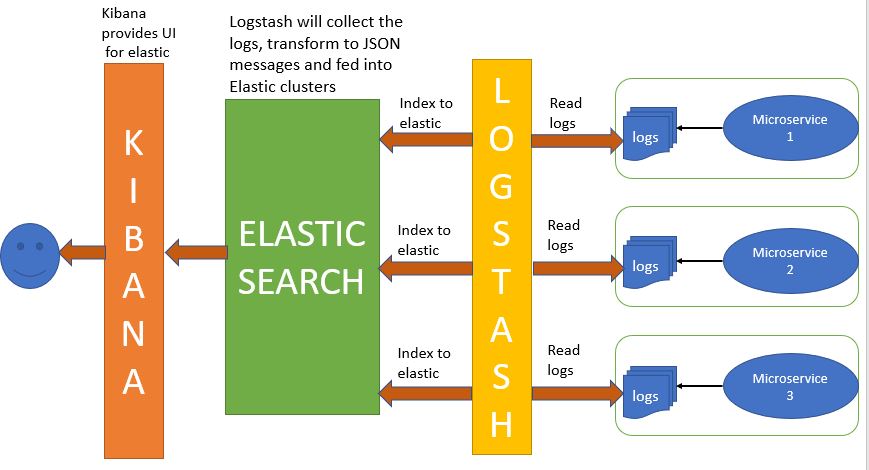
**Spring Boot Distributed Logging with ELK**

**Download the ELK Stack Components.**

https://artifacts.elastic.co/downloads/elasticsearch/elasticsearch-7.7.1-windows-x86\_64.zip

<https://artifacts.elastic.co/downloads/logstash/logstash-7.7.1.zip>

<https://artifacts.elastic.co/downloads/kibana/kibana-7.7.1-windows-x86_64.zip>

****

1.ElasticSearch dir : Run the **elasticsearch.bat** using the command prompt.

Elasticsearch server is accessed at localhost:9200

2.Kibana Dir : Modify the /config/kibana.yml in Kibana dir to point to the elasticsearch instance. This will be 9200.

So uncomment the following line in kibana.yml-

elasticsearch.hosts: ["http://localhost:9200"]

Run the Kibana/bin/kibana.bat using the command prompt. kibana UI is accessed at localhost:5601

**For LogStash configuration**

Create a configuration file named logstash.conf as per the details below and put it in logstash/bin dir.

input {

file {

type => "java"

path => "C:/elk/spring-boot-elk.log"

codec => multiline {

pattern => "^%{YEAR}-%{MONTHNUM}-%{MONTHDAY} %{TIME}.\*"

negate => "true"

what => "previous"

}

}

}

filter {

#If log line contains tab character followed by 'at' then we will tag that entry as stacktrace

if [message] =~ "\tat" {

grok {

match => ["message", "^(\tat)"]

add\_tag => ["stacktrace"]

}

}

}

output {

stdout {

codec => rubydebug

}

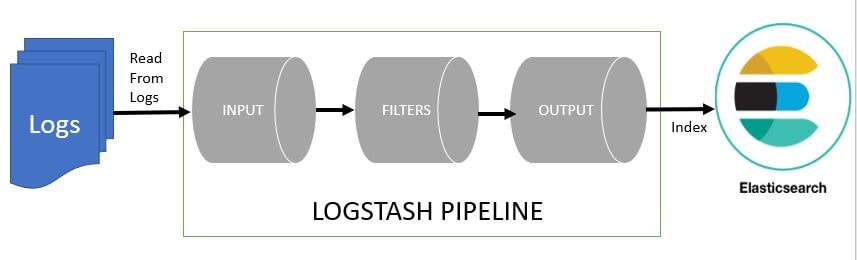
# Sending properly parsed log events to elasticsearch

elasticsearch {

hosts => ["localhost:9200"]

}

}



Start logstash using the command prompt from logsyash/bin dir

logstash -f logstash.conf

4.Start the spring boot application by running the spring-boot-elk as a SprinGBootApp.  
Logs will be generated in C:/elk folder.

Test the spring-boot-elk outpiut with <http://localhost:9090/test> and <http://localhost:9090/exception> urls in the browser multiple times.

In order to visualize and explore data in Kibana, create an index pattern to retrieve data from Elasticsearch.

5. Open the kibana UI console on http://localhost:5601/app/kibana/discover

and create an index pattern logstash-\* to see the indexed data

Define Index Pattern **logstash**-\*

Select the Time Filter filed name To will use this field to filter the data by time. [E.g.@timestamp](mailto:E.g.@timestamp)

Click on create index pattern.

Open Dicovre menu by <http://localhost:5601/app/kibana/discover>

Select the Index pattern you have created on the left side.

In the Filter option on tpop left add @timestamp : 21:14:49.261 or @timestamp : \*

Or message:info or message:error

And click on update button right side.

-- you can create new dashboard to monitor and visualize the the real time data from the logs.

--Visulaize the data locally with visulaizer in Kibana.

<http://localhost:5601/app/kibana#/visualize--> Create New Visulaization from Log file by slecting Visulaization type and log file.

Logstash is reading log files using the logstash filereader on local system.

To read data from multiple server log files and index it to elasticsearch.

-- option is to install logstash on all the servers and then index it to the elasticsearch server.

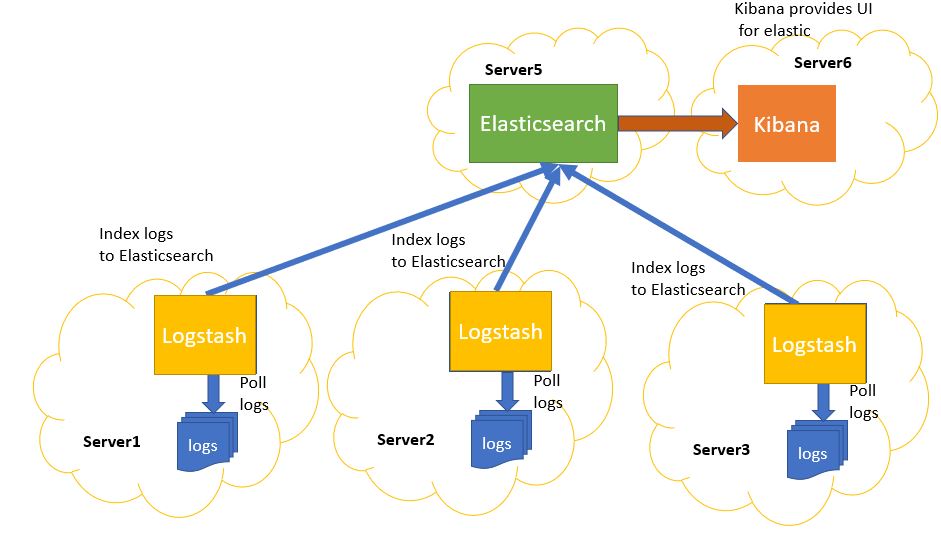
**Logstash consumes a lot of resources so it is not an optimum solution to have logstash installed on all fileservers.**

Instead we can use Beats in such scenarios on the local machines to collect the logs and forward this logs to the **Logstash server.**

Beats are lightweight data shippers that we install as agents on servers to send specific types of operational data to Logstash. We will then filebeat to multiple servers, these will then read the log files and send it to logstash.

To read data from multiple server log files and index it to elasticsearch.

-- option is to install logstash on all the servers and then index it to the elasticsearch server.

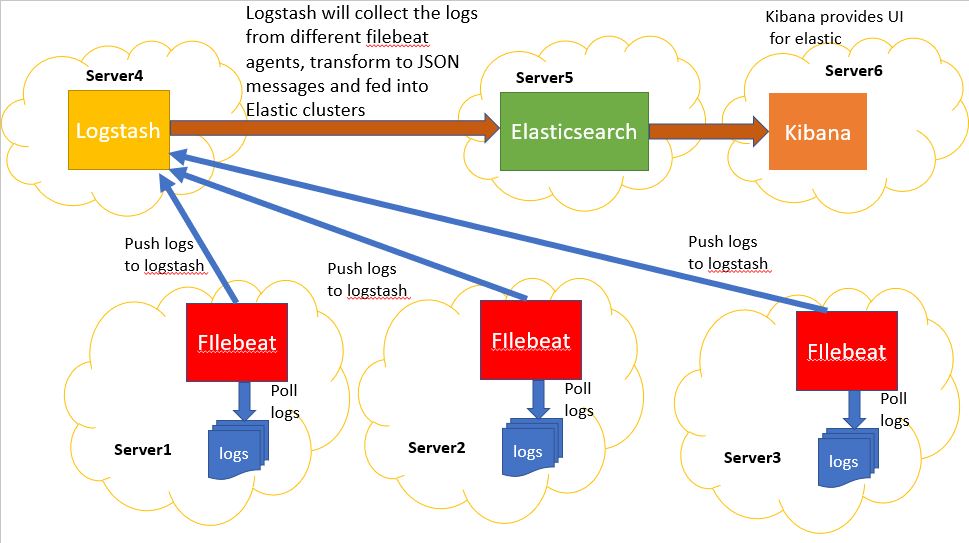


**Logstash consumes a lot of resources so it is not an optimum solution to have logstash installed on all fileservers.**

Instead we can use Beats in such scenarios on the local machines to collect the logs and forward this logs to the **Logstash server.**

Beats are lightweight data shippers that we install as agents on servers to send specific types of operational data to Logstash. We will then filebeat to multiple servers, these will then read the log files and send it to logstash.

-------------



Kibana URLs

http://localhost:5601/app/logs

http://localhost:5601/app/kibana#/management?\_g=()

http://localhost:5601/app/kibana#/management/kibana/index\_patterns?\_g=()

\*\*\*\*\*\*\*\*\*\*\*\*\*