema Operativo Software de Aplicación.

Santos González, P. (19 de Febrero de 2019). *Características de la arquitectura de Elasticsearch*. Obtenido de https://openwebinars.net/blog/caracteristicas-de-la-arquitectura-de-elasticsearch/

SG. (2012). *OpenJDK: open source Java hecho realidad*. Obtenido de https://sg.com.mx/content/view/173

Soto, J. A. (25 de Julio de 2020). *¿Qué es PowerShell y para qué sirve?* Obtenido de https://www.geeknetic.es/PowerShell/que-es-y-para-que-sirve

Stephen Emmott, A. M. (17 de 03 de 2021). Magic Quadrant for Insight Engines.