

# Elastic ELK Stack

Workshop – Projeto Connor



# Programação

01

## Histórico

História – ELK Stack

03

## Beats & Logstash

Instalação, e Utilização

02

## Elasticsearch

Instalação, e Utilização

04

## Kibana

Instalação, e Utilização

# ELK Stack Histórico





# COMPANY HISTORY





# Netflix

Processo de notificações

# Uber


Marketplace

# Slack

Security

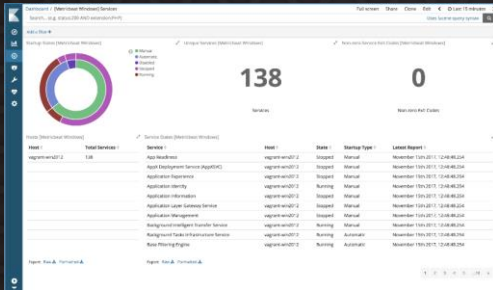
# Microsoft

Azure Monitoring

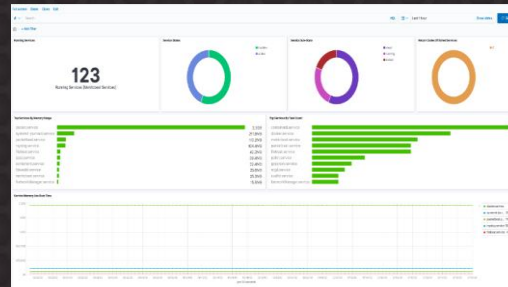


# Integrações

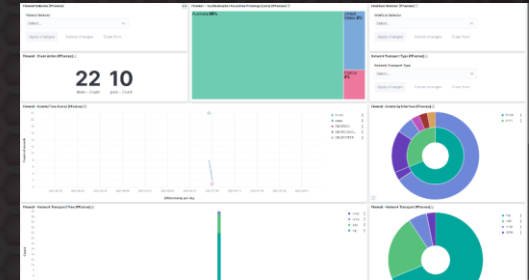
## 277 integrações com diferente plataformas



Windows OS



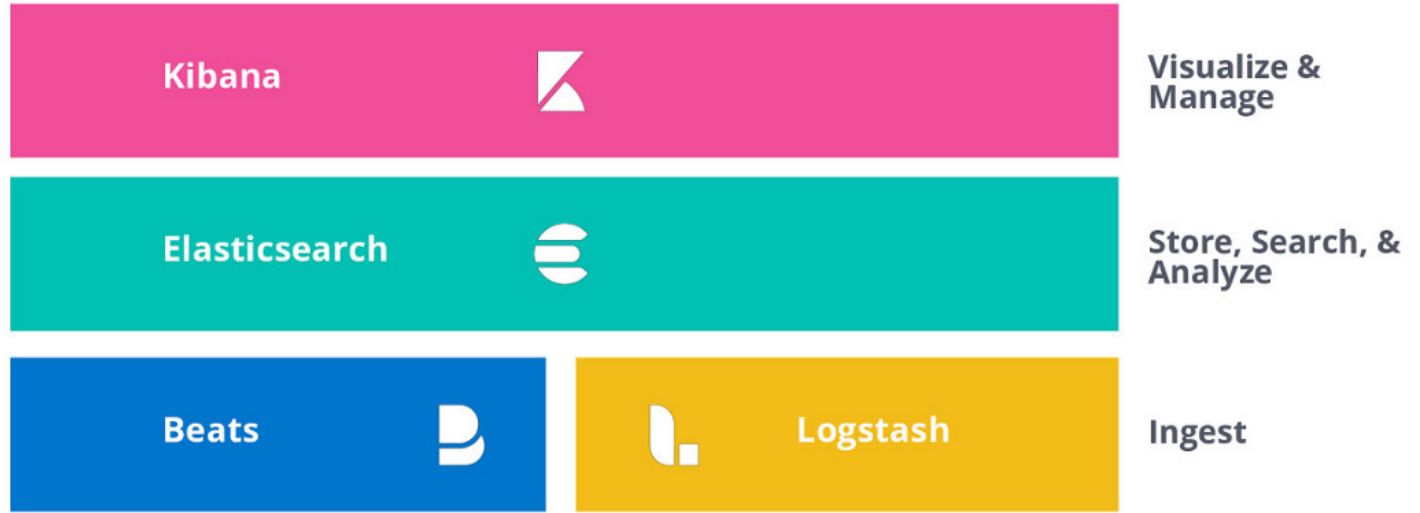
Linux Server



PfSense

# Elastic Stack

## SOLUTIONS



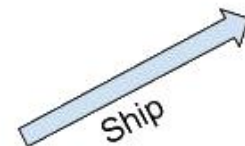
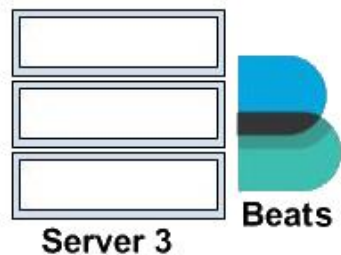
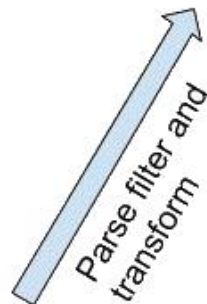
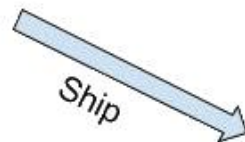
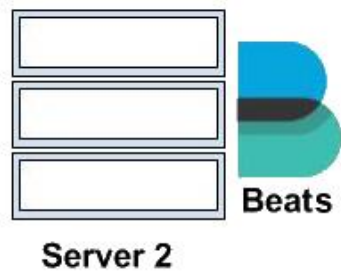
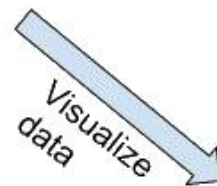
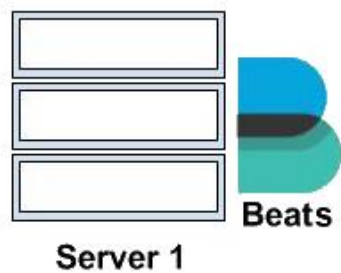
### SaaS



### SELF-MANAGED









# ELK Stack Logstash



# ELK Stack Instalação



# Instalação Elasticsearch

Configuração VM:

- 6 Gb RAM
- 20 Gb HD
- 2 CPU



# Instalação Elasticsearch

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

```
sudo apt-get install apt-transport-https
```

```
echo "deb https://artifacts.elastic.co/packages/7.x/apt stable main" | sudo  
tee -a /etc/apt/sources.list.d/elasticsearch-7.x.list
```

```
sudo apt-get update
```

```
sudo apt-get install elasticsearch
```

```
nano /etc/elasticsearch/elasticsearch.yml
```





# Instalação Elasticsearch

```
-node.name: node-1  
-network.host: 0.0.0.0  
-discovery.seeds_hosts: ["127.0.0.1"]  
-cluster.initial_master_nodes: ["node-1"]
```



# Instalação Elasticsearch

```
sudo /bin/systemctl daemon-reload
```

```
sudo /bin/systemctl enable elasticsearch.service
```

```
sudo systemctl start elasticsearch.service
```



# Instalação Kibana

```
sudo apt-get install kibana
```

```
nano /etc/kibana/kibana.yml
```

```
nano /etc/kibana/kibana.yml
```



# Instalação Kibana

```
-server.port: 5601  
-server.host: "0.0.0.0"  
-elasticsearch.hosts["http://127.0.0.1:9200"]
```





# Instalação Kibana

```
sudo /bin/systemctl daemon-reload
```

```
sudo /bin/systemctl enable kibana.service
```

```
sudo systemctl start kibana.service
```



# Instalação Logstash

```
sudo apt-get install logstash
```

```
cd /etc/logstash/
```

```
sudo /bin/systemctl enable logstash
```

```
sudo systemctl start logstash
```



# Instalação do Beats

- `curl -L -O https://artifacts.elastic.co/downloads/beats/filebeat/filebeat-oss-7.17.6-amd64.deb`
- `sudo dpkg -i filebeat-oss-7.17.6-amd64.deb`
- `sudo filebeat modules list`
- `sudo filebeat modules enable system`
- `sudo filebeat -e -c /etc/filebeat/filebeat.yml`
- `sudo systemctl enable filebeat`
- `sudo systemctl start filebeat`

```

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

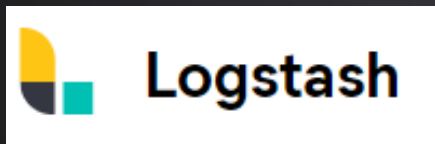
# Protocol - either `http` (default) or `https`.
#protocol: "https"

# Authentication credentials - either API key or username
#api_key: "id:api_key"
#username: "elastic"
#password: "changeme"

# ----- Logstash Output -----
output.logstash:
# The Logstash hosts
hosts: ["localhost:5044"]
```

# Logstash

O Logstash é um pipeline gratuito e aberto de processamento de dados do lado do servidor que faz a ingestão de dados de inúmeras fontes, transforma-os e envia-os para o Elasticsearch.





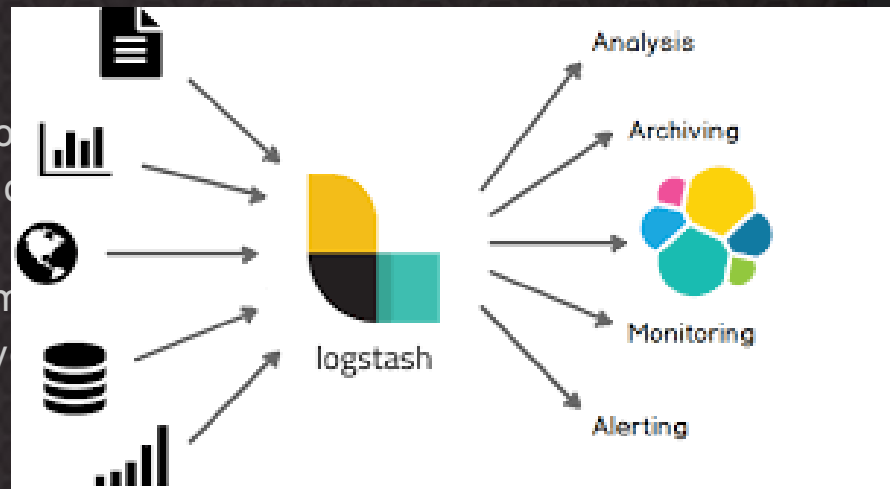
# Aplicação do Logstash

- Manipular logs
- Coletar dados em tempo real
- Manipular e tratar dados
- Envio para Elasticsearch
- Recebe, normaliza e envia dados de múltiplas fontes
- Plugins



# Funcionamento

- Entradas, filtros e saídas
- O Logstash faz dinamicamente consumo independentemente do formato e da complexidade
- Estrutura a partir de dados não estruturados com endereços IP, anonimiza ou exclui campos sensíveis



```
input {
  file {
    path => ["/var/log/auth.log"]
  }
}
filter {
  grok {
    match => ["message", "%{SYSLOGTIMESTAMP:syslog_timestamp}"]
    overwrite => ["message"]
  }
  mutate {
    rename => ["@timestamp","time"]
    gsub => ["message", "[\\\"\\\""]",""]
  }
}
output {
  elasticsearch {
    hosts => ["http://localhost:9200"]
    index => "logs-%{+YYYY.MM.dd}"
    document_type => "ubuntu_logs"
  }
  stdout{}
}
```

- conf.d
- logstash.yml

# Conceitos

- Pipeline
- JSON
- Estrutura de um Log
- Expressões Regulares (Regex)
- <https://github.com/hpcugent/logstash-patterns/blob/master/files/grok-patterns>

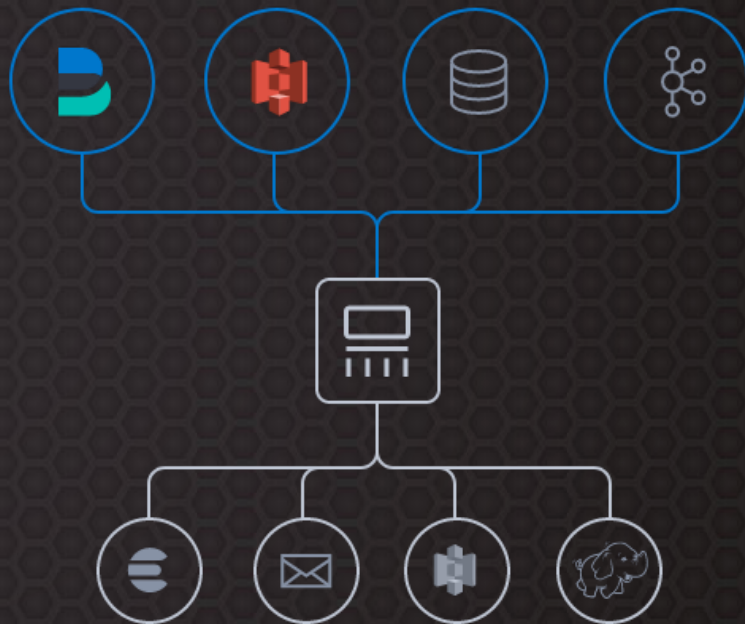
*May 20 11:37:47 felipe-VirtualBox sudo: pam\_unix(sudo:session): session opened for user root by felipe(uid=0)*

no dia 20 de maio, às 11h37, o usuário *felipe*, de UID 0, requisitou na máquina *felipe-VirtualBox* as permissões de superusuário (*root*) por meio do comando *sudo*.



# Entradas

Os dados podem estar dispersos ou isolados em muitos sistemas e em muitos formatos. O Logstash oferece suporte a **uma variedade de entradas** que importa eventos de inúmeras fontes comuns, tudo ao mesmo tempo. Consome logs, métricas, aplicativos web, armazenamentos de dados e vários serviços da AWS, tudo em um fluxo contínuo.



# Entradas

- **HTTP:** Expor uma interface http para receber dados como JSON de outras aplicações.
- **File:** Consumir dados de um arquivo.

Dentre outras possibilidades que podem ser consumidas na referência abaixo.

ref: <https://www.elastic.co/guide/en/logstash/current/input-plugins.html>

```
input {  
  file {  
    path => "/etc/logstash/iris.csv"  
    start_position => "beginning"  
    sincedb_path => "NULL"  
  }  
}
```

# HTTP

- `cd /etc/logstash/conf.d`
- `touch http.conf`
- `nano http.conf`
- `systemctl restart logstash`
- `cd /usr/share/logstash`
- `sudo sudo bin/logstash -f /etc/logstash/conf.d/http.conf`

```
input {  
  http {  
    host => "0.0.0.0"  
    port => "8080"  
  }  
}  
output {  
  elasticsearch {  
    hosts => ["http://localhost:9200"]  
    index => "http-%{+YYYY.MM.dd}"  
  }  
  stdout{}  
}
```





# FILE - CSV

- `cd /etc/logstash/conf.d`
- `touch csv.conf`
- `nano csv.conf`
- `systemctl restart logstash`
- `cd /usr/share/logstash`
- `sudo sudo bin/logstash -f /etc/logstash/conf.d/csv.conf`

```
{
  "@version" => "1",
  "PetalWidthCm" => "2.3",
  "PetalLengthCm" => "5.4",
  "Species" => "Iris-virginica",
  "SepalLengthCm" => "6.2",
  "@timestamp" => 2022-10-05T21:50:00.672Z,
  "message" => "149,6.2,3.4,5.4,2.3,Iris-virginica",
  "path" => "/etc/logstash/iris.csv",
  "id" => "149",
  "SepalWidthCm" => "3.4",
  "host" => "es"
}
{
  "@version" => "1",
  "PetalWidthCm" => "1.8",
  "PetalLengthCm" => "5.1",
  "Species" => "Iris-virginica",
  "SepalLengthCm" => "5.9",
  "@timestamp" => 2022-10-05T21:50:00.672Z,
  "message" => "150,5.9,3.0,5.1,1.8,Iris-virginica",
  "path" => "/etc/logstash/iris.csv",
  "id" => "150",
  "SepalWidthCm" => "3.0",
  "host" => "es"
}
```

```
input {
  file {
    path => "/etc/logstash/iris.csv"
    start_position => "beginning"
    sincedb_path => "NULL"
  }
}
filter {
  csv {
    separator => ",",
    columns => ["Id", "SepalLengthCm", "SepalWidthCm", "PetalLengthCm",
  }
}
output {
  elasticsearch {
    hosts => "http://localhost:9200"
    index => "iris"
  }
  stdout {}
}
```





# FILE - log

```

{
    "host" => "es",
    "tags" => [
        [0] "_grokparsefailure"
    ],
    "message" => "Oct  5 21:44:25 es sudo: pam_unix(sudo:session): session opened for user root
=0) by steh(uid=0)",
    "@version" => "1",
    "time" => 2022-10-05T21:44:56.490Z,
    "path" => "/var/log/auth.log"
}

{
    "host" => "es",
    "tags" => [
        [0] "_grokparsefailure"
    ],
    "message" => "Oct  5 21:44:25 es sudo:      root : TTY=pts/1 ; PWD=/usr/share/logstash ; U
ot ; COMMAND=bin/logstash -f /etc/logstash/logs.conf",
    "@version" => "1",
    "time" => 2022-10-05T21:44:56.490Z,
    "path" => "/var/log/auth.log"
}

{
    "host" => "es",
    "tags" => [
        [0] "_grokparsefailure"
    ],
    "message" => "Oct  5 21:44:25 es sudo: pam_unix(sudo:session): session opened for user root
=0) by root(uid=0)",
    "@version" => "1",
    "time" => 2022-10-05T21:44:56.490Z,
    "path" => "/var/log/auth.log"
}

```

# Tipos de Log

- `/etc/mail/maillog` = Arquivo que registra os logs do Servidor de E-mails;
- `/var/log/messages` = Contém registros de acesso ao sistema e em alguns casos registros do IPTABLES;
- `/var/log/httpd/(access, error ou agent.log)` = Logs do Servidor Web Apache;
- `/var/log/lpr.log` = logs de impressoras;
- `/etc/log/daemon.log` = Logs de serviços em geral;
- `/var/log/syslog`: log do sistema;
- `/var/log/auth.log`: log de autenticação;
- `/var/log/kern.log`: log do kernel;
- `/var/log/cron.log`: log crond;
- `/var/log/lighttpd`: log de erro e acesso a Lighttpd;
- `/var/log/boot.log`: registro de inicialização do sistema;
- `/var/log/mysqld.log`: registro de banco de dados MySQL;



Tcp

Beats

Snmp

```
input {
  tcp {
    port => 5000
    type => syslog

    codec => multiline {
      pattern => "^\{TIMESTAMP_ISO8601} "
      negate => true
      what => "previous"
    }
  }
}

output {
  elasticsearch {
    hosts => "elasticsearch:9200"
  }
}
```

```
input {
  snmptrap {
    port => "162"
    community => ["public"]
    codec => plain{ charset => "UTF-8" }
  }
}

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

# Filtros

À medida que os dados trafegam da fonte ao armazenamento, os filtros do Logstash analisam cada evento, identificam campos nomeados para desenvolver a estrutura e os transformam para convergirem em um formato comum para proporcionar uma análise.

- O Logstash transforma e prepara dinamicamente os dados, independentemente do formato ou da complexidade
- Obter a estrutura de dados não estruturados com o grok
- Decifrar coordenadas geográficas de endereços IP
- Anonimizar os dados, excluir os campos sensíveis completamente





# Filtros

- **Grok:** Um plugin que é feito para transformar dados não estruturados em um dado estruturado e com a possibilidade de receber queries.
- **Date:** Usado para formatar o campo de data.
- **CSV:** Pega os dados de um CSV e coloca em campos individuais.

ref: <https://www.elastic.co/guide/en/logstash/current/filter-plugins.html>



# Grok

```
match => { "message" => [  
    '%{TIMESTAMP_ISO8601:time} %{LOGLEVEL:logLevel}  
    %{GREEDYDATA:logMessage}',  
    '%{IP:clientIP} %{WORD:httpMethod} %{URIPATH:url}'  
  ] }  
}  
}
```

Date

CSV

```
filter {
  if [type] == "syslog" {
    grok {
      match => { "message" => "%{SYSLOGTIMESTAMP:syslog_timestamp} %{SYSLOGHOST:syslog_host} %{SYSLOGMESSAGE:message}" }
      add_field => [ "received_at", "%{@timestamp}" ]
      add_field => [ "received_from", "%{host}" ]
    }
    date {
      match => [ "syslog_timestamp", "MMM d HH:mm:ss", "MMM dd HH:mm:ss" ]
    }
  }
}
```

```
filter {
  csv {
    separator => ","
    columns =>
    ["Visit_Status","Time_Delay","City","City_id","Patient_Age",
    "Zipcode","Latitude","Longitude","Pathology","Visiting_Date",
    "Id_type","Id_personal","Number_Home_Visits","Is_Patient_Minor","Geo_point"]
  }
  date {
    match => ["Visiting_Date","dd-MM-YYYY HH:mm"]
    target => "Visiting_Date"
  }
  mutate {convert => ["Number_Home_Visits", "integer"]}
  mutate {convert => ["City_id", "integer"]}
  mutate {convert => ["Id_personal", "integer"]}
  mutate {convert => ["Id_type", "integer"]}
  mutate {convert => ["Zipcode", "integer"]}
}
```

# Saída

O Elasticsearch é a saída mais usada de possibilidades de busca.

O Logstash tem **uma variedade de saídas** que permitem rotear os dados para onde quiser.

```
output {  
  elasticsearch {  
    hosts => ["http://localhost:9200"]  
    index => "logs-%{+YYYY.MM.dd}"  
    document_type => "ubuntu_logs"  
  }  
  stdout{}  
}
```



# ELK Stack Logstash & Beats



# Beats

O Beats é uma plataforma gratuita e aberta para agentes de dados. Eles enviam dados de centenas ou milhares de computadores e sistemas para o Logstash ou o Elasticsearch.



# Instalação do Beats

- `curl -L -O https://artifacts.elastic.co/downloads/beats/filebeat/filebeat-oss-7.17.6-amd64.deb`
- `sudo dpkg -i filebeat-oss-7.17.6-amd64.deb`
- `sudo filebeat modules list`
- `sudo filebeat modules enable system`
- `sudo filebeat -e -c /etc/filebeat/filebeat.yml`
- `sudo systemctl enable filebeat`
- `sudo systemctl start filebeat`

```

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

# Protocol - either `http` (default) or `https`.
#protocol: "https"

# Authentication credentials - either API key or username
#api_key: "id:api_key"
#username: "elastic"
#password: "changeme"

# ----- Logstash Output -----
output.logstash:
# The Logstash hosts
hosts: ["localhost:5044"]
```

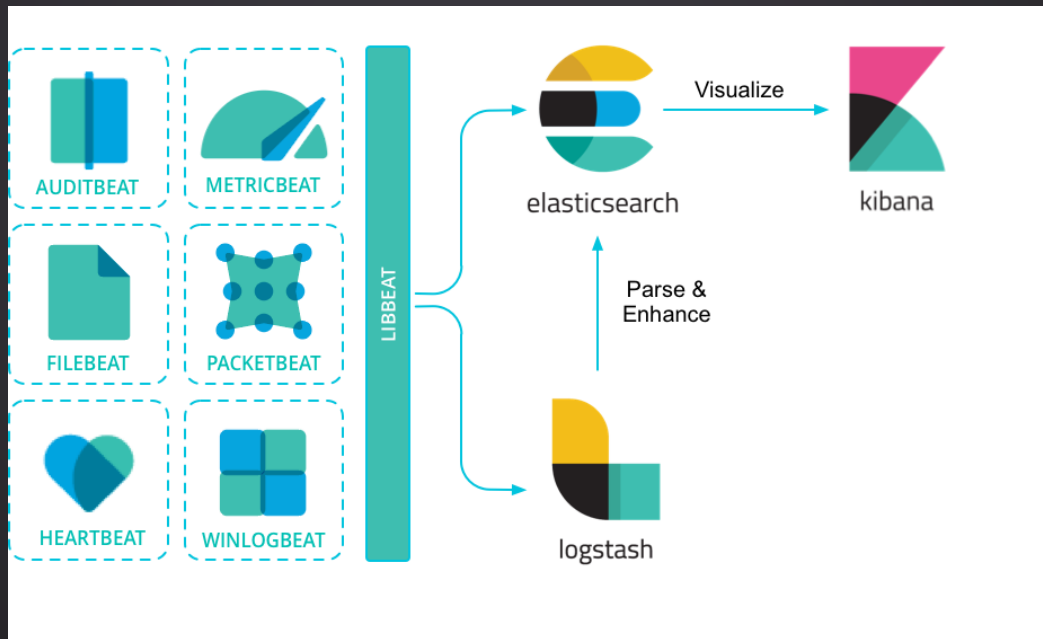
# Para que é utilizado o Beats

- Pequenos agentes leves, que mandam as informações para Elasticsearch ou Logstash
- Tem vários tipos de beats
- Logs, métricas, dados de rede, dados de auditoria, monitoramento uptime
- Criar próprio plugin



# Funcionamento

Os Beats são excelentes para reunir dados. Eles ficam nos servidores, containers, ou são implantados como funções e centralizam os dados no Elasticsearch.



# Tipos



## Filebeat

Arquivos de log



## Metricbeat

Métricas



## Packetbeat

Dados de rede



## Winlogbeat

Logs de evento do Windows



## Auditbeat

Dados de auditoria



## Heartbeat

Monitoramento de  
disponibilidade



## Functionbeat

Agente de envio sem servidor



# Filebeat

- `sudo apt install apache2`
- `sudo ufw allow 'Apache'`
- `sudo systemctl status apache2`
- Volta a máquina para host-only
- `hostname -l`
- `cd /usr/share/filebeat`
- `systemctl restart filebeat`
- `cd /usr/share/logstash`
- `sudo sudo bin/logstash -f /etc/logstash/conf.d/beats.conf`

```
{
  "service" => {
    "type" => "system"
  },
  "@version" => "1",
  "host" => {
    "id" => "4cf8c2bbce57476295a2bd7b9048579e",
    "architecture" => "x86_64",
    "containerized" => false,
    "name" => "es",
    "mac" => [
      [0] "08:00:27:51:35:fa"
    ],
    "hostname" => "es",
    "os" => {
      "kernel" => "5.15.0-48-generic",
      "name" => "Ubuntu",
      "family" => "debian",
      "type" => "linux",
      "codename" => "jammy",
      "version" => "22.04.1 LTS (Jammy Jellyfish)",
      "platform" => "ubuntu"
    },
    "ip" => [
      [0] "192.168.56.101",
      [1] "fe80::a00:27ff:fe51:35fa"
    ]
  },
  "@timestamp" => 2022-10-05T23:41:45.657Z,
  "fileset" => {
    "name" => "auth"
  },
  "tags" => [root@es:/usr/share/logstash# ^C
```

# ELK Stack Elasticsearch







# Definições

- Desenvolvido sob o Apache Lucene
- Busca textual, e fonética por relevância (PT-BR nativo)
- Banco de dados não relacional (noSQL)





# Definições

- Baseado em uma interface REST (Representational State Transfer)
- Utiliza HTTP para comunicar com web services
- Orientado a documento, trabalha com JSON
- Aplicável em qualquer linguagem





# Configurações

1. Config/elasticsearch.yml
2. Config/jvm.options



# Config/elasticsearch.yml

```
GNU nano 6.2 /etc/elasticsearch/elasticsearch.yml S
# ===== Elasticsearch Configuration =====
#
# NOTE: Elasticsearch comes with reasonable defaults for most settings.
#       Before you set out to tweak and tune the configuration, make sure you
#       understand what are you trying to accomplish and the consequences.
#
# The primary way of configuring a node is via this file. This template lists
# the most important settings you may want to configure for a production cluster.
#
# Please consult the documentation for further information on configuration options:
# https://www.elastic.co/guide/en/elasticsearch/reference/index.html
#
# ----- Cluster -----
#
# Use a descriptive name for your cluster:
#
#cluster.name: my-application
#
# ----- Node -----
#
# Use a descriptive name for the node:
#
#node.name: node-1
#
# Add custom attributes to the node:
#
#node.attr.rack: r1
#
# ----- Paths -----
#
# Path to directory where to store the data (separate multiple locations by comma):
#
#path.data: /var/lib/elasticsearch
[ Soft wrapping of overlong lines enabled ]
^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute   ^C Location  M-U Undo
^X Exit      ^R Read File ^_ Replace   ^U Paste     ^J Justify   ^_ Go To Line M-E Redo
```



# Config/jvm.options

```
GNU nano 6.2 /etc/elasticsearch/jvm.options
#####
##
## JVM configuration
##
#####
##
## WARNING: DO NOT EDIT THIS FILE. If you want to override the
## JVM options in this file, or set any additional options, you
## should create one or more files in the jvm.options.d
## directory containing your adjustments.
##
## See https://www.elastic.co/guide/en/elasticsearch/reference/7.17/jvm-options.html
## for more information.
##
#####

#####
## IMPORTANT: JVM heap size
#####
##
## The heap size is automatically configured by Elasticsearch
## based on the available memory in your system and the roles
## each node is configured to fulfill. If specifying heap is
## required, it should be done through a file in jvm.options.d,
## and the min and max should be set to the same value. For
## example, to set the heap to 4 GB, create a new file in the
## jvm.options.d directory containing these lines:
##
## -Xms4g
## -Xmx4g
##
```

# Nós (Nodes)

- Instância operando elasticsearch
- Deve haver apenas uma instância elasticsearch por servidor em ambiente de produção

## Recomendações

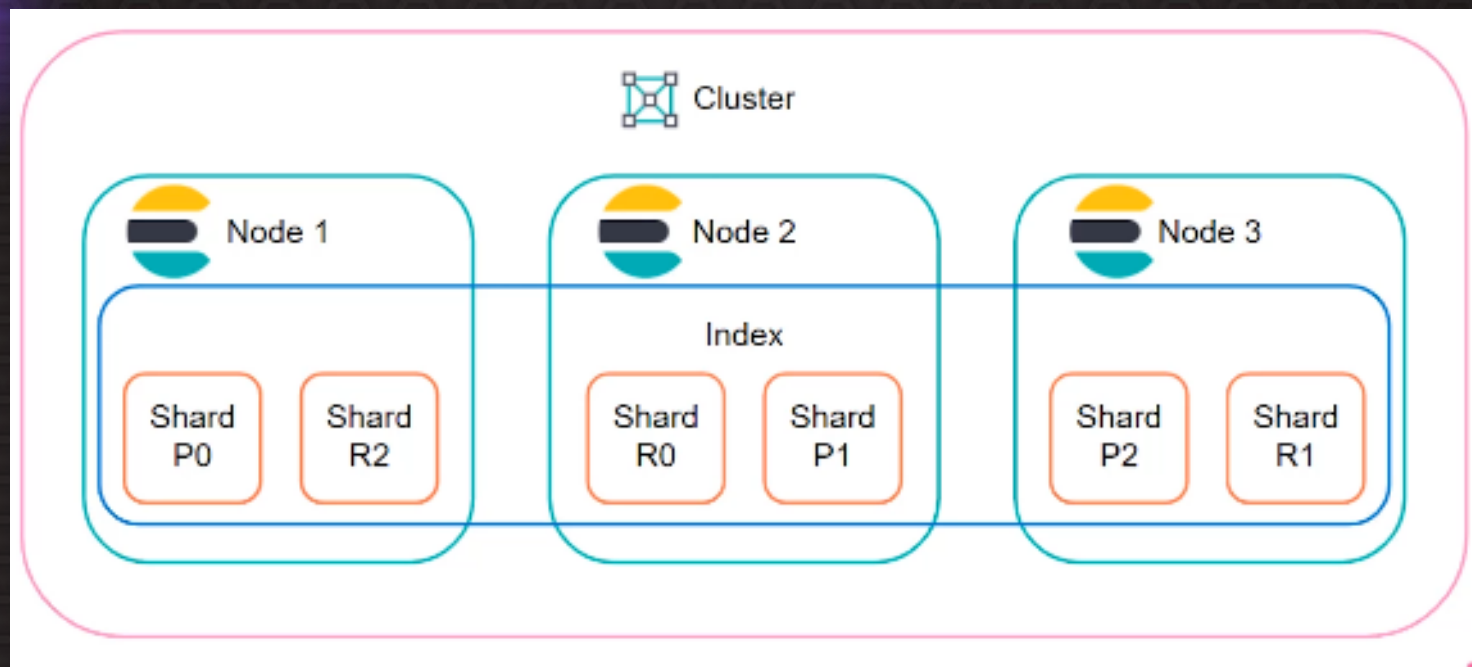
- Metade da memória do servidor para heap
- Não ultrapassar 32Gb de heap
- Configuração mínima e máxima de heap iguais. Xms igual a Xmx



# Cluster

- Um ou mais nós compartilhando o mesmo cluster.name
- Dentro de um cluster os nós possuem diferentes funções:
  - ❖ Master: Responsável pelas configurações e alterações de um cluster elasticsearch
  - ❖ Data: Responsável pelas operações relacionadas a dados
  - ❖ Ingest: Responsável pelos pré-processamentos de dados. Ingest pipelines, \_reindex, etc





```
# ----- Cluster -----  
#  
# Use a descriptive name for your cluster:  
#  
#cluster.name: my-application  
#
```





# Índices

- Namespace que aponta para um ou mais shards

# Shards

- Instância única do Lucene
- Sempre que um índice é criado, um instância única no Lucene é gerada
- Gera índices invertidos
- Cada shard é um motor de busca por si só
- Clientes não acessam o shard diretamente, o acesso é feito pelo índice
- Shard Primário x Réplica





Cluster



Node 1



Node 2



Node 3

Index

Shard  
P0

Shard  
R2

Shard  
R0

Shard  
P1

Shard  
P2

Shard  
R1

# Analogia

- Cluster → Banco de dados
- Índices → Tabelas
- Documentos → linhas



# Testando Elasticsearch

- `curl -XGET 127.0.0.1:9200`

```
root@elkubuntu:/# curl -XGET 127.0.0.1:9200
{
  "name" : "node-1",
  "cluster_name" : "elasticsearch",
  "cluster_uuid" : "DNTPddTUS3amb0V_ymc9fw",
  "version" : {
    "number" : "7.17.6",
    "build_flavor" : "default",
    "build_type" : "deb",
    "build_hash" : "f65e9d338dc1d07b642e14a27f338990148ee5b6",
    "build_date" : "2022-08-23T11:08:48.893373482Z",
    "build_snapshot" : false,
    "lucene_version" : "8.11.1",
    "minimum_wire_compatibility_version" : "6.8.0",
    "minimum_index_compatibility_version" : "6.0.0-beta1"
  },
  "tagline" : "You Know, for Search"
}
root@elkubuntu:/# _
```



# Testando Elasticsearch

## Download schema

- `wget http://media.sundog-soft.com/es7/shakes-mapping.json`

```
root@elkubuntu:/# wget http://media.sundog-soft.com/es7/shakes-mapping.json
--2022-10-06 09:17:40--  http://media.sundog-soft.com/es7/shakes-mapping.json
Resolving media.sundog-soft.com (media.sundog-soft.com)... 54.231.164.137
Connecting to media.sundog-soft.com (media.sundog-soft.com)|54.231.164.137|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 215 [application/octet-stream]
Saving to: 'shakes-mapping.json'

shakes-mapping.json      100%[=====>]          215  --.-KB/s    in 0s

2022-10-06 09:17:41 (51.8 MB/s) - 'shakes-mapping.json' saved [215/215]

root@elkubuntu:/#
```

# Testando Elasticsearch

```
{  
  "mappings" : {  
    "properties" : {  
      "speaker" : { "type": "keyword" },  
      "play_name" : { "type": "keyword" },  
      "line_id" : { "type" : "integer" },  
      "speech_number" : { "type" : "integer" }  
    }  
  }  
}
```



# Testando Elasticsearch

## Download schema

- `wget http://media.sundog-soft.com/es7/shakes-mapping.json`

```
root@elkubuntu:/# wget http://media.sundog-soft.com/es7/shakes-mapping.json
--2022-10-06 09:17:40--  http://media.sundog-soft.com/es7/shakes-mapping.json
Resolving media.sundog-soft.com (media.sundog-soft.com)... 54.231.164.137
Connecting to media.sundog-soft.com (media.sundog-soft.com)|54.231.164.137|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 215 [application/octet-stream]
Saving to: 'shakes-mapping.json'

shakes-mapping.json      100%[=====>]          215  --.-KB/s    in 0s

2022-10-06 09:17:41 (51.8 MB/s) - 'shakes-mapping.json' saved [215/215]

root@elkubuntu:/#
```



# Testando Elasticsearch

Criar o índice:

- `curl -H 'Content-Type: application/json' -XPUT 127.0.0.1:9200/shakespeare --data-binary @shakes-mapping.json`

```
root@elkubuntu:/# curl -H 'Content-Type: application/json' -XPUT 127.0.0.1:9200/shakespeare --data-binary @shakes-mapping.json
{"acknowledged":true,"shards_acknowledged":true,"index":"shakespeare"}root@elkubuntu:/#
```





# Testando Elasticsearch

Baixar obra:

- `wget http://media.sundog-soft.com/es7/Shakespeare_7.0.json`

```
root@elkubuntu:/# wget http://media.sundog-soft.com/es7/shakespeare_7.0.json
--2022-10-06 09:33:49-- http://media.sundog-soft.com/es7/shakespeare_7.0.json
Resolving media.sundog-soft.com (media.sundog-soft.com)... 52.216.161.243
Connecting to media.sundog-soft.com (media.sundog-soft.com)|52.216.161.243|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 25327465 (24M) [application/octet-stream]
Saving to: 'shakespeare_7.0.json'

shakespeare_7.0.json      100%[=====>]  24.15M  7.35MB/s   in 3.9s

2022-10-06 09:33:54 (6.12 MB/s) - 'shakespeare_7.0.json' saved [25327465/25327465]
```



# Testando Elasticsearch

Indexar conteúdo:

- `curl -H 'Content-Type: application/json' -XPOST '127.0.0.1:9200/shakespeare/_bulk?pretty' -data-binary @shakespeare_7.0.json`

```
{
  "index" : {
    "_index" : "shakespeare",
    "_type" : "_doc",
    "_id" : "3710",
    "_version" : 1,
    "result" : "created",
    "_shards" : {
      "total" : 2,
      "successful" : 1,
      "failed" : 0
    },
    "_seq_no" : 3710,
    "_primary_term" : 1,
    "status" : 201
  }
}
```

```
{
  "index" : {
    "_index" : "shakespeare",
    "_type" : "_doc",
    "_id" : "111395",
    "_version" : 1,
    "result" : "created",
    "_shards" : {
      "total" : 2,
      "successful" : 1,
      "failed" : 0
    },
    "_seq_no" : 111395,
    "_primary_term" : 1,
    "status" : 201
  }
}
```

# Testando Elasticsearch

Realizar busca:

```
curl -H 'Content-Type: application/json' -XGET '127.0.0.1:9200/Shakespeare/_search?pretty' -d '{
```

```
{
  "query" : {
    "match_phrase" : {
      "text_entry" : "to be or not to be"
    }
  }
}
```



# Testando Elasticsearch

```
root@elkubuntu:/# curl -H 'Content-Type: application/json' -XGET '127.0.0.1:9200/shakespeare/_search?pretty' -d'
{
  "query":{
    "match_phrase":{
      "text_entry": "to be or not to be"
    }
  }
}
```



```
    "took" : 1,
    "timed_out" : false,
    "_shards" : {
      "total" : 1,
      "successful" : 1,
      "skipped" : 0,
      "failed" : 0
    },
    "hits" : {
      "total" : {
        "value" : 1,
        "relation" : "eq"
      },
      "max_score" : 13.889601,
      "hits" : [
        {
          "_index" : "shakespeare",
          "_type" : "_doc",
          "_id" : "34229",
          "_score" : 13.889601,
          "_source" : {
            "type" : "line",
            "line_id" : 34230,
            "play_name" : "Hamlet",
            "speech_number" : 19,
            "line_number" : "3.1.64",
            "speaker" : "HAMLET",
            "text_entry" : "To be, or not to be: that is the question:"
          }
        }
      ]
    }
  ]
}
root@elkubuntu:/# _
```



# Como a busca é feita?

## Documento 1

Espaço: a fronteira final

## Documento 2

Um design de interiores busca recriar espaço no ambiente

## Índice invertido

Espaço: 1,2

a: 1

fronteira: 1

final: 1

design: 2

ambiente: 2



# Como a busca é feita?

**TF-IDF (relevância) = Term Frequency \* Inverse Document Frequency**

- Term Frequency: frequência com que uma palavra aparece em um documento
- Document Frequency: frequência com que um termo aparece em todos os documentos



# ELK Stack Kibana







# Definições

Kibana é a interface de usuário do Elastic Stack. Permite que:

- Buscar, Observar e proteger os dados
- Analisar os dados, independente do tipo
- Gerenciar, monitorar e proteger o Elastic Stack





# Welcome home



## Enterprise Search

Create search experiences with a refined set of APIs and tools.



## Observability

Consolidate your logs, metrics, application traces, and system availability with purpose-built UIs.



## Security

Prevent, collect, detect, and respond to threats for unified protection across your infrastructure.



## Analytics

Explore, visualize, and analyze your data using a powerful suite of analytical tools and applications.

## Get started by adding integrations

To start working with your data, use one of our many ingest options. Collect data from an app or service, or upload a file. If you're not ready to use your own data, add a sample data set.

[+ Add integrations](#)

[📄 Try sample data](#)

[📁 Upload a file](#)



# Links

- <https://www.elastic.co/pt/logstash/>
- <https://www.elastic.co/guide/en/logstash/current/plugins-filters-grok.html>
- <https://www.elastic.co/guide/en/logstash/current/plugins-inputs-http.html>
- <https://www.elastic.co/guide/en/beats/filebeat/8.4/filebeat-installation-configuration.html>
- <https://www.elastic.co/guide/en/beats/libbeat/current/beats-reference.html>
- <https://www.elastic.co/guide/en/logstash/current/filter-plugins.html>
- <https://github.com/hpcugent/logstash-patterns/blob/master/files/grok-patterns>
- <https://www.elastic.co/pt/what-is/elasticsearch>
- <https://www.elastic.co/pt/kibana/>

# ELK Stack

