DevOps Sesión 15 (2022-03-30) ELK

Documentación relacionada

> - ./5-Topic 705 Service Operations/Material Curso ELK/1-Laboratorios ELK.pdf
> - ./5-Topic 705 Service Operations/Presentacion Herramientas para el manejo de logs.pdf
> - ./5-Topic 705 Service Operations/Clase Monitorizacion.txt

packetbeat

- https://www.elastic.co/guide/en/beats/packetbeat/current/index.html
- https://www.elastic.co/es/downloads/beats/packetbeat
- ./5-Topic 705 Service Operations/Material Curso ELK/1-Laboratorios ELK.pdf pag 37

```
rpm -ivh /root/packetbeat-7.0.0-x86_64.rpm
```

- ; /etc/packetbeat/packetbeat.yml

```
14 packetbeat.interfaces.device: any # packetbeat devices (se pueden usar nombres o posición)
131 setup.dashboards.enabled: true
149   host: "192.168.93.128:5601"
223 xpack.monitoring.enabled: true
230 xpack.monitoring.elasticsearch:
```

```
packetbeat devices
packetbeat test config -c packetbeat.yml
systemctl start packetbeat.service
systemctl status packetbeat.service
```

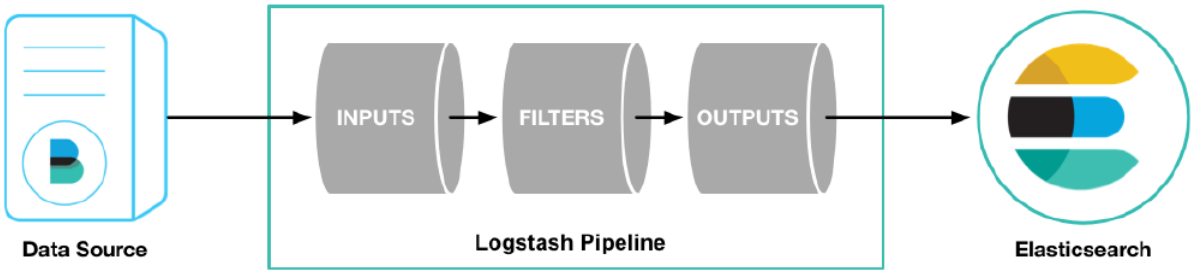
- En Kibana, vamos **Dashboard** y buscamos **Packetbeat Flows ECS**

auditbeat

- ./5-Topic 705 Service Operations/Material Curso ELK/1-Laboratorios ELK.pdf pag 55

logstash

- ./5-Topic 705 Service Operations/Clase Monitorizacion.txt línea 151
- ./5-Topic 705 Service Operations/Material Curso ELK/1-Laboratorios ELK.pdf pag 59
- preprocesador



- codec: es una transformación en la salida que va a hacer logstash con la información que estemos trabajando, es decir un codec es si a mi me llega en un formato que logstash ya entiende lo que puede hacer logstash es enviarla en un formato concreto, por ejemplo ahora utiliza el codec de JSON para que me la devuelva en este formato.

lab

- ```
  input {
      stdin {}
  }

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

  → convierte la entrada de teclado en cadenas JSON (y más info)

- ```
  cp /elk/example.conf  /etc/logstash/
  /usr/share/logstash/bin/logstash -f /etc/logstash/example.conf
  ```

lab codec multiline

- ```
  input {
      stdin {
          codec => multiline {
              pattern => "^fin"
              negate => "true"
              what => "next"
          }
      }
  }

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

- ```
  cp /elk/example-codec-multiline.conf /etc/logstash/
  ```

```
/usr/share/logstash/bin/logstash -f /etc/logstash/example-codec-multiline.conf
```

lab file

- https://www.elastic.co/guide/en/logstash/current/plugins-inputs-file.html
- example-codec-file.conf

```
input {
    file {
        path => "/logs/access_log"
        exclude => "*.gz"
        start_position => "beginning"
        sincedb_path => "/logs/access.sincedb"
    }
}

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

    - **sincedb_path**: no rutas relativas, puntero que indica la posición «leída» (por si se corta)
- `/usr/share/logstash/bin/logstash -f /etc/logstash/example-codec-file.conf`
- example-codec-file-2.conf

```
input {
    file {
        path => "/logs/log-generator*.log"
        exclude => "*.gz"
        start_position => "beginning"
        sincedb_path => "/logs/log-generator.sincedb"
        codec => multiline {
            pattern => "^(DEBUG|INFO|ERROR|TRACE|FATAL|WARN).*"
            negate => "true"
            what => "previous"
        }
    }
}

output {
    stdout {
        # codec => json_lines
    }
}
```

- `/usr/share/logstash/bin/logstash -f /etc/logstash/example-codec-file-2.conf`

lab filebeat

- `rpm -ivh /root/filebeat-6.7.1-x86_64.rpm`

- ```
  29     - /logs/log-generator.log
  149 #output.elasticsearch:
  150   # Array of hosts to connect to.
  151   #hosts: ["localhost:9200"]
  162 output.logstash:
  163   # The Logstash hosts
  164   hosts: ["localhost:5044"]
  ```

- example-beat.conf

  ```
  input {
      beats {
          port => 5044
      }
  }

  output {
      stdout {}
  }
  ```

- ```
  systemctl start filebeat.service
  /usr/share/logstash/bin/logstash -f /etc/logstash/example-beat.conf
  ```

filters: grok

- https://www.elastic.co/guide/en/logstash/current/plugins-filters-grok.html
- https://grokdebug.herokuapp.com/
- Kibana → Dev Tools → Grok Debugger
- https://programmerclick.com/article/74971006708/
- https://github.com/logstash-plugins/logstash-patterns-core/tree/main/patterns/ecs-v1
- https://github.com/logstash-plugins/logstash-patterns-core/tree/main/patterns

inputs

filters

outputs

TODO