10.3 Alerting

Otra de las funcionalidades más interesantes a la hora de trabajar con Elasticsearch y X-pack es **Alerting**. Con ella, podremos crear consultas que si cumplen una cierta condición se ejecute una acción, es decir, si el resultado a esta consulta es `X`, quiero que se me notifique mediante correo por ejemplo.

En nuestro caso, para la prueba de concepto podemos realizar la monitorización de una página web mediante Heartbeat, y que se nos notifique en el caso de que se caiga la misma.

Levantar la página web

Para levantar una web de pruebas de forma rápida se puede utilizar Apache:

```
apt install apache2
service apache2 start
```

Para comprobar que se ha subido correctamente bastará con acceder a través del navegador web, por ejemplo: http://192.168.1.31.

Instalación y configuración de Heartbeat

Al igual que se realizó en su correspondiente tema de Beats, deberemos descargar este componente e instalarlo:

```
apt install heartbeat-elastic
sudo update-rc.d heartbeat-elastic defaults 95 10
```

y la configuración de `heartbeat.yml` deberá ser monitorizando la web previamente levantada y enviando los datos a Elasticsearch.

```
  # List or urls to query
  urls: ["http://192.168.1.30:80"]

  setup.dashboards.enabled: true

  setup.kibana:
  # Kibana Host
  host: "192.168.1.31:5601"

output.elasticsearch:
  # Array of hosts to connect to.
  hosts: ["192.168.1.31:9200"]
```

Y ejecutar el servicio mediante:

```
service heartbeat-elastic start
```

Configuración del correo en Elasticsearch

Para que Elasticsearch pueda enviar correos o notificaciones, deberán configurarse desde `elasticsearch.yml`. En nuestro caso, las pruebas se harán enviando correos y para ello se deberá configurar un servidor de correo, por ejemplo, un Postfix en local:

```
apt-get install mailutils
```

Y modificar las siguientes líneas del archivo de configuración `/etc/postfix/main.cf`:

```
smtpd_use_tls=no
mynetworks = ... 192.168.1.0/24
```

Reiniciando posteriormente el servicio para que se apliquen los cambios:

```
sudo service postfix restart
```

Una buena forma de probar si el servicio funciona correctamente será mediante el siguiente comando, sustituyendo el final por el correo al que se quiera enviar:

```
echo "This is the body of the email" | mail -s "This is the subject line" aaaaaa@d
omain.com
```

Si el correo llega, se podrá configurar Elasticsearch para que haga uso de dicho servicio añadiendo las siguientes líneas a `elasticsearch.yml`:

```
xpack.notification.email.account:
    company_account:
        profile: company
        smtp:
            auth: false
            host: "192.168.1.31"
            port: 25
```

Alerta

La alerta será la siguiente:

```
PUT _xpack/watcher/watch/test
{
```

```
  "trigger": {
    "schedule": {
      "interval": "1m"
    }
  },
  "input": {
    "search": {
      "request": {
        "indices": "heartbeat-*",
        "types": "doc",
        "body": {
          "query": {
            "bool": {
              "should": [
                {"wildcard": { "http.url": "*192.168.1.30*" }}
              ]
            }
          },
          "sort": [
            {
              "@timestamp": {
                "order": "desc"
              }
            }
          ],
          "size": 1
        }
      }
    }
  },
  "condition": {
    "compare": {
      "ctx.payload.hits.hits.0._source.monitor.status": {
        "eq": "down"
      }
    }
  },
  "actions" : {
    "send_email" : {
    "throttle_period": "15m",
      "email" : {
        "from" : "root@elastic01",
        "to" : "aaaaaa@domain.com",
        "subject" : "ALERT APACHE: Server Down",
        "body" : "Se han detectado eventos de una posible caida del servidor {{ctx
.payload.hits.hits.0._source.http.url}}  \n  - Hora del evento: {{ctx.payload.hits
.hits.0._source.@timestamp}} \n"
      }
```

```
    }
  }
}
```

Para ejecutarla:

```
POST _xpack/watcher/watch/test/_execute
```

En el caso de que hubiera que borrarla para cambiar cualquier parámetro:

```
DELETE _xpack/watcher/watch/test
```