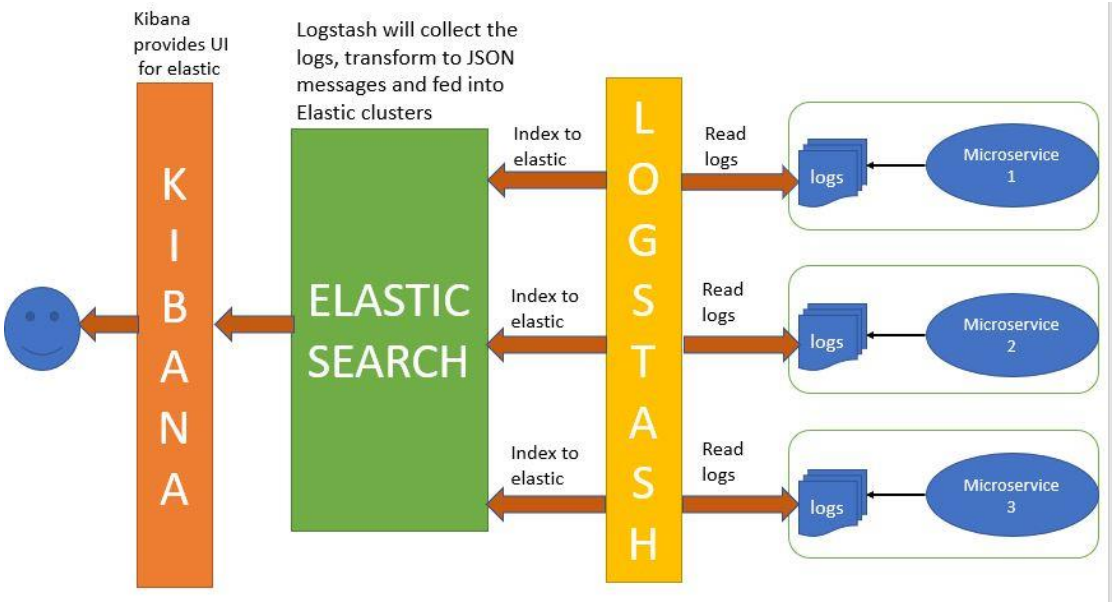
Logging your microservice using ELK

ELK stands for the 3 services we use for this action:

1. **E**lasticSearch
2. **L**ogStash
3. **K**ibana



(See credit below)

The tutorial includes those steps:

1. Installation
2. Project Properties
3. Code implementation
4. Data search

1. Installation

Download ElasticSearch from:
https://www.elastic.co/downloads/kibana
Unzip the file to your file system and
Run: /bin/kibana.bat

Download kibana from:
https://www.elastic.co/downloads/kibana
Unzip the file to your file system and
Run: /bin/kibana.bat

Download LogStash from:

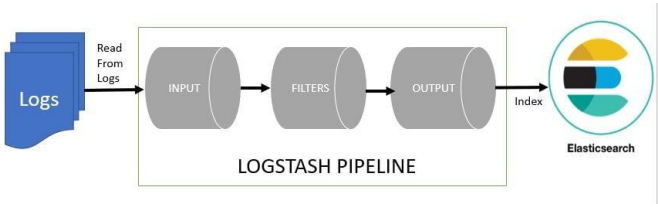
https://www.elastic.co/downloads/logstash

Unzip the file to your file system

Create conf file (This is a configurations file. See below example)

Run: /bin/logstash -f logstash.conf (Where logstash.conf is the config fite you created)

The conf file includes 3 sections: input, filter and output:



(See credit below)

In this guide, we will use the application log file as input, and the ElasticSearch as output.

```
input {
  file {
    type => "java"
    path => "C:/logDemo.log"
    codec => multiline {
      pattern => "^%{YEAR}-%{MONTHNUM}-%{MONTHDAY} %{TIME}.*"
      negate => "true"
      what => "next"
    }
  }
}

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"]
  }
}
```

** **EXAMPLE:** You can use the attached file logDemo.conf

You can see documentation of the conf file in:
https://www.elastic.co/guide/en/logstash/current/configuration.html

2. Project Properties

In the **application.properties** file, add this configuration:

/src/main/resources/application.properties

```
logging.file=C:/logDemo.log
```

*** **IMPORTANT:** make sure to use the same file path in the conf file of the Logstash.

3. Code implementation

In your app, create an instance of Logger (org.apache.logging.log4j.Logger) using this code:

```
Logger LOG = LogManager.getLogger(YOUR_CLASS.class.getName());
```

By using the function ***Log ()*** you add a new line to the log file. This function accept the parameters:
level - the logging level: Level (org.apache.logging.log4j.Level)
message - the message string to log: String

Example:

```
String response = "This is Logging Demo, Hello " + name +", "+ new Date();

logger.log(Level.INFO, "helloWorld(String) : " + response);
```

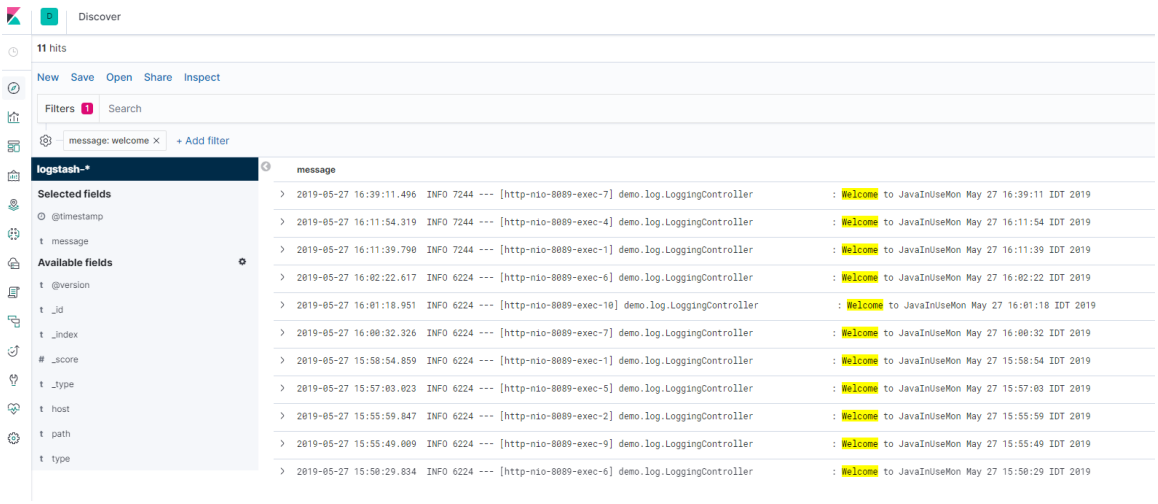
3. Data Search

Go to your browser and enter: http://localhost:5601

This is the **Kibana** dashboard

On first time you must add index for the Logstash data:
GoTo: Management > Index pattern > create an index pattern **logstash-***

Then – GoTo: Discover > see the logged data.



** **Credit:** https://www.javainuse.com/spring/springboot-microservice-elk