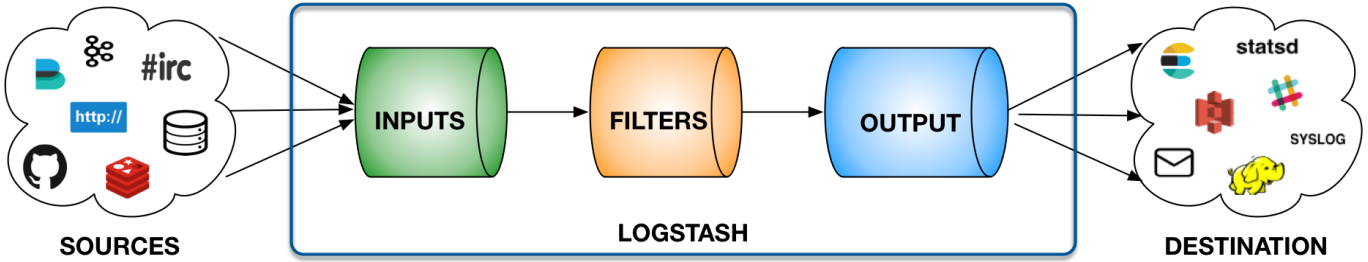
PROCESAMIENTO EN LOGSTASH

1.

INPUT, FILTER & OUTPUT

FLUJO DE INFORMACIÓN

- La **entrada** por fichero en JSON
- La **salida** en *stdout()* modo debug
- La **configuración** a través de un fichero



ENTRADA DE INFORMACIÓN DESDE FICHERO

Lee desde el comienzo el fichero test1.json en formato Json:

```
input {
    file {
        path => "/home/openweb/Documents/dataset/
        test.json"
         start_position => "beginning"
         codec => "json"
}}
```

SALIDA DE INFORMACIÓN POR STDOUT()

Se utiliza la salida *stdout()* para poder imprimir por pantalla la información tras el procesado.

El codec *ruby* debug imprimirá usando la librería *“awesome_print”*.

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

FLUJO DE INFORMACIÓN

Se ejecuta Logstash de forma manual:

```
/usr/share/logstash/bin/logstash -f /etc/logstash/
conf.d/logstash.conf —path.settings=/etc/logstash
```

Se lanza un mensaje de entrada:

```
echo '{ "name":"John", "surname":"Pitt",
"age":"30", "cars":[ "Ford", "BMW", "Fiat" ] }' >> "/
home/opweb/Documents/dataset/test.json"
```

FLUJO DE INFORMACIÓN

Resultado (*sin procesamiento intermedio*):

FILTER MUTATE

Permite realizar modificaciones en los campos como renombrar, borrar, reemplazar, modificaciones…

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

FILTER GROK

Trabaja con texto sin esquema y lo estructura.

```
filter{
   grok {
     match => { "personalInformation" => [ "Information:
%{WORD:Name} %{WORD:Surname} %
{NUMBER:age} %{NUMBER:height}" ] }
   }
}
```

FILTER CIDR & GEOIP

Objetivo excluir las IPs internas de la geolocalización.

```
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"]}
…
```

FILTER CIDR & GEOIP

Se añade la geolocalización de las IPs que pasen el filtro.

```
…
        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"] } } }
}
```

2.

CODECS

CODECS

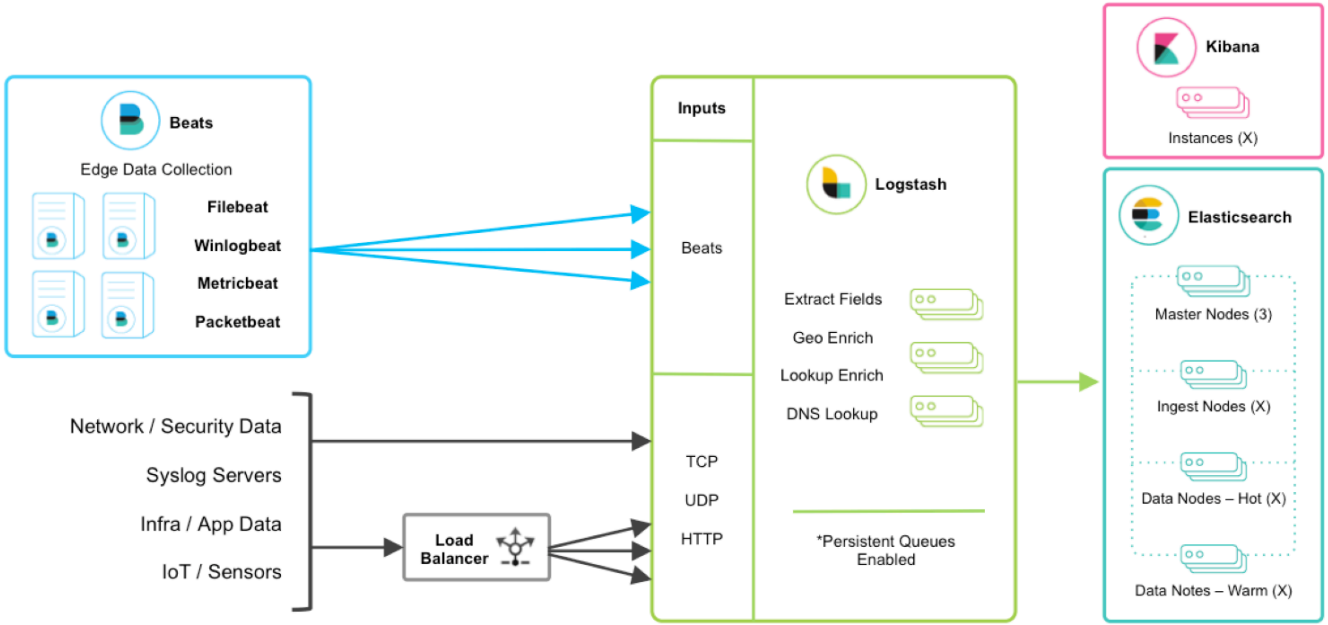
- Son otro filtro que pueden operar como parte del *input* o *output*.
- Transporte <-> Formato
- Populares
  - JSON: codifica y descodifica en formato JSON.
  - Multiline: Une líneas para forma un único evento, por ejemplo una excepción en Java.

Otras posibilidades: *avro*, *cef*, *collectd*, *fluent*, *nmap*, *plain*, *s3_plain*…

3.

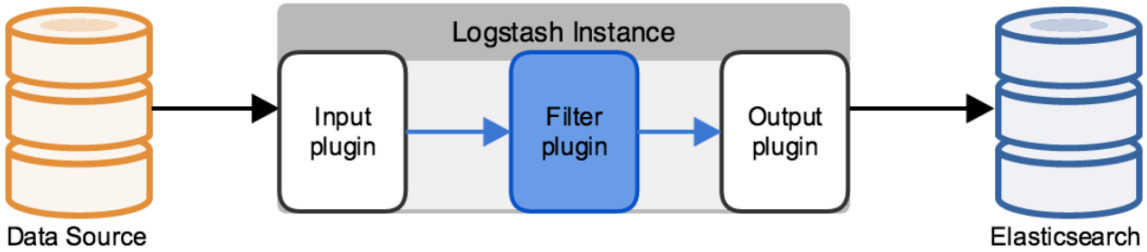
ESCALADO

ESCALADO


Beats
Edge Data Collection
Filebeat
Winlogbeat
Metricbeat
Packetbeat
Network / Security Data
Syslog Servers
Infra / App Data
IoT / Sensors
Load Balancer
Inputs
Beats
TCP
UDP
HTTP
Logstash
Extract Fields
Geo Enrich
Lookup Enrich
DNS Lookup
*Persistent Queues Enabled
Kibana
Instances (X)
Elasticsearch
Master Nodes (3)
Ingest Nodes (X)
Data Nodes – Hot (X)
Data Notes – Warm (X)

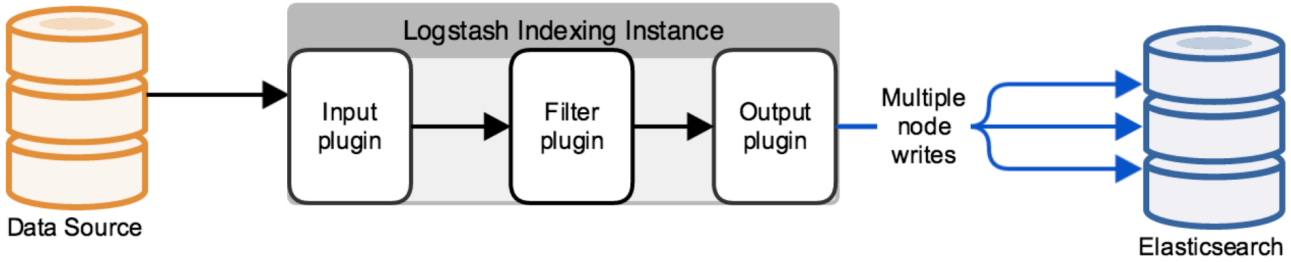
ESCALADO

- Procesamiento simple a un nodo ES.



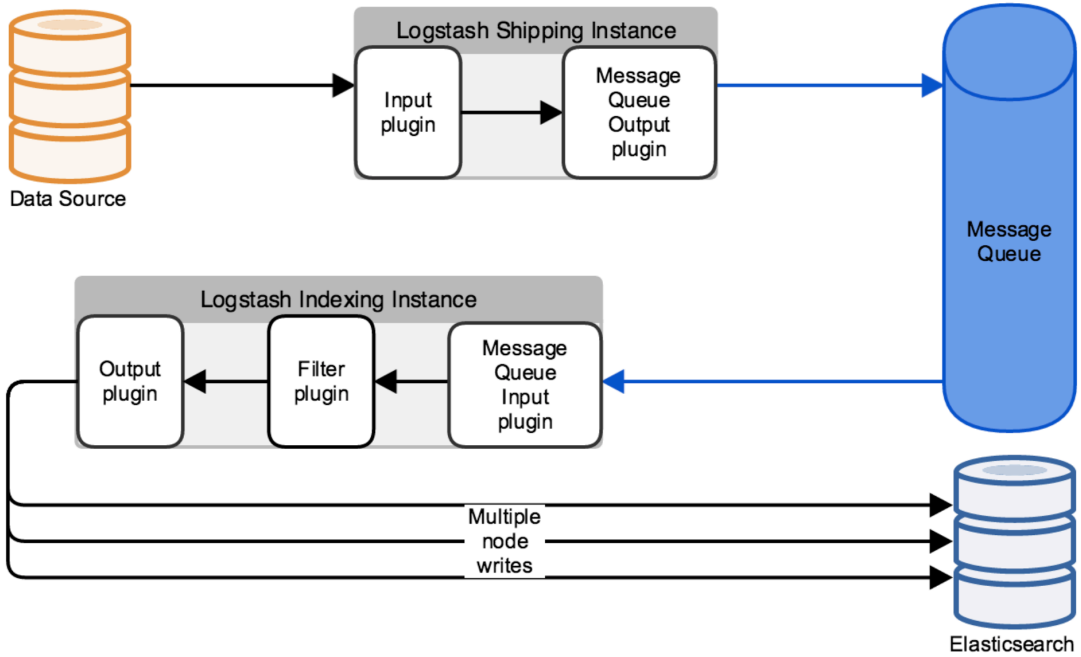
ESCALADO

- Procesamiento simple a un clúster ES



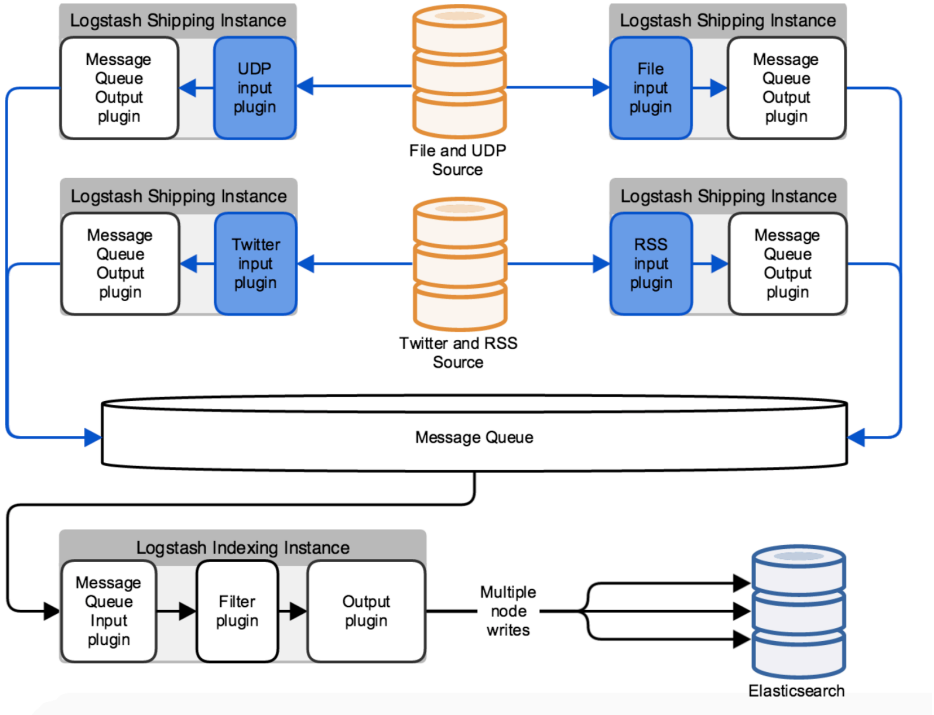
ESCALADO

- Uso de Buffer



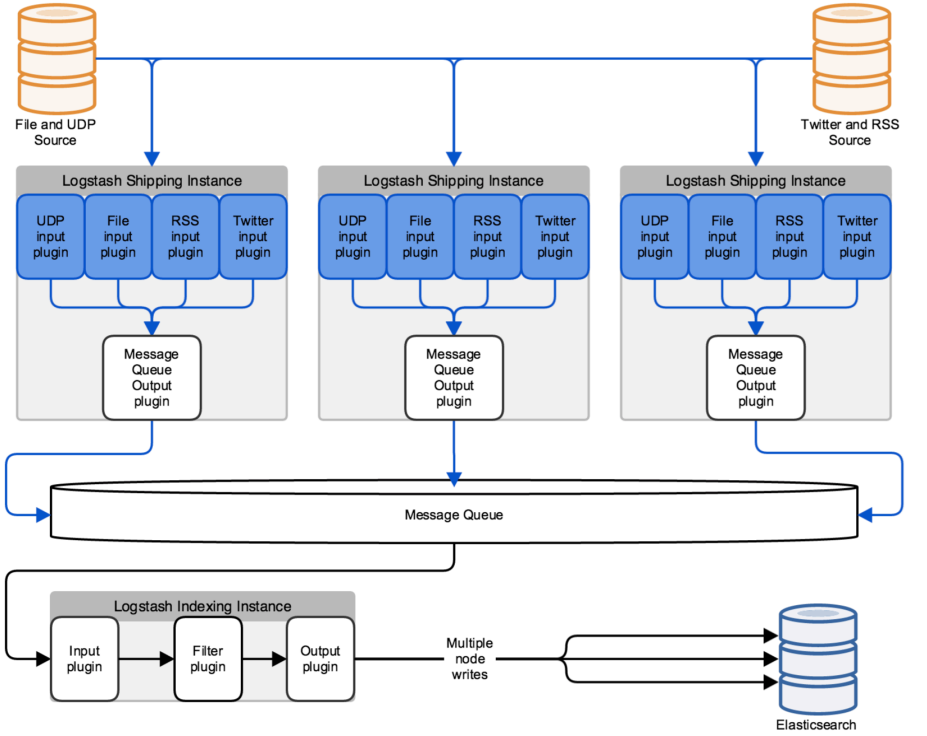
ESCALADO

- Múltiples conexiones



ESCALADO

- Alta disponibilidad



ESCALADO

- Multihilo