

# Gestión de Logs

ELK

# Introducción

# ¿Qué son los logs?

En informática, se usa el término registro, log o historial de log para referirse a la grabación secuencial en un archivo o en una base de datos de todos los acontecimientos (eventos o acciones) que afectan a un proceso particular (aplicación, actividad de una red informática, etc.). De esta forma constituye una evidencia del comportamiento del sistema.

```
2016-02-26 15:26:07.652737 I [1917:Puma 001] [127.0.0.1] [jbloggs] Rails — Started GET "/groups" for 127.0.0.1 at 2016-02-26 15:26:07 +0000
2016-02-26 15:26:07.830436 I [1917:Puma 001] [127.0.0.1] [jbloggs] GroupsController — Processing #index
2016-02-26 15:26:08.089099 I [1917:Puma 001] [127.0.0.1] [jbloggs] ActionView::Base — Rendered cascading_gals/_groups_select.html.haml (28.4ms)
2016-02-26 15:26:08.104141 I [1917:Puma 001] [127.0.0.1] [jbloggs] ActionView::Base — Rendered search/_search_form.html.haml (74.3ms)
2016-02-26 15:26:08.204800 I [1917:Puma 001] [127.0.0.1] [jbloggs] ActionView::Base — Rendered action_links/_crud.html.haml (11.8ms)
2016-02-26 15:26:08.205121 I [1917:Puma 001] [127.0.0.1] [jbloggs] ActionView::Base — Rendered groups/_group.html.haml (39.3ms)
2016-02-26 15:26:08.213000 I [1917:Puma 001] [127.0.0.1] [jbloggs] ActionView::Base — Rendered groups/index.html.haml within layouts/application (212.3ms)
2016-02-26 15:26:08.573078 I [1917:Puma 001] [127.0.0.1] [jbloggs] ActionView::Base — Rendered layouts/_ie_conditional.html.haml (116.2ms)
2016-02-26 15:26:08.700857 I [1917:Puma 001] [127.0.0.1] [jbloggs] ActionView::Base — Rendered layouts/_menu.html.haml (125.9ms)
```

# ¿Para qué guardamos logs?

Encontrar errores y sus trazas.

Explicar el comportamiento del sistema en su conjunto.

Detectar posibles riesgos y amenazas (hackers, fallos repetitivos, etc)

Conocer el comportamiento de los usuarios

Obtener datos estadísticos de uso de la aplicación.

# ¿Cómo vemos los logs?

grep

tail

cat/less/more

vim/notepad

# Desafíos en el manejo de logs

Generalmente tenemos muchas aplicaciones funcionando en paralelo y reportando sus logs. (servidores, app, sistema)

Pasar de un manejo de logs a un análisis de los logs

¿Cómo hacemos para conectar logs de distintas fuentes?

¿Cómo hacemos para referencias y correlacionar esta información?

# Necesidades para hacer análisis de logs

- Poder fácilmente conectar distintas fuentes de logs
- Correlacionarlos y referenciarlos
- Buscar, filtrar e identificar
- Visualizar lo que está pasando (por ejemplo ctos logs x segundo está generando una determinada aplicación)
- Detección de anomalías y alertas
- ¿Cómo hacemos una retención de los logs sensibles?

# ELK

1. Un **agente de recogida de logs** en nuestras aplicaciones.

**Logstash.**

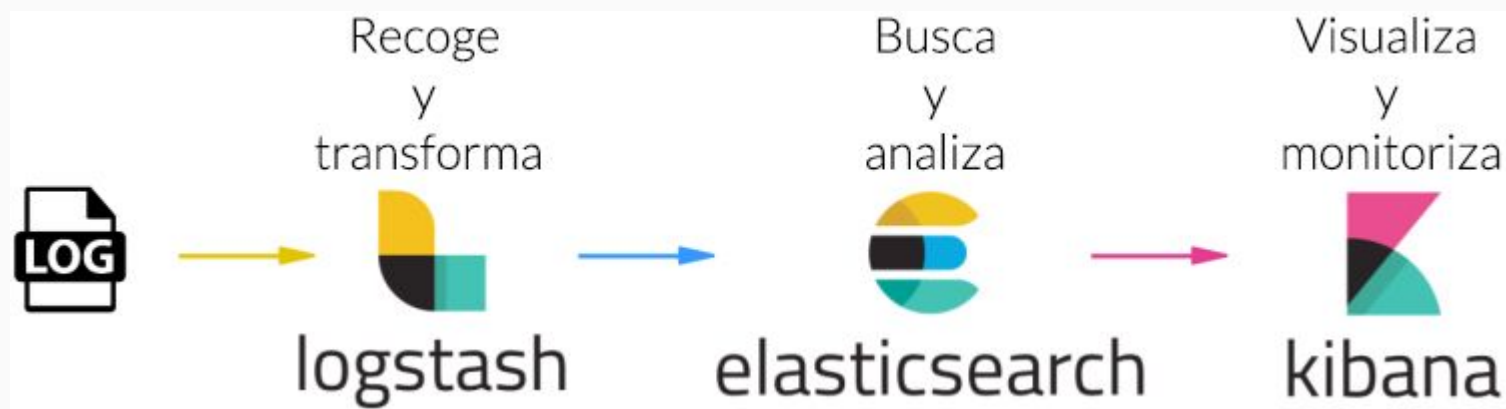
2. Una **base de datos** donde almacenar, indexar y buscar los eventos de log de las aplicaciones.

**Elasticsearch.**

3. Una **aplicación frontend** donde los usuarios puedan consultar los eventos más interesantes y estar informados de cualquier incidencia.

**Kibana.**





# ¿Qué resuelve Logstash?

**Falta de consistencia:** puedo estar recolectando logs con distintos formatos y cantidad de información

```
Mar 23 22:05:24 Macintosh com.apple.launchd[1]
```

```
(h
```

```
120707 0:40:34 4 Connect root@localhost on  
4 Query select @@version_comment limit 1  
120707 0:40:45 4 Query select * from mysql.user  
120707 0:41:18 5 Query hello world
```

```
120707 0:37:09 InnoDB: Mutexes and rw_locks use GCC atomic builtins  
120707 0:37:09 InnoDB: Compressed tables use zlib 1.2.5  
120707 0:37:09 InnoDB: Using Linux native AIO  
120707 0:37:09 InnoDB: Initializing buffer pool, size = 128.0M  
120707 0:37:09 InnoDB: Completed initialization of buffer pool
```

# ¿Qué resuelve Logstash?

**Formato de tiempo:** los formatos no siempre coinciden y el tiempo suele ser fundamental para analizar un proceso.

130460505

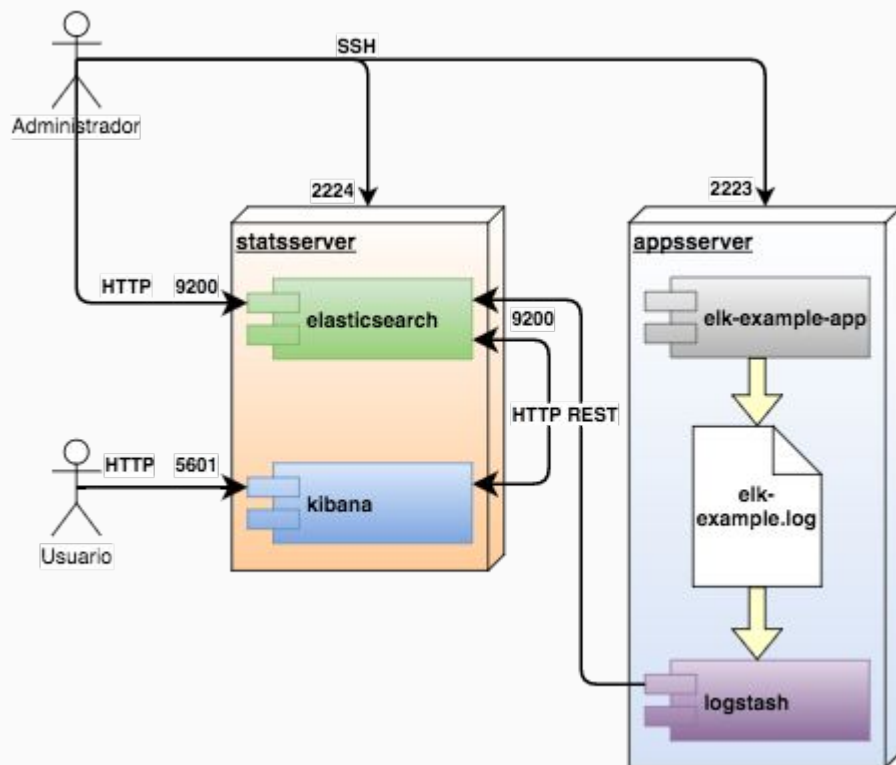
Oct 11 20:21:47

@4000000037c219bf2ef02e94

[29/Apr/2011:07:05:26 +0000]

020805 13:51:24

# Una arquitectura posible



1. `sudo docker pull sebp/elk`
2. `sudo docker run -p 5601:5601 -p 9200:9200 -p 5044:5044 -it --name elk sebp/elk`
  - 5601 (Kibana web interface).
  - 9200 (Elasticsearch JSON interface).
  - 5044 (Logstash Beats interface)

# 1ro Tenemos que enviar los logs al almacén

En el mundo Elastic existe un agente llamado filebeat

Este tiene módulos para configurarlo, pero también se puede configurar a mano

# Logs en Java - JHipster

# Escribir logs en una clase - Niveles

```
@RestController
public class LoggingController {

    Logger logger = LoggerFactory.getLogger(LoggingController.class);

    @RequestMapping("/")
    public String index() {
        logger.trace("A TRACE Message");
        logger.debug("A DEBUG Message");
        logger.info("An INFO Message");
        logger.warn("A WARN Message");
        logger.error("An ERROR Message");

        return "Howdy! Check out the Logs to see the output...";
    }
}
```



# Configuraciones

Para crear logs en archivos debemos modificar la configuración en:

**logback-spring.xml**

También podremos cambiar los niveles de log por defecto.

Con esta opción podremos utilizar filebeat en nuestro sistema para enviar los logs a elk. Junto con logs del sistema operativo y otras apps del entorno.

# Enviar logs a Logstash

`application-dev.yml` or `application-prod.yml` :

```
jhipster:
  logging:
    logstash:
      enabled: true
      host: localhost
      port: 5000
      queueSize: 512
```

# Configurar Logstash (logstash.conf)

```
input {  
  tcp {  
    port => "5000"  
    type => syslog  
    codec => json_lines  
  }  
}  
  
output {  
  elasticsearch {  
    hosts => ["${ELASTICSEARCH_HOST}:${ELASTICSEARCH_PORT}"]  
    index => "logs-%{+YYYY.MM.dd}"  
  }  
}
```

En el docker elk /opt/logstash/config/

# Docker ELK

```
docker run -p 5601:5601 -p 9200:9200 -p 5044:5044 -p 5000:5000 -it --name elk sebp/elk
```

- 5601 (interfaz web de Kibana).
- 9200 (interfaz Elasticsearch JSON).
- 5044 (interfaz de Logstash Beats, recibe registros de Beats como Filebeat; consulte los registros de reenvío con la sección Filebeat).
- 5000 (Interface Logstash Lumberjack, recibe registros de los reenviadores de Logstash; consulte los registros de reenvío con la sección de reenviadores de Logstash).

# File Beat

File Beat capturará los logs de los archivos del sistema y lo enviará a Logstash

Si lo hacemos en Docker

<https://www.elastic.co/guide/en/beats/filebeat/current/running-on-docker.html>

# Crear un docker de la aplicación

```
./mvnw package -Pprod verify jib:dockerBuild
```

```
docker-compose -f src/main/docker/app.yml up
```