6.2 Instalación y Funciones

Al igual que Elasticsearch, **Logstash** necesita *Java* para funcionar por lo que serán necesarios los mismos pasos. A continuación se muestran:

Instalación de Java

El primer paso será instalar Java. En nuestro caso instalaremos la versión v8 soportada y recomendada.

```
add-apt-repository ppa:webupd8team/java
apt-get update
apt-get install oracle-java8-installer
```

Se recomienda también cambiar la variable de entorno para especificar la ruta correcta de JAVA. Será necesario modificar el archivo `/etc/environment` y añadir la siguiente línea:

```
JAVA_HOME="/usr/lib/jvm/java-8-oracle"
```

Y recargar las variables para que use la última introducida con el comando:

```
source /etc/environment
```

Instalación de Logstash

Se importa la clave PGP

```
wget -qO - https://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo apt-key add -
```

Será necesario tener instalado el paquete `apt-transport-https` y añadir el repositorio de Elastic.

```
sudo apt-get install apt-transport-https
echo "deb https://artifacts.elastic.co/packages/6.x/apt stable main" | sudo tee -a
 /etc/apt/sources.list.d/elastic-6.x.list
```

Finalmente se puede instalar Logstash:

```
sudo apt-get update && sudo apt-get install logstash
```

- **Primer procesado**

  bin/logstash -e 'input { stdin { } } output { stdout {} }'

- **Primer fichero logstash.conf**

```
input {
  file {
    path => "/home/elastic/Documents/datos.json"
    start_position => "beginning"
    codec => "json"
  }
}

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

Para la ejecución de Logstash utilizando un fichero se deberá utilizar el siguiente comando:

```
/usr/share/logstash/bin/logstash -f /etc/logstash/conf.d/logstash.conf --path.sett
ings=/etc/logstash
```

- **Mutate**

```
filter {
  mutate {
    remove_field => [ "@version" ]
    gsub => ["surname", " - ", ""]
  }
}
```

Input:

```
echo '{ "name":"David", "surname":" - Sanchez", "company":"ProveedorA", "money":"3
00" }' >> "/home/elastic/Documents/datos.json"
```

- **If - Else**

```
filter {
  mutate {
    remove_field => [ "@version" ]
    gsub => ["surname", " - ", ""]
  }

  if [company] =~ /^Prov*/ {
    mutate {
      add_field => { "user" => "Proveedor" }
    }
  } else {
    mutate {
      add_field => { "user" => "Client" }
    }
  }
}
```

Input:

```
echo '{ "name":"Jesus", "surname":"Hernandez", "company":"Empresa1SA", "money":"10
0" }' >> "/home/elastic/Documents/datos.json"
```

- **Grok**

```
filter {
   grok {
     match => { "message" => [ "factura: %{WORD:name}-%{WORD:surname} \[%{WORD:com
pany}\] %{NUMBER:money}" ] }
   }
}
```

Input:

```
echo '{ "message":"factura: Pedro-Lopez [Empresa2] 40" }' >> "/home/elastic/Docume
nts/datos.json"
```

- **Cidr & Geoip**

```
filter {
    if [srcip] and [srcip] != "N/A" {
        cidr {
                add_tag => ["src_ip_priv"]
                address => ["%{srcip}"]
                network => ["172.16.0.0/12","10.0.0.0/8","192.168.0.0/16","169.254
.0.0/16","0.0.0.0/32"]
        }
        if "src_ip_priv" not in [tags] {
                geoip {
                        target => "src_geoip"
                        source => "srcip"
                        fields => ["city_name", "continent_code", "country_code2",
 "country_code3", "country_name", "ip", "latitude", "longitude", "location"]
                }
        }
    }
}
```

Input:

```
echo '{ "message":"factura: Pedro-Lopez [Empresa2] 40", "srcip":"119.176.103.117"
}' >> "/home/elastic/Documents/datos.json"
echo '{ "message":"factura: Marta-Costa [ProveedorB] 10", "srcip":"192.168.1.10" }
' >> "/home/elastic/Documents/datos.json"
```