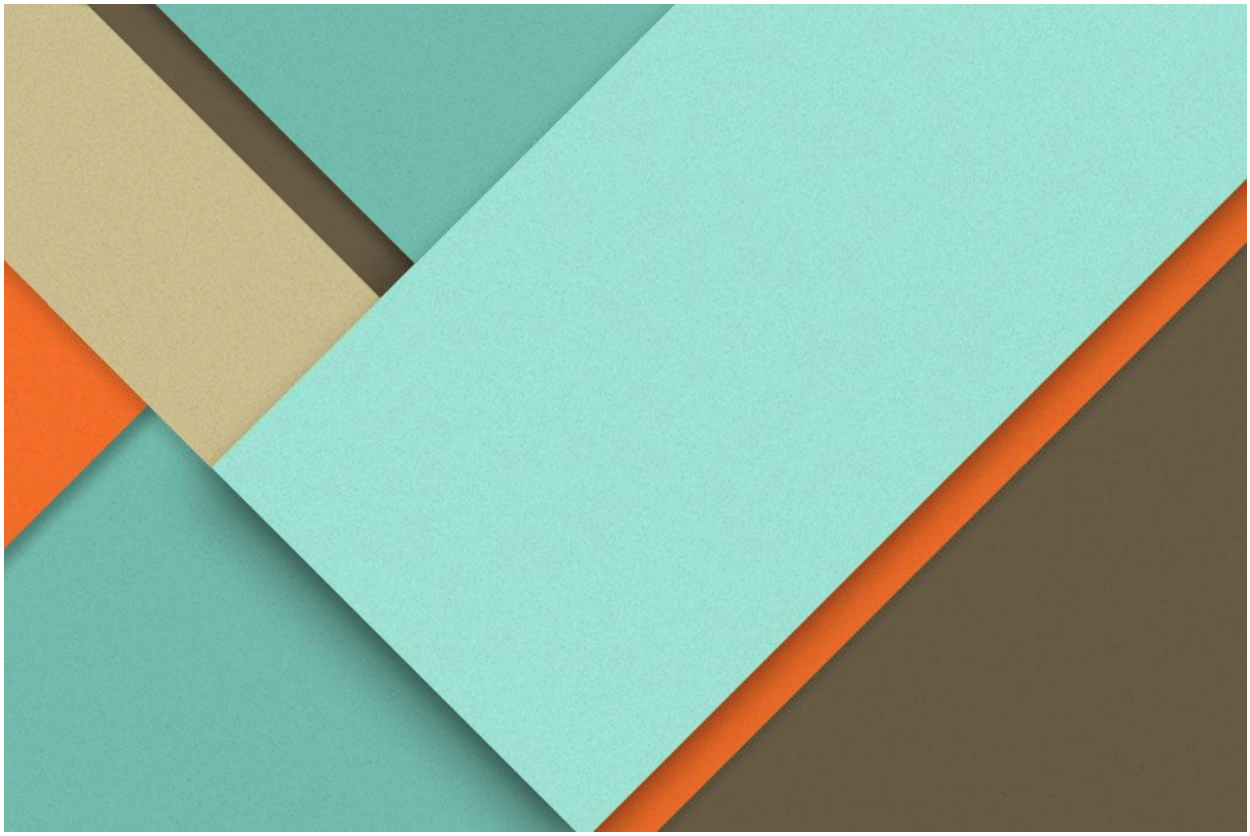


Analisis de log

11/11/2018

ALAOUI MHAMMEDI Mohammed

Repositorio : https://github.com/ilFreddo48/gestion_log

Visión general

El análisis de logs se ha convertido en una actividad esencial a desarrollar durante una auditoría SEO. Los logs de los servidores Web nos permiten comprender cómo los rastreadores de los buscadores interactúan con nuestro sitio, y un análisis de los mismos pueden permitirnos identificar información crítica para controlar la seguridad y el rendimiento de la página web. El objetivo fundamental de este análisis es detectar comportamientos fuera del común, por ejemplo una votación se realiza en España normalmente tiene que ser reflejada que todos los clientes conectados tienen que tener una ip de España para garantizar un voto transparente .

Para este objetivo vamos a implementar este sistema mediante ELK stack lo que es una combinación de tres herramientas Elasticsearch, Logstash y KIbana vamos a hablar en detalle sobre estas herramientas en el apartado del estado del arte.

Contexto

El objetivo general del trabajo es la puesta en práctica de los contenidos teórico

prácticos de la asignatura. En este proyecto, se implementará un analizador de los fichero log del sistema, donde analiza los ficheros para obtener metadata sobre los usuario que se logean en el momento de una votación .

Durante todo el proceso, se gestionará el código mediante un repositorio

localizado en GitHub y al cual, se suscribirá un servicio de integración continua, Travis CI, que ejecutará las pruebas alojadas en el repositorio con cada cambio de código.

Planificación del proyecto

Toda la implementación del proyecto ha sido desarrollada por Mohammed ALAOUI MHAMMEDI:

- Documentación
- Investigación y Desarrollo

La planificación

se basa en un único Milestone (M6), el cual incluye,

- Configuración del servidor de integración continua.
- Implementación del código.
- Documentación del proyecto

Estado del arte:

Elasticsearch es un servidor de búsqueda basado en Lucene. Provee un motor de búsqueda de texto completo, distribuido y con capacidad de multi-tenencia con una interfaz web RESTful y con documentos JSON. Elasticsearch está desarrollado en Java y está publicado como código abierto bajo las condiciones de la licencia Apache.

Ventajas

Se podrían enumerar varias ventajas que brinda esta herramienta. Algunas de las más destacables son las siguientes:

- Al estar desarrollado en Java, es compatible en todas las plataformas donde Java lo sea.
- Tiene una gran velocidad de respuesta.
- Es distribuido, lo que lo hace fácilmente escalable y adaptable a las distintas situaciones.
- Simple realiza respaldos de los datos almacenados.
- Utiliza objetos JSON como respuesta, por lo que es fácil de invocar desde varios lenguajes de programación.

Desventajas

Como todo, ElasticSearch posee algunas desventajas:

- Sólo soporta como tipos de respuesta JSON, lo que lo limita al no soportar otros lenguajes, como CSV ó XML.
- Algunas situaciones pueden generar casos de "split brain".

Elasticsearch puede ser usado para buscar todo tipo de documentos. La búsqueda es escalable y casi en tiempo real, soportando multi-tenencia.[57] "Es distribuido, haciendo que los índices se puedan dividir en fragmentos y cada uno teniendo cero o más réplicas. Cada nodo alberga uno o más fragmentos, actuando como un coordinador para delegar operaciones a los fragmentos correctos. El rebalanceo y ruteo se realizan automáticamente.

Logstash es una herramienta para la administración de logs. Esta herramienta se puede utilizar para recolectar, parsear y guardar los logs para futuras búsquedas.[2] La aplicación se encuentra basada en jRuby y requiere de Java Virtual Machine para correr. Como corre en JVM puede ser ejecutada en cualquier Sistema Operativo que corra JVM (Linux, Mac OS X, Windows).[3]

Logstash soporta un número de entradas, códecs, filtros y salidas. Las entradas son las fuentes de datos. Los códecs esencialmente convierten un formato de

entrada en un formato aceptado por Logstash, así como también transforman del formato de Logstash al formato deseado de salida. Estos son utilizados comúnmente si la fuente de datos no es una línea de texto plano. Los filtros son acciones que se utilizan para procesar en los eventos y permiten modificarlos o eliminar eventos luego de ser procesados. Finalmente, las salidas son los destinos donde los datos procesados deben ser derivados.

En Logstash y con una infraestructura distribuida, cada servidor web debe ser configurado para correr Lumberjack (es opcional pero altamente recomendado para economizar recursos). Lumberjack hace un forward de los logs a un servidor corriendo Logstash con una entrada de Lumberjack. Como Lumberjack require SSL, los logs van a ser encriptados del servidor web al servidor de logs central.

Un servidor central de logs tiene la debilidad de ser único ante una falla. Es por eso que debe pensarse en una opción que contemple la disponibilidad del sistema.

Logstash puede ser configurado para utilizar múltiples servidores pero solo enviará los logs a uno de ellos hasta que ese servidor falle. Si sucede esto, todos los logs previamente recolectados no serán accesibles hasta que ese host sea nuevamente habilitado. Básicamente, Logstash soporta un servidor corriendo como máster y servidores en espera (shippers).

Kibana is an open source data visualization plugin for Elasticsearch. It provides visualization capabilities on top of the content indexed on an Elasticsearch cluster. Users can create bar, line and scatter plots, or pie charts and maps on top of large volumes of data.

The combination of Elasticsearch, Logstash, and Kibana, referred to as the "Elastic Stack" (formerly the "ELK stack"), is available as a product or service. Logstash provides an input stream to Elastic for storage and search, and Kibana accesses the data for visualizations such as dashboards.

Entorno de desarrollo:

Como este proyecto no necesita código, lo que tenemos que modificar los ficheros de configuración de las herramientas. Estos servicios se despliegan en contenedores docker para facilitar la tarea del despliegue, de otro lado para gestionar los contenedores y gestionarlos hemos usado Docker-compose mediante un fichero llamado docker-compose.yml

1. Docker
2. Docker-compose

Gestionamos los cambios a traves de Git y Github que son dos herramientas que se comunican a través de la cuenta de Github. Git permite hacer subida de código, gestionar ramas, etc. a través de su consola, quedando reflejado cada cambio en Github.

Gestion del código fuente :

Un commit se indica el cambio que se ha realizado de la siguiente forma:

- **Título**: Empieza con un verbo en participio, la función que se ha modificado/creado/borrado y después el archivo que se va a modificar.

- **Comentario**: Explicación detallada de lo que se ha realizado.

Integración continua :

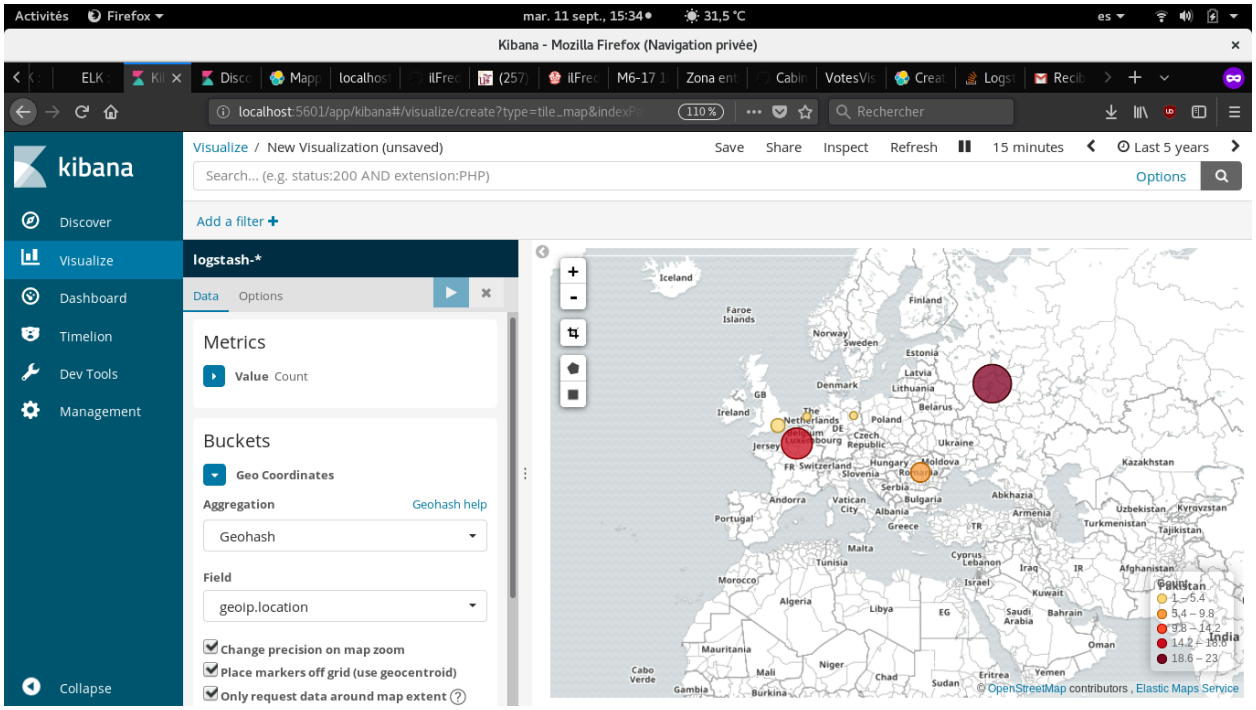
El servicio de integración continua a usar será Travis CI. Mediante webhooks, GitHub notifica a Travis de que ha habido un cambio de código. Este, clona el repositorio y ejecuta el archivo de configuración .travis.yml localizado en él.

El proceso de integración continua que hemos usado es el propuesto por la asignatura Travis CI. Su funcionamiento se basa en clonar el repositorio de GitHub a un nuevo entorno virtual y llevar a cabo una serie de tareas para construir y probar tu código. Si alguna de

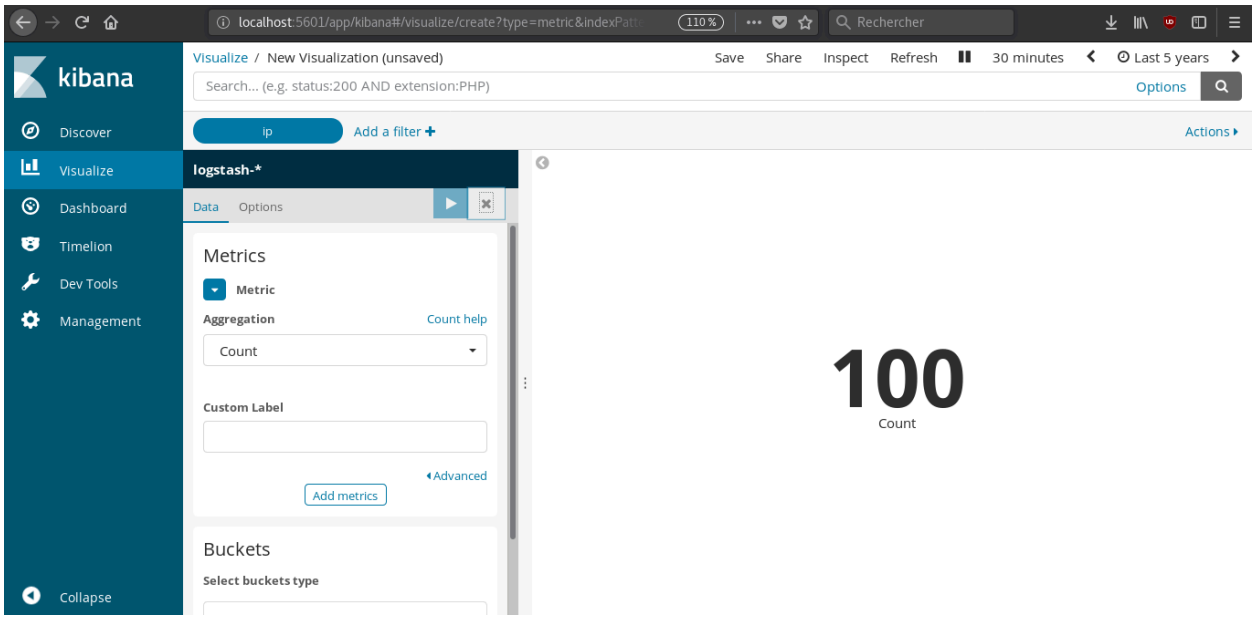
estas tareas falla, el build se considera broken. Si no falla ninguna tarea, el build se considera passed y Travis CI puede desplegar tu código al servidor de producción.

A través del archivo .travis.yml, interactua con Github haciendo que se cree una release nueva cada vez que se realiza algún commit a la rama master.

Resultados :



como se puede ver en el visualizador Kibana podemos los usuarios de donde se han conectado a través del ip que está en el fichero de log, como el log estaba un log de prueba que ha proporcionado elastic salen sitios fuera de España. Asi podemos tener una vista general de la localización de los usuarios y la densidad según se encuentran,. Podemos ver también el número total de usuarios .



y muchas más se puede hacer con estos poderosos instrumentos que nos facilitan muchas tareas sin meternos en escribir lineas de codigos .

Trabajo futuro:

1. Se puede añadir al visualizador de KIbana varias vistas :
    a. resultado de las votaciones de forma gráfica
2. Desplegar el sistema en un servidor de producción.