

BILKENT UNIVERSITY
ENGINEERING FACULTY
DEPARTMENT OF COMPUTER ENGINEERING

CS 399
**SUMMER TRAINING
REPORT**

Cüneyt EREM

21202398

Performed at

Türkiye İş Bankası

Teknoloji ve Operasyon Merkezi-Tutom, İstanbul
(Technology and Operation Center)

07.08.2017 – 08.09.2017 *(extended)*

Table of Contents

1	Introduction	3
2	Company Information	3
2.1	About the company	3
2.2	About your department.....	3
2.3	About the hardware and software systems.....	3
2.4	About your supervisor	4
3	Work Done	4
4	Performance and Outcomes.....	23
4.1	Applying Knowledge and Skills Learned at Bilkent	233
4.2	Solving Engineering Problems	233
4.3	Team Work	233
4.4	Multi-Disciplinary Work.....	233
4.5	Professional and Ethical Issues.....	233
4.6	Impact of Engineering Solutions.....	244
4.7	Locating Sources and Self-Learning.....	244
4.8	Knowledge about Contemporary Issues.....	244
4.9	Using New Tools and Technologies	244
5	Conclusions.....	244
	References.....	255
	Appendices	266

1 Introduction

In this report, Internship in İşbank will be discussed in detail. In the first part, information about the company, department, hardware-software systems, and supervisor will be talked. Work done part will show what kind of work is done and how is the process of the work. After that, performance and outcomes will be defined with the conclusion.

I worked at IT department of Türkiye İş Bankası in Tuzla/İstanbul. İşbank is one of the largest banks in Turkey and it has a big technology department in Tuzla center Tutom including Softech, Data management, and Information Technology Dept. İşbank is institutional company and it has definite rules and awesome system to proceed itself efficiently, therefore, working there was a big opportunity for me because I wanted to work such a big company. Also, because Bilkent is not in Istanbul, being there could give me an idea about this city for the future plan. I was in application servers' team and I was responsible for using Elastic Stack technology for authentication for users and detecting an anomaly in server log data. My part was important in terms of the project because the first part was one of the tasks that department must do. The second part was not the target but they wanted to decide whether anomaly detection will work for them. Therefore, security and anomaly detection was important tasks for the team.

2 Company Information

2.1 About the company

The first truly national bank in the Turkish Republic was founded on 26 August 1924 by Kemal Atatürk at the First Economy Congress in İzmir. İşbank began operations with two branches and 37 staff under the leadership of Celal Bayar, its first general manager. The bank was established with donations of cash and gold bullion from Muslim supporters in the future Pakistan sent via the Imperial Bank of India to support the Turkish War of Independence [1]. Nowadays, it is Turkey's one of largest bank. İşbank was ranked 96th in a survey of "The World's Biggest 1000 Banks". It was ranked and 447 on the Forbes Global 2000 list for 2015. It had a gross profit for 2015 of TRY 3.023 billion and USD 6.8 billion in Tier I Capital, as defined by Basel's Bank for International Settlements [1].

2.2 About your department

BT – Bilgi Teknolojileri Bölümü (IT Department)
Uygulama Sunucuları Yönetimi Grubu (IIS/.NET) (Application servers' management group). Windows Was IIS: Windows Process Activation Service generalizes the Internet Information Services (IIS) process model, removing the dependency on HTTP. All the features of IIS that were previously available only to HTTP applications are now available to applications hosting Windows Communication Foundation (WCF) services, by using non-HTTP protocols [2].

2.3 About the hardware and software systems

Systems that are used: Windows 2008 R2 64 bit / WAS 8.5.5, Virtual server applications on VMWare technology. Data Power of IBM has been also using.

2.4 About your supervisor

Head of the Application Servers Team:

Özgür Aydın, System Director, İTÜ Electronics Engineering Department, graduated at 2000. Master thesis at Marmara Üniversitesi Enformatik at 2003

Computer Engineer **Supervisor:**

Barışcan Elkama, System Specialist, Izmir Yasar University Computer Engineering Department, graduated at 2013. E-mail: bariscan.elkama@isbank.com.tr

3 Work Done

According to elasticsearch official website, elasticsearch is a search engine based on Lucene. It provides a distributed, multitenant-capable full-text search engine with an Http web interface and schema-free JSON documents. It is developed in Java and released as open source under the Apache License. It is most popular enterprise search engine and developed alongside a data-collection and log-parsing engine called Logstash, and analytics and visualization platform called Kibana. Loading data from database, servers etc, and beats are used like filebeat. To use more advanced methods of monitoring the data, machine learning to detect an anomaly in log data etc. To analyze the server log data in a more efficient way, the elastic stack is used [3]. Authentication for different users to access only specific field is required. For example, one team members are responsible just for accessing client name and surname, other team members are responsible only for credit card numbers etc. Therefore, authentication is provided through log data. Another objective which is done is to analyze the data by using machine learning with watches property. Machine learning can be used to detect anomalies in log data. Normally, anomalies can be found by just looking at the screen but there are thousands of lines of logs, to find every anomaly in this data, machine learning detection is useful because by finding the anomaly, server faults can be found and fixed fast. When an anomaly is detected, watches can urge the user via e-mail immediately. Now, let's see how elastic stack is used. Elastic stack (ELK) can be installed on Windows or Linux. Sometimes version incompatibility can occur, to avoid this issue, the last version should be installed like elasticsearch 5.5.1, logstash 5.5.1, filebeat 5.5.1 and kibana 5.5.1.

To start with elastic, Java JDK-8 is installed. In terminal is opened on desktop and the following command is written one by one;

```
sudo add-apt-repository ppa:webupd8team/java
sudo apt-get update
sudo apt-get install oracle-java8-installer
sudo apt-get install oracle-java8-set-default
```

Elastic can be used in two ways. First one is running as service and the second one is installing from the elastic website. After installing zip folders for elasticsearch, logstash, filebeat, and kibana, they are extracted on the desktop (or whatever you want). For each tool, following commands are used to start elastic stack components;

```
kibana;
name1@name1-VirtualBox:~/Desktop/kibana-5.5.1-linux-x86_64$ bin/kibana
elasticsearch;
name1@name1-VirtualBox:~/Desktop/elasticsearch-5.5.1$ bin/elasticsearch
logstash;
name1@name1-VirtualBox:~/Desktop/logstash-5.5.1/bin$ ./logstash -f first-
pipeline.conf
```

(first-pipeline.conf includes properties how to send data to elasticsearch, it is explained in next pages.)

Running logstash as a service has different start/stop commands from normal way.

As a service to start/stop;

```
name1@name1-VirtualBox:~/Desktop/logstash-5.5.1/bin$ sudo systemctl stop
logstash.service
name1@name1-VirtualBox:~/Desktop/logstash-5.5.1/bin$ sudo systemctl start
logstash.service
```

Running as service also loads the files into specific folders like etc, home, so when running on service, the process should be compatible with service rules otherwise errors will occur [4]. In this project, normal installation is used.

Filebeat needs to be run as root to have access filebeat rights.

```
sudo su
```

<entering password of the linux>

After doing all filebeat files into root, some files can be converted into user again but to start the filebeat, files should be owned onto root again;

```
root@name1-VirtualBox:/home/name1/Desktop/filebeat-5.5.1-linux-x86_64# chown
name1:root /home/name1/Desktop/filebeat-5.5.1-linux-x86_64/filebeat.yml
```

```
root@name1-VirtualBox:/home/name1/Desktop/filebeat-5.5.1-linux-x86_64# chown
root:root /home/name1/Desktop/filebeat-5.5.1-linux-x86_64/filebeat.yml
```

To start the filebeat correctly, every time data memory should be cleaned therefore these two commands need to be entered at each start;

```
root@name1-VirtualBox:/home/name1/Desktop/filebeat-5.5.1-linux-x86_64# rm
data/registry
```

```
root@name1-VirtualBox:/home/name1/Desktop/filebeat-5.5.1-linux-x86_64#. /filebeat
-e -c filebeat.yml -d "publish"
```

So by doing these commands, elasticsearch is started first, then logstash is started and connected to elasticsearch but it waits to be loaded by data, therefore filebeat is connected to logstash and all data is loaded from filebeat to elasticsearch through logstash. There are different types of beats like cloud, database etc. that is why logstash is used as a bridge between elasticsearch and filebeat. Filebeat holds some specific coming from the server including HTTP, string, int etc. values.

Elastic, logstash and filebeat are combined, to see the actions into elasticsearch, kibana is used as an interface. After kibana is started, the program is ready to be used.

In default, kibana can be run but to use the machine learning, graph, monitoring etc. features, x-pack needs to be installed [5]. X-pack is installed into elasticsearch, logstash, and kibana separately by these commands;

```
name1@name1-VirtualBox:~/Desktop/logstash-5.5.1$ bin/logstash-plugin install x-pack
```

```
name1@name1-VirtualBox:~/Desktop/kibana-5.5.1-linux-x86_64$ bin/kibana-plugin install x-pack
```

```
name1@name1-VirtualBox:~/Desktop/elasticsearch-5.5.1$ bin/elasticsearch-plugin install x-pack
```

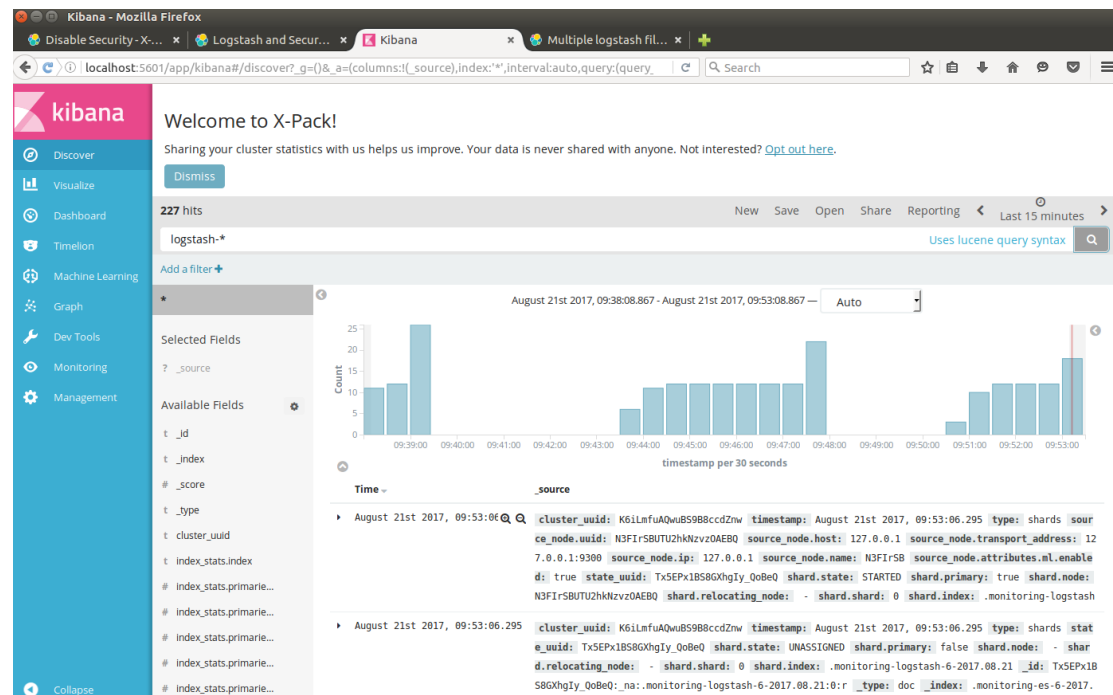


Figure 1

In Figure-2, kibana shows the logstash index on the data. After x-pack is installed, it comes with default username and password like; username: elastic, password:changeme.

They can be removed by adding following lines into yml configuration files. `elasticsearch.yml` and `kibana.yml`;
`xpack.security.enabled: false`

When yml files are configured, elasticsearch, logstash, filebeat etc should be restarted again.

Security;

Authentication is important for security reasons. For this project, it is important because the server sends information but one user should access his own information, another user should access into her own information and they should not see each other's information. Especially banking systems are highly related to this field. The duty is to separate different index fields that should be seen by different users and security will be provided through these thoughts [6]. When user elastic logs in (Figure-3) to the kibana and it adds the index of mobilbank and kredikarti into elasticsearch, the user can access to both of the information successfully (Figure 4, 5).

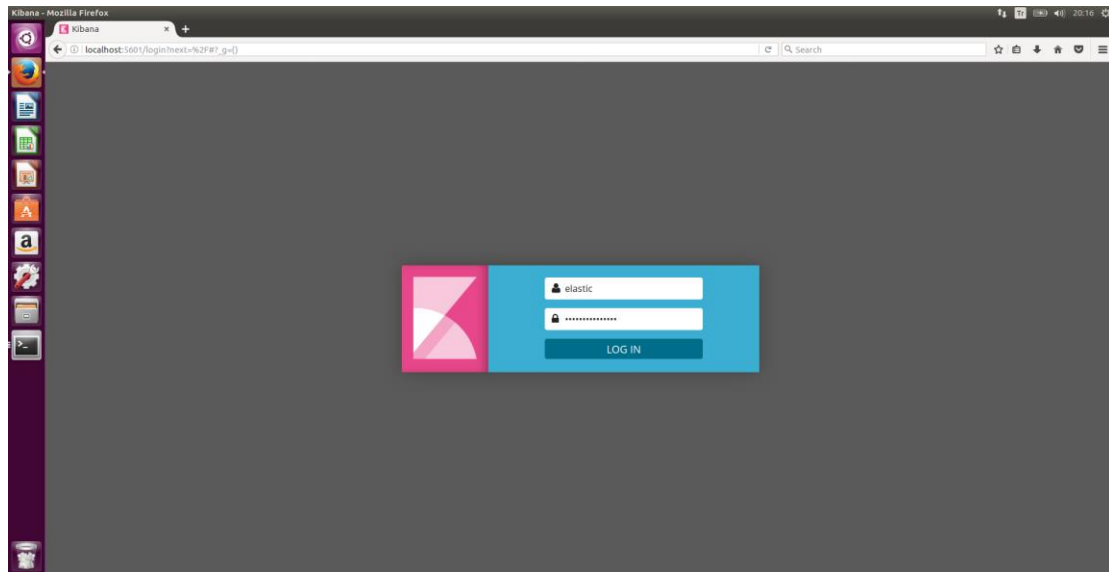


Figure 2

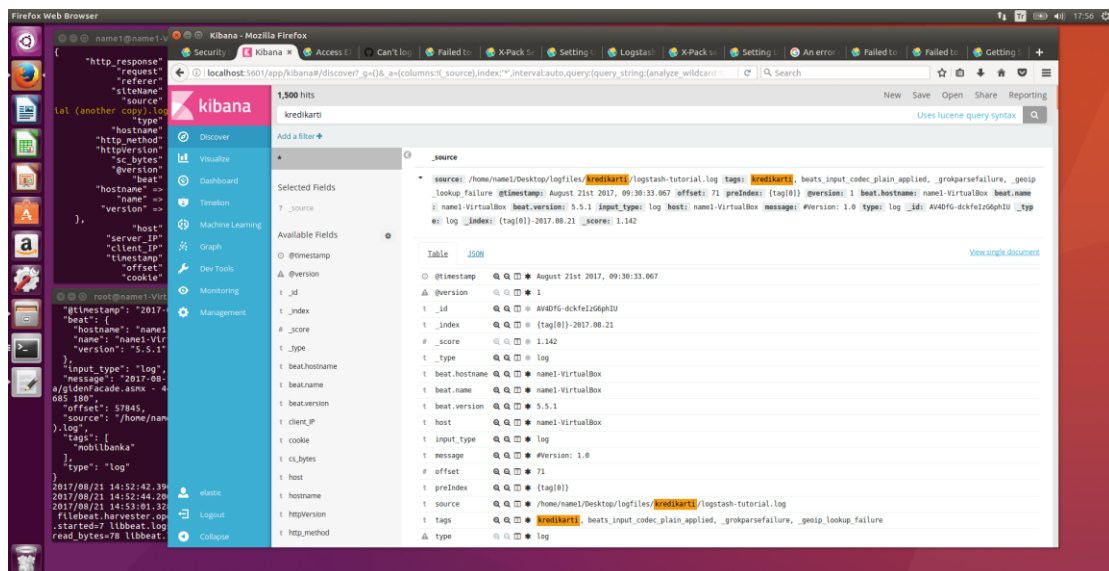


Figure 3

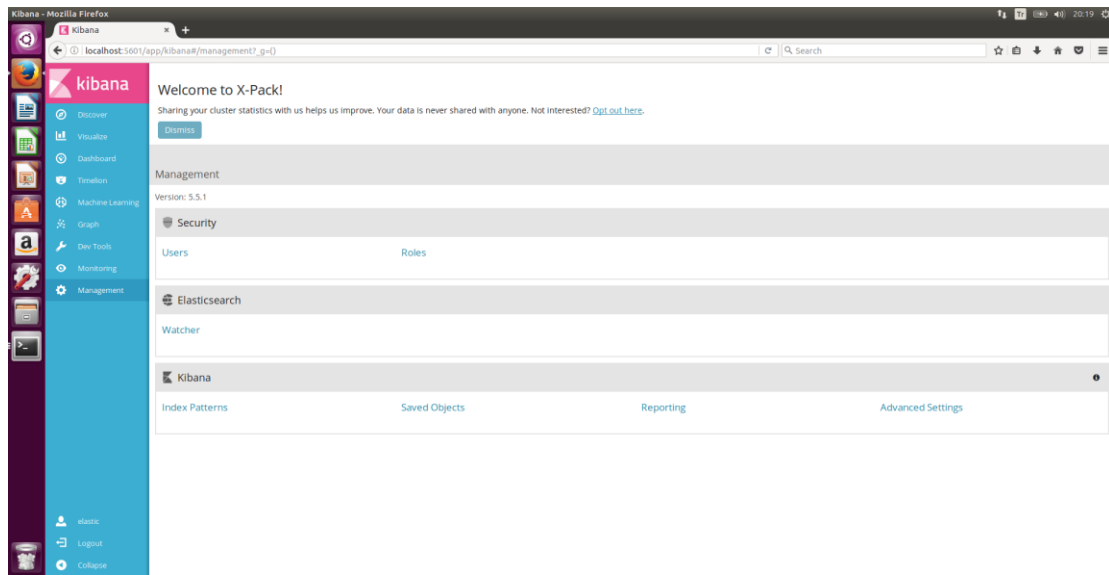


Figure 6

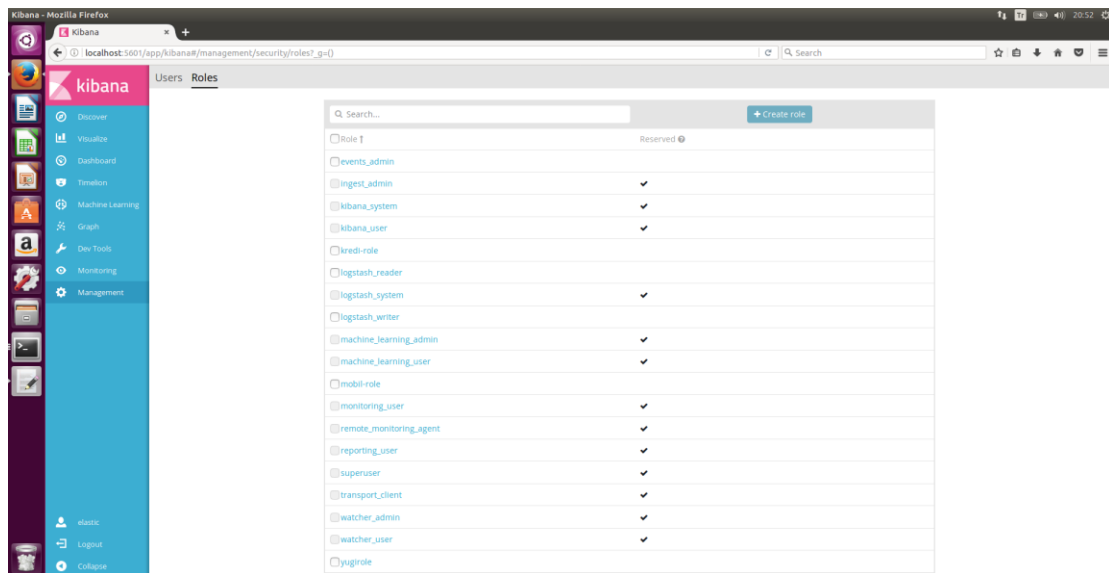


Figure 7

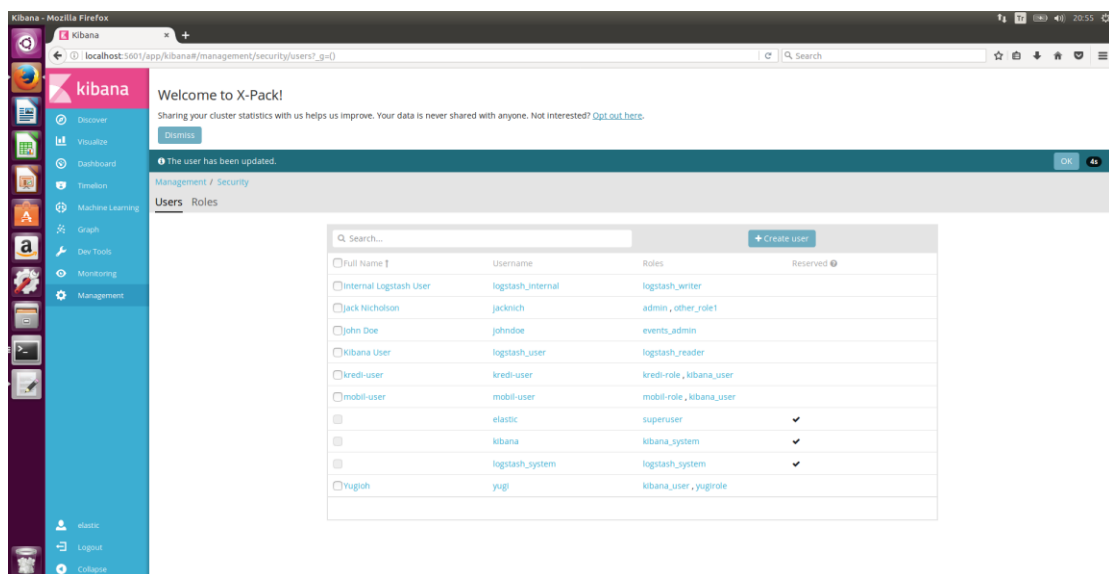


Figure 8

New users and roles are created by creating button and specified features are selected (Figure-8, 9). To modify the specific user authentication, elasticsearch.yml, logstash.yml, kibana.yml, and filebeat.yml should be configured. There are some different types of users to provide authentication. One of the most used ones is native user authentication [7].

elasticsearch.yml (not final status);

```
#native user auth
xpack.security.authc:
  realms:
    native1:
      type: native
      order: 0
```

It provides native user authentication and it can be prioritized in terms of types like native, file etc.

Sometimes, logstash gives memory error after too many logs sent, to fix the error, logstash JVM option should be changed;
logstash JVM.option;
-Xmx2g is converted to -Xmx1g for memory health

logstash.yml;

```
xpack.monitoring.elasticsearch.username: elastic
xpack.monitoring.elasticsearch.password: elasticpassword
```

kibana.yml;

```
elasticsearch.username: elastic
elasticsearch.password: elasticpassword
```

```
#xpack.security.enabled: false
```

```
xpack.security.encryptionKey: "e386d5f380dd962614538ad70d7e9745760f7e8e"
#xpack.reporting.encryptionKey: "e386d5f380dd962614538ad70d7e9745760f7e8e"
```

In elastic website guideline, it says username and password should be logstash name and password but it is wrong and it gives an authentication error because of unknown reason. To handle the problem, they are converted to elastic u and p as viewed above.

filebeat.yml;

filebeat.prospectors:

```
# Each - is a prospector. Most options can be set at the prospector level, so
# you can use different prospectors for various configurations.
# Below are the prospector specific configurations.
```

```
- input_type: log
```

```
# Paths that should be crawled and fetched. Glob based paths.
paths:
```

```

- /home/name1/Desktop/logfiles/kredikarti/*
fields:
  tags: ["kredikarti"]
  fields_under_root: true
#- /home/name1/Desktop/logstash-tutorial.log
#- c:\programdata\elasticsearch\logs\*

- input_type: log

paths:
  - /home/name1/Desktop/logfiles/mobilbanka/*
fields:
  tags: ["mobilbanka"]
  fields_under_root: true

#----- Logstash output -----

hosts: ["localhost:5044"]

filebeat receives the log files from the desktop as it seen. It sends the data to
logstash localhost:5044. Data for both mobilbanka and kredikarti are sent separately
to the elasticsearch.

In logstash bin folder, first-pipeline.conf is created;

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

filter{

  grok
  {
    match => ["message", "%{TIMESTAMP_ISO8601:timestamp} %{WORD:site_ID}
    %{WORD:hostname} %{IP:server_IP} %{WORD:http_method}
    %{URIPATHPARAM:request} (%{NOTSPACE:uri_query}) %{INT:port:integer}
    (?:%{WORD:userName}|-) %{IP:client_IP} HTTP/%{NOTSPACE:httpVersion}
    (%{NOTSPACE:cookie}) (?:%{NOTSPACE:referrer}|-) %{NOTSPACE:siteName}
    %{INT:http_response:integer} %{INT:sub_response:integer}
    %{INT:sc_win32_status:integer} %{INT:sc_bytes:integer} %{INT:cs_bytes:integer}
    %{INT:time_taken:integer}"]
  }

  mutate
  {
    add_field => { "preIndex" => "%{[tags][0]}" } lowercase => "%{preIndex}" }
  }

  output
  {

    elasticsearch

```

```
{
  hosts => ["localhost:9200"]
  document_type => "%{[@metadata][type]}"
  index => "%{preIndex}-%{+YYYY.MM.dd}"
  template_overwrite => true
  template => "/home/name1/Desktop/usy_indexTemp.json"
  template_name => "usylog_template"
  user => "elastic"
  password => "elasticpassword"
}

stdout { codec => rubydebug }
}
```

There are three parts for the logstash conf file. The input receives the beats port which is 5044 host connects to filebeat. In the filter, server log files are received which matches with a timestamp, IP, hostname, HTTP, integer, string etc. After received the logline, a new field is added by mutating which contains one pre-index with the tag[0] that is mobilbanka or kredikarti for different matches. In the output, it is connected to localhost:9200 where the elasticsearch default place is. Username and password are also inserted and other required features inserted for the template.

After these changes, filebeat receives the two different logs from desktop and sends them to logstash. Logstash sends proper data to specific elasticsearch user field with logstash conf file.

When specific user enters his own account, user only sees the specific index field that is wanted, user Kredi-user sees kredikarti indexes (Figure-10) but not mobilbanka indexes (Figure-11) (where elastic superuser can see both, Kredi-user and mobile-user has Kredi-role and mobile-role respectively and both of them have also Kibana-user feature to see the management, discover etc. features, otherwise they cannot see anything on kibana).

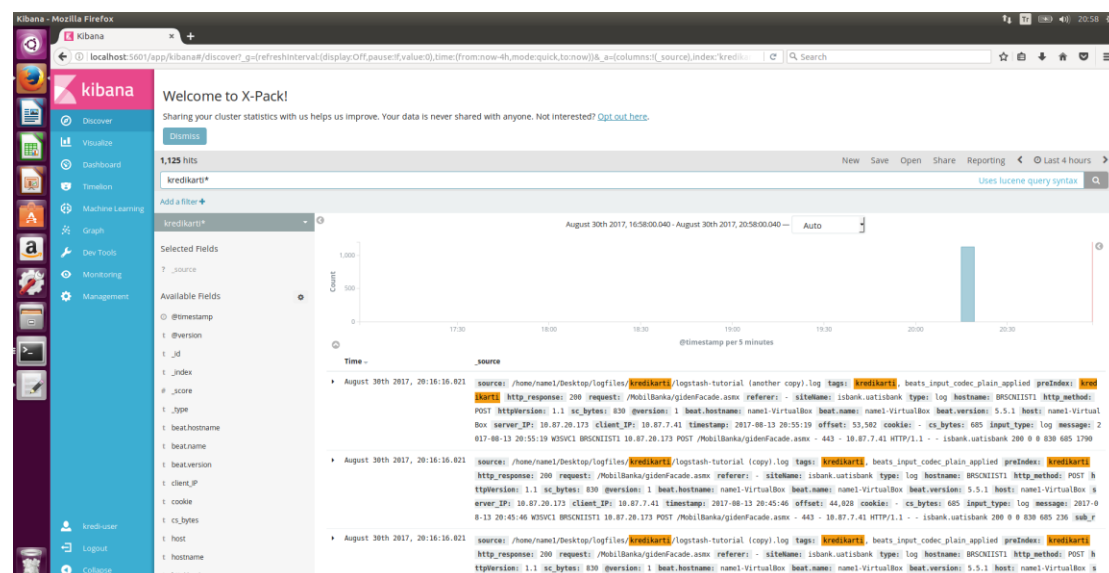


Figure 9

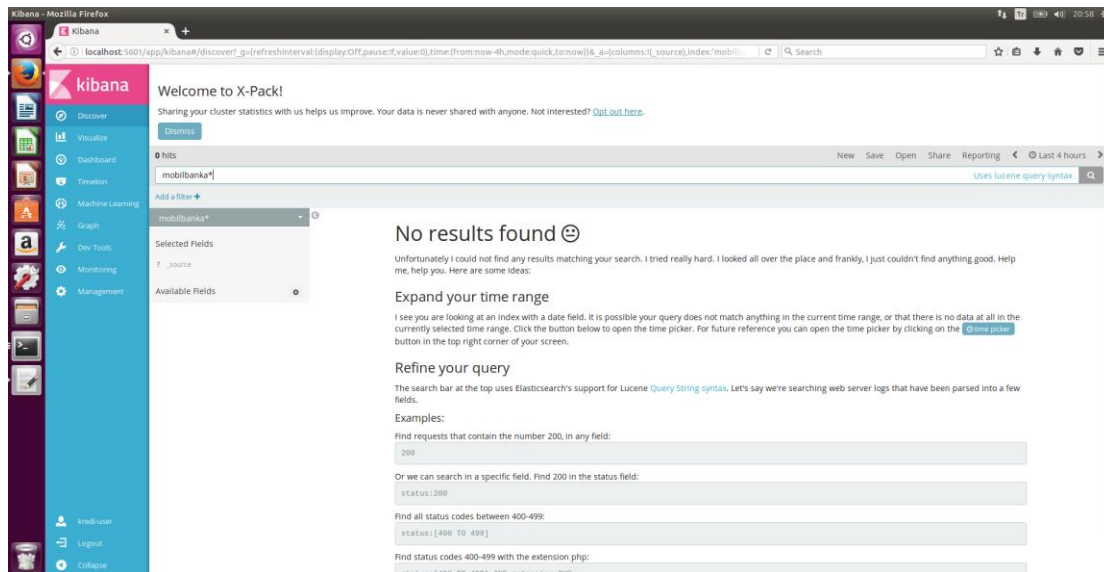


Figure 10

In log files and kredikarti files, there are logstash-tutorial.log files contain server data.

Machine Learning;

Another X-pack property is Machine Learning. To detect the anomaly in the server metric and log data, machine learning analysis gives extremely good results especially for the finding server crashes and problems [8]. When problems detected, alerting via watches are used to urge the user via email [9].

To activate machine learning properties, elasticsearch and kibana need to be changed by adding ml features. Also, to add alerting email into elasticsearch, specified Gmail part should be added. This configuration provides sending email in the specified time interval.

elasticsearch.yml (final situation);

```
#if security wants to be closed
#xpack.security.enabled: false
```

```
#machine learning activates
xpack.ml.enabled: true
node.ml: true
```

```
#native user auth
xpack.security.authc:
  realms:
    native1:
      type: native
      order: 0
```

```
#email alert watch is open
xpack.notification.email.account:
  gmail_account:
```

```
profile: gmail
smtp:
  auth: true
  starttls.enable: true
  host: smtp.gmail.com
  port: 587
  user: example@mail.com
  password: pass
```

Other yml files are the same.

To add metric files into elasticsearch, the terminal is opened into elasticsearch-5.5.1. There are four json server metric files and all of them contain similar server data which can be seen in Appendices-3.

After json folders moved into elasticsearch-5.1.1, following commands are written by the curl in the terminal to prepare elasticsearch space settings and mapping to load the data. User and password authentication should be added if security is open;

```
curl -u elastic:elasticpassword -X PUT -H 'Content-Type: application/json'
http://localhost:9200/server-metrics -d '{
  "settings":{
    "number_of_shards":1,
    "number_of_replicas":0
  },
  "mappings":{
    "metric":{
      "properties":{
        "@timestamp":{
          "type":"date"
        },
        "accept":{
          "type":"long"
        },
        "deny":{
          "type":"long"
        },
        "host":{
          "type":"keyword"
        },
        "response":{
          "type":"float"
        },
        "service":{
          "type":"keyword"
        },
        "total":{
          "type":"long"
        }
      }
    }
  }
}
```

After mapping added successfully, all json data can be loaded safely into elasticsearch Node map;

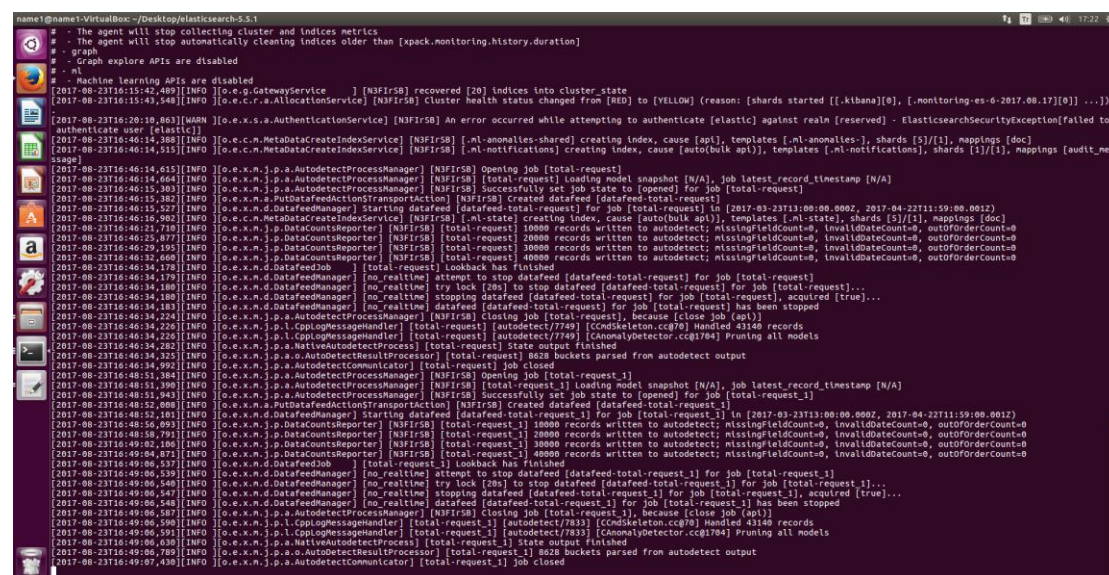
```
curl -u elastic:elasticpassword -X POST -H "Content-Type: application/json"
http://localhost:9200/server-metrics/_bulk --data-binary "@server-metrics_1.json"
```

```
curl -u elastic:elasticpassword -X POST -H "Content-Type: application/json"
http://localhost:9200/server-metrics/_bulk --data-binary "@server-metrics_2.json"
```

```
curl -u elastic:elasticpassword -X POST -H "Content-Type: application/json"
http://localhost:9200/server-metrics/_bulk --data-binary "@server-metrics_3.json"
```

```
curl -u elastic:elasticpassword -X POST -H "Content-Type: application/json"
http://localhost:9200/server-metrics/_bulk --data-binary "@server-metrics_4.json"
```

Now, all data are loaded by Linux bulk feature. This is the command by loading data with the bulk from one location to another and load data can be seen in Figure-12.



```
name1@name1-VirtualBox:~/Desktop/elasticsearch-5.5.1
# - The agent will stop collecting cluster and indices metrics
# - The agent will stop automatically cleaning indices older than [xpack.monitoring.history.duration]
# - graph
# - Graph explore APIs are disabled
# - ml
# - Machine Learning APIs are disabled
[2017-08-23T16:15:42.489] INFO [o.e.g.GatewayService] [N3FIR58] recovered [20] indices into cluster state
[2017-08-23T16:15:43.548] INFO [o.e.c.r.a.AllocationService] [N3FIR58] Cluster health status changed from [RED] to [YELLOW] (reason: [shards started [[.kibana][0], [.monitoring-es-6-2017.08.17][0]] ...])
[2017-08-23T16:28:18.863] WARN [o.e.x.s.a.AuthenticationService] [N3FIR58] An error occurred while attempting to authenticate [elastic] against realm [reserved] - ElasticsearchSecurityException[failed to
authenticate user [elastic]]
[2017-08-23T16:46:14.388] INFO [o.e.c.m.MetadataCreateIndexService] [N3FIR58] [.ml-anomalies-shared] creating index, cause [api], templates [.ml-anomalies-], shards [5]/[1], mappings [doc]
[2017-08-23T16:46:14.515] INFO [o.e.c.m.MetadataCreateIndexService] [N3FIR58] [.ml-notifications] creating index, cause [auto(bulk api)], templates [.ml-notifications], shards [1]/[1], mappings [audit_me
ssage]
[2017-08-23T16:46:14.615] INFO [o.e.x.m.p.a.AutodetectProcessManager] [N3FIR58] Opening Job [total-request]
[2017-08-23T16:46:14.644] INFO [o.e.x.m.p.a.AutodetectProcessManager] [N3FIR58] [total-request] Loading model snapshot [N/A], job latest_record_timestamp [N/A]
[2017-08-23T16:46:15.383] INFO [o.e.x.m.p.a.AutodetectProcessManager] [N3FIR58] Successfully set job state to [opened] for job [total-request]
[2017-08-23T16:46:15.382] INFO [o.e.x.m.a.PathDataFeedActionTransportAction] [N3FIR58] Created datafeed [datafeed-total-request]
[2017-08-23T16:46:15.527] INFO [o.e.x.m.a.DatafeedManager] [N3FIR58] Starting datafeed [datafeed-total-request] for job [total-request] in [2017-03-23T13:00:00.000Z, 2017-04-22T11:59:00.001Z]
[2017-08-23T16:46:16.982] INFO [o.e.c.m.MetadataCreateIndexService] [N3FIR58] [.ml-state] creating index, cause [auto(bulk api)], templates [.ml-state], shards [5]/[1], mappings [doc]
[2017-08-23T16:46:25.710] INFO [o.e.x.m.p.a.DataCountsReporter] [N3FIR58] [total-request] 10000 records written to autodetect; missingfieldcount=0, invaliddatecount=0, outofordercount=0
[2017-08-23T16:46:25.877] INFO [o.e.x.m.p.a.DataCountsReporter] [N3FIR58] [total-request] 20000 records written to autodetect; missingfieldcount=0, invaliddatecount=0, outofordercount=0
[2017-08-23T16:46:29.195] INFO [o.e.x.m.p.a.DataCountsReporter] [N3FIR58] [total-request] 30000 records written to autodetect; missingfieldcount=0, invaliddatecount=0, outofordercount=0
[2017-08-23T16:46:32.468] INFO [o.e.x.m.p.a.DataCountsReporter] [N3FIR58] [total-request] 40000 records written to autodetect; missingfieldcount=0, invaliddatecount=0, outofordercount=0
[2017-08-23T16:46:34.178] INFO [o.e.x.m.a.DatafeedJob] [total-request] Lookback has finished
[2017-08-23T16:46:34.178] INFO [o.e.x.m.a.DatafeedManager] [no_realtime] attempt to stop datafeed [datafeed-total-request] for job [total-request]
[2017-08-23T16:46:34.180] INFO [o.e.x.m.a.DatafeedManager] [no_realtime] try lock [20s] to stop datafeed [datafeed-total-request] for job [total-request]...
[2017-08-23T16:46:34.180] INFO [o.e.x.m.a.DatafeedManager] [no_realtime] stopping datafeed [datafeed-total-request] for job [total-request], acquired [true]...
[2017-08-23T16:46:34.183] INFO [o.e.x.m.a.DatafeedManager] [no_realtime] datafeed [datafeed-total-request] for job [total-request] has been stopped
[2017-08-23T16:46:34.224] INFO [o.e.x.m.p.a.AutodetectProcessManager] [N3FIR58] Closing job [total-request], because [close job (api)]
[2017-08-23T16:46:34.226] INFO [o.e.x.m.p.l.CpplogMessageHandler] [total-request] [autodetect/7749] [CCondSkeleton.cc@78] Handled 43140 records
[2017-08-23T16:46:34.226] INFO [o.e.x.m.p.l.CpplogMessageHandler] [total-request] [autodetect/7749] [CCondSkeleton.cc@78] Pruning all models
[2017-08-23T16:46:34.282] INFO [o.e.x.m.p.a.NativeAutodetectProcess] [total-request] State output finished
[2017-08-23T16:46:34.325] INFO [o.e.x.m.p.a.AutodetectResultProcessor] [total-request] 8028 buckets parsed from autodetect output
[2017-08-23T16:46:34.992] INFO [o.e.x.m.p.a.AutodetectCommunicator] [total-request] Job closed
[2017-08-23T16:48:51.384] INFO [o.e.x.m.p.a.AutodetectProcessManager] [N3FIR58] Opening Job [total-request-1]
[2017-08-23T16:48:51.390] INFO [o.e.x.m.p.a.AutodetectProcessManager] [N3FIR58] Successfully set job state to [opened] for job [total-request-1]
[2017-08-23T16:48:51.943] INFO [o.e.x.m.p.a.AutodetectProcessManager] [N3FIR58] Loading model snapshot [N/A], job latest_record_timestamp [N/A]
[2017-08-23T16:48:52.088] INFO [o.e.x.m.a.PathDataFeedActionTransportAction] [N3FIR58] Created datafeed [datafeed-total-request-1]
[2017-08-23T16:48:52.181] INFO [o.e.x.m.a.DatafeedManager] [N3FIR58] Starting datafeed [datafeed-total-request-1] for job [total-request-1] in [2017-03-23T13:00:00.000Z, 2017-04-22T11:59:00.001Z]
[2017-08-23T16:48:56.493] INFO [o.e.x.m.p.a.DataCountsReporter] [N3FIR58] [total-request-1] 10000 records written to autodetect; missingfieldcount=0, invaliddatecount=0, outofordercount=0
[2017-08-23T16:48:58.791] INFO [o.e.x.m.p.a.DataCountsReporter] [N3FIR58] [total-request-1] 20000 records written to autodetect; missingfieldcount=0, invaliddatecount=0, outofordercount=0
[2017-08-23T16:49:02.106] INFO [o.e.x.m.p.a.DataCountsReporter] [N3FIR58] [total-request-1] 30000 records written to autodetect; missingfieldcount=0, invaliddatecount=0, outofordercount=0
[2017-08-23T16:49:04.871] INFO [o.e.x.m.p.a.DataCountsReporter] [N3FIR58] [total-request-1] 40000 records written to autodetect; missingfieldcount=0, invaliddatecount=0, outofordercount=0
[2017-08-23T16:49:06.537] INFO [o.e.x.m.a.DatafeedJob] [total-request-1] Lookback has finished
[2017-08-23T16:49:06.539] INFO [o.e.x.m.a.DatafeedManager] [no_realtime] attempt to stop datafeed [datafeed-total-request-1] for job [total-request-1]
[2017-08-23T16:49:06.540] INFO [o.e.x.m.a.DatafeedManager] [no_realtime] try lock [20s] to stop datafeed [datafeed-total-request-1] for job [total-request-1]...
[2017-08-23T16:49:06.547] INFO [o.e.x.m.a.DatafeedManager] [no_realtime] stopping datafeed [datafeed-total-request-1] for job [total-request-1], acquired [true]...
[2017-08-23T16:49:06.548] INFO [o.e.x.m.a.DatafeedManager] [no_realtime] datafeed [datafeed-total-request-1] for job [total-request-1] has been stopped
[2017-08-23T16:49:06.587] INFO [o.e.x.m.p.a.AutodetectProcessManager] [N3FIR58] Closing job [total-request-1], because [close job (api)]
[2017-08-23T16:49:06.590] INFO [o.e.x.m.p.l.CpplogMessageHandler] [total-request-1] [autodetect/7833] [CCondSkeleton.cc@78] Handled 43140 records
[2017-08-23T16:49:06.630] INFO [o.e.x.m.p.l.CpplogMessageHandler] [total-request-1] [autodetect/7833] [CCondSkeleton.cc@78] Pruning all models
[2017-08-23T16:49:06.789] INFO [o.e.x.m.p.a.NativeAutodetectProcess] [total-request-1] State output finished
[2017-08-23T16:49:06.789] INFO [o.e.x.m.p.a.AutodetectResultProcessor] [total-request-1] 8028 buckets parsed from autodetect output
[2017-08-23T16:49:07.439] INFO [o.e.x.m.p.a.AutodetectCommunicator] [total-request-1] Job closed
```



```

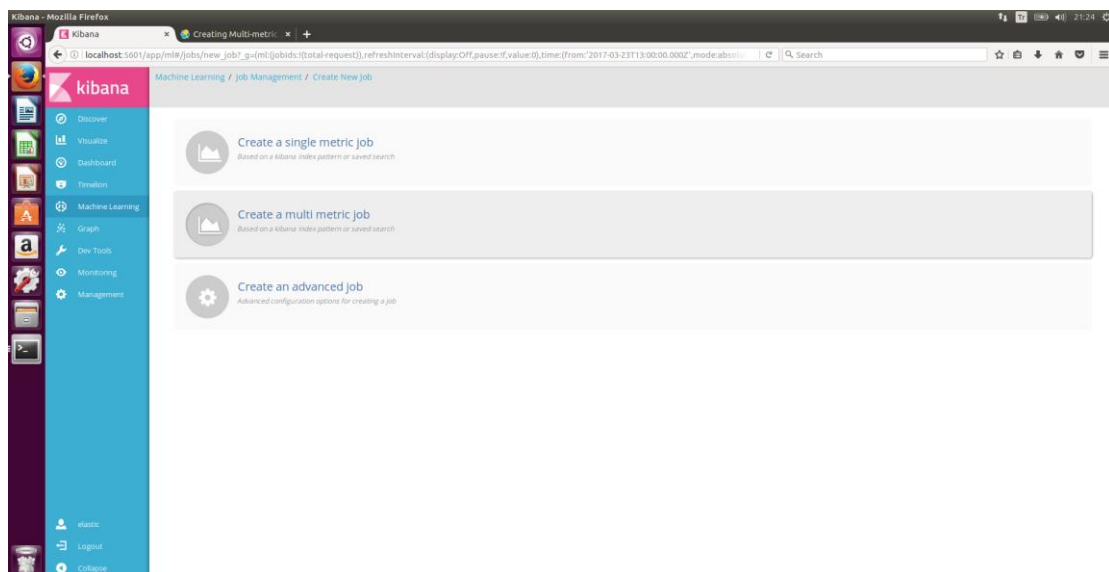
name1@name1-VirtualBox: ~/Desktop/elasticsearch-5.5.1/bin
op, "version":1,"result":"created","shards":{"total":1,"successful":1,"failed":0},"created":true,"status":201},"index":{"_index":"server-metrics","type":"metric","id":"90710","version":1,"result":"created","shards":{"total":1,"successful":1,"failed":0},"created":true,"status":201},"index":{"_index":"server-metrics","type":"metric","id":"90711","version":1,"result":"created","shards":{"total":1,"successful":1,"failed":0},"created":true,"status":201},"index":{"_index":"server-metrics","type":"metric","id":"90712","version":1,"result":"created","shards":{"total":1,"successful":1,"failed":0},"created":true,"status":201},"index":{"_index":"server-metrics","type":"metric","id":"90713","version":1,"result":"created","shards":{"total":1,"successful":1,"failed":0},"created":true,"status":201},"index":{"_index":"server-metrics","type":"metric","id":"90714","version":1,"result":"created","shards":{"total":1,"successful":1,"failed":0},"created":true,"status":201},"index":{"_index":"server-metrics","type":"metric","id":"90715","version":1,"result":"created","shards":{"total":1,"successful":1,"failed":0},"created":true,"status":201},"index":{"_index":"server-metrics","type":"metric","id":"90716","version":1,"result":"created","shards":{"total":1,"successful":1,"failed":0},"created":true,"status":201}}}}name1@name1-VirtualBox: ~/Desktop/elasticsearch-5.5.1/bin$

name1@name1-VirtualBox: ~/Desktop/elasticsearch-5.5.1/bin$ curl 'http://localhost:9200/_cat/indices?v' -u elastic:elasticpassword
health status index      uuid      pri rep docs.count docs.deleted store.size pri.store.size
yellow open   monitoring-es-6-2017.08.23 7b0d0cd9gl-e1Gh3ua5Tg 1 1 41158 214 31.9mb 31.9mb
green open   security      dEaonMc-Ta2ku1I0SRe1_A 1 0 13 1 17.4kb 17.4kb
yellow open kibana         Mx038hs1annuP1lnXVn6 1 1 2 0 28.2kb 28.2kb
yellow open monitoring-es-6-2017.08.21 wx0MeH8Qumywe2wqen_Q 1 1 44507 99 28.7mb 28.7mb
yellow open triggered_watches gRuJlyVpQ0G97fHwmgH5A 1 1 0 0 11.6kb 11.6kb
yellow open watcher-history-3-2017.08.23 tL4ydg9pQeg7K0o1cdW 1 1 1450 0 2.6mb 2.6mb
yellow open monitoring-logstash-6-2017.08.21 2LAd8Bf3Ff6s14mwuMl9w 1 1 720 0 345.2kb 345.2kb
yellow open watches      l6dep-X1T2-T0PvJ1anLw 1 1 4 0 39.7kb 39.7kb
yellow open monitoring-kibana-6-2017.08.21 EwyuQ-1gr0JedFaa1ccXQ 1 1 1629 0 879.9kb 879.9kb
yellow open monitoring-kibana-6-2017.08.17 b1Cesd8e2Vyd5178Ccw-g 1 1 1143 0 657.7kb 657.7kb
yellow open monitoring-logstash-6-2017.08.17 ocxb5V1Q7-ctnd-tsl4Gc 1 1 86 0 84.5kb 84.5kb
yellow open watcher-history-3-2017.08.21 eFppa97K9G_CbJw8G25g 1 1 1760 0 1.4mb 1.4mb
yellow open logstash-2017.08.21 LzV1bm387uKk1FA4M2rQ 5 1 375 0 192.1kb 192.1kb
green open server-metrics 4-031fNG101n2lba4Z4Q 1 0 905940 0 111.5mb 111.5mb
yellow open monitoring-logstash-6-2017.08.23 qC0_3hwQxy_NN_gSPDGeg 1 1 1214 0 519.7kb 519.7kb
yellow open [tag[0]]-2017.08.21 u0HnFJkQ9KqL27fFv51g 1 1 3000 0 2.3mb 2.3mb
yellow open watcher-history-3-2017.08.17 hB2M61cnb_2CgldHw38Ng 1 1 995 0 927.9kb 927.9kb
yellow open monitoring-es-6-2017.08.17 yUjF0BA75_lq8uXJc0w 1 1 16850 17 10.7mb 10.7mb
yellow open monitoring-alerts-6 RG2ehMwTK0m2KANC1g3Q 1 1 1 0 12.4kb 12.4kb
name1@name1-VirtualBox: ~/Desktop/elasticsearch-5.5.1/bin$ curl 'http://localhost:9200/_cat/indices?v' -u elastic:elasticpassword
health status index      uuid      pri rep docs.count docs.deleted store.size pri.store.size
yellow open   monitoring-es-6-2017.08.23 7b0d0cd9gl-e1Gh3ua5Tg 1 1 41158 214 31.9mb 31.9mb
green open   security      dEaonMc-Ta2ku1I0SRe1_A 1 0 13 1 17.4kb 17.4kb
yellow open kibana         Mx038hs1annuP1lnXVn6 1 1 2 0 28.2kb 28.2kb
yellow open monitoring-es-6-2017.08.21 wx0MeH8Qumywe2wqen_Q 1 1 44507 99 28.7mb 28.7mb
yellow open triggered_watches gRuJlyVpQ0G97fHwmgH5A 1 1 0 0 11.6kb 11.6kb
yellow open watcher-history-3-2017.08.23 tL4ydg9pQeg7K0o1cdW 1 1 1710 0 2.8mb 2.8mb
yellow open monitoring-logstash-6-2017.08.21 2LAd8Bf3Ff6s14mwuMl9w 1 1 720 0 345.2kb 345.2kb
yellow open watches      l6dep-X1T2-T0PvJ1anLw 1 1 4 0 31.5kb 31.5kb
yellow open monitoring-kibana-6-2017.08.21 EwyuQ-1gr0JedFaa1ccXQ 1 1 1629 0 879.9kb 879.9kb
yellow open .n1.notifications -Ushw61lad-3nZwFhMQ 1 1 16 0 33.5kb 33.5kb
yellow open monitoring-kibana-6-2017.08.17 b1Cesd8e2Vyd5178Ccw-g 1 1 1143 0 657.7kb 657.7kb
yellow open monitoring-logstash-6-2017.08.17 ocxb5V1Q7-ctnd-tsl4Gc 1 1 86 0 84.5kb 84.5kb
yellow open watcher-history-3-2017.08.21 eFppa97K9G_CbJw8G25g 1 1 1760 0 1.4mb 1.4mb
yellow open .n1.state       jM81Vyn1n6sKxJY31lvvg 5 1 375 0 48.8kb 48.8kb
green open server-metrics 4-031fNG101n2lba4Z4Q 1 0 905940 0 111.5mb 111.5mb
yellow open monitoring-logstash-6-2017.08.23 qC0_3hwQxy_NN_gSPDGeg 1 1 1214 0 519.7kb 519.7kb
yellow open [tag[0]]-2017.08.21 u0HnFJkQ9KqL27fFv51g 1 1 3000 0 2.3mb 2.3mb
yellow open .n1.notifications -Ushw61lad-3nZwFhMQ 1 1 35234 199 7.5mb 7.5mb
yellow open watcher-history-3-2017.08.17 hB2M61cnb_2CgldHw38Ng 1 1 995 0 927.9kb 927.9kb
yellow open monitoring-es-6-2017.08.17 yUjF0BA75_lq8uXJc0w 1 1 16850 17 10.7mb 10.7mb
yellow open monitoring-alerts-6 RG2ehMwTK0m2KANC1g3Q 1 1 1 0 12.4kb 12.4kb
yellow open monitoring-kibana-6-2017.08.23 PF-KJ3yeTm-z5QyP5A3A 1 1 1961 0 1.7mb 1.7mb
name1@name1-VirtualBox: ~/Desktop/elasticsearch-5.5.1/bin$

```

Figure 11

After restarting elastic, data can be analyzed. In machine learning section on kibana, create a job, select create a single metric, a select time interval (time can be selected as an interval or analysis can be done in a real job which is very important for real-time analysis for the servers) then filters are specified like count or sum etc. Then the key field is selected as host, server etc. After filled the job id, the job can be created and analysis is begun as Figure-13.



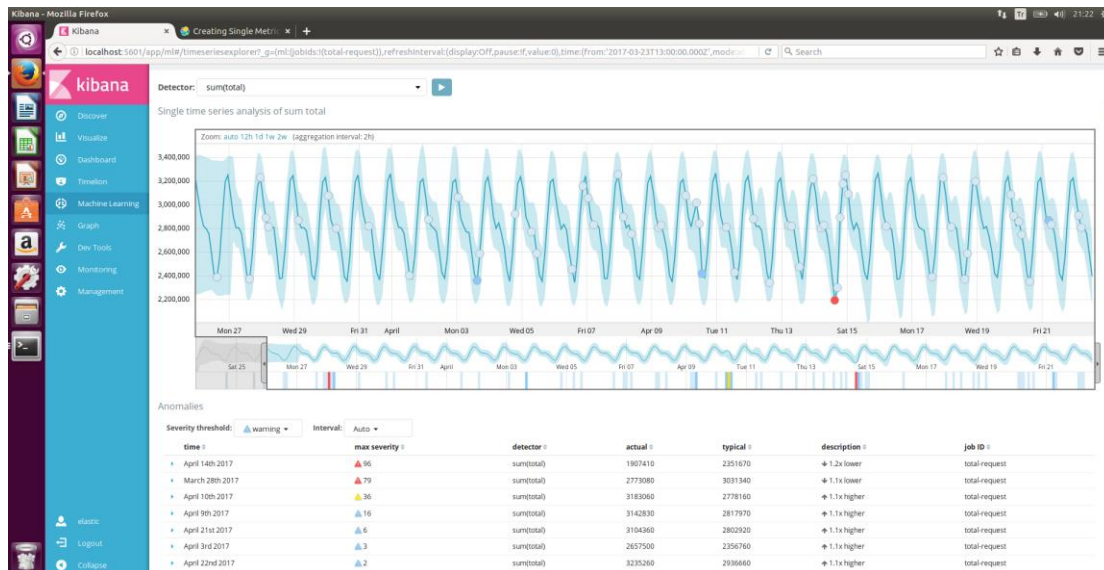
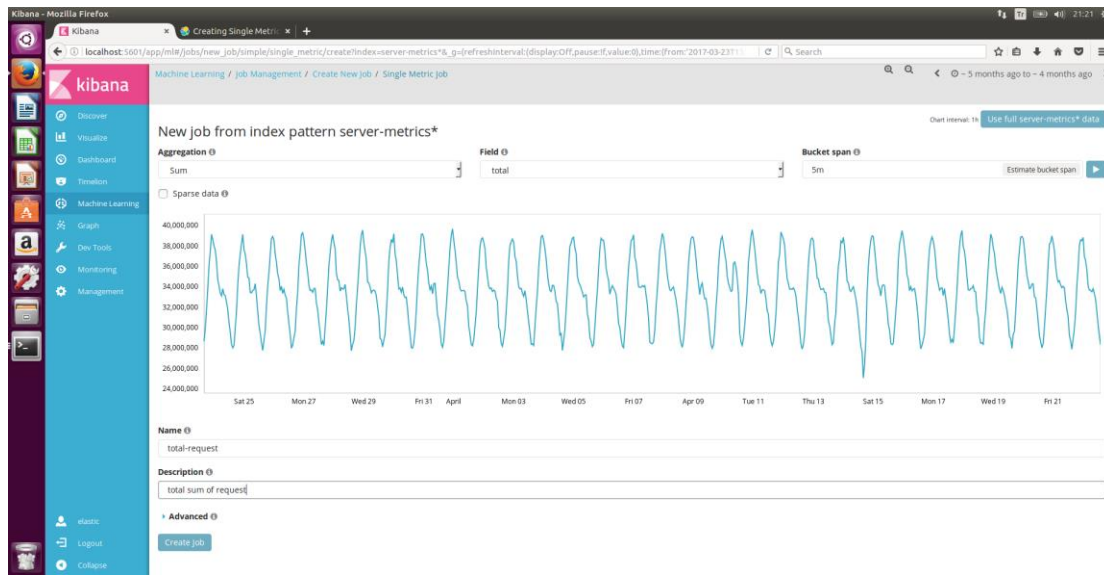


Figure 12

In Figure-13, after machine learning algorithms start to analyze the single metric data, it starts the training and understanding for the first three or four-wave [10]. After that, it understands the analogous behavior of the data changes. When it continuous the analyzing, it improves the correctness of analysis and job is finishes at the end or can continue in real time. This is the single metric view that shows the critical situations in data.

Analyzing interval can be increased and the big area can be seen or decreased to see the narrow area in more detail. Severity threshold is grouped by three distinct type like critical, warning etc. Critical type shows that there is an anomaly and it also shows the max severity, probability features to analyze the situation. In Figure-14, anomaly explorer displays the anomaly points by red squares that are clickable. By selecting the square, it shows the anomaly graph and anomalies with features.

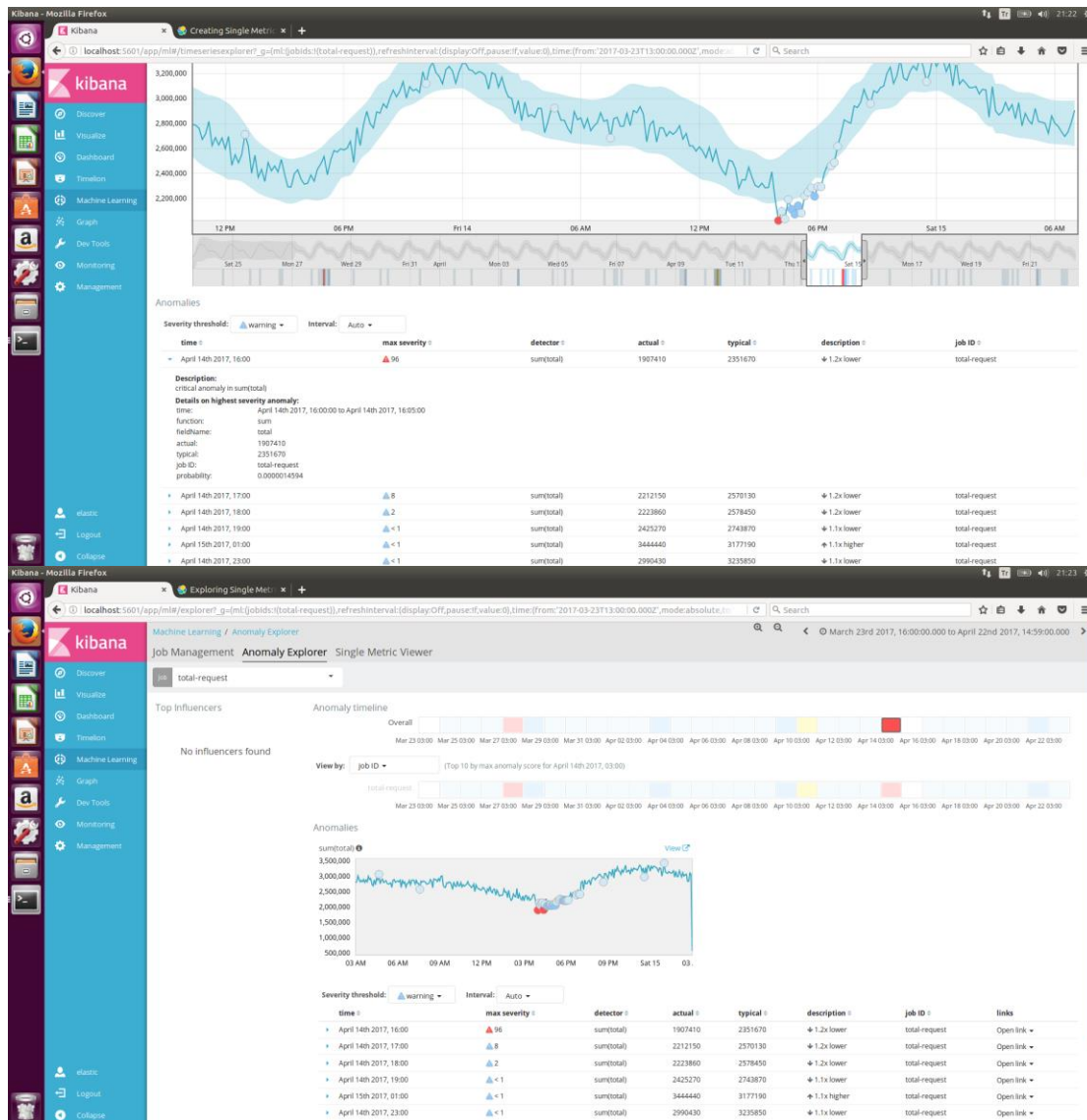


Figure 13

Same things are done for a multi-metric job that is similar to single metric but it can combine multiple of the metrics with different fields and key fields. So, whenever one anomaly occurs in one of the metrics, it combines all of them and the same detection is done as Figure-15 photos [11]. The multimetric job is efficient because lots of different problems can occur in hosts and servers. By interpreting the similar behaviors on the hosts and servers, the problem can be fixed fast.

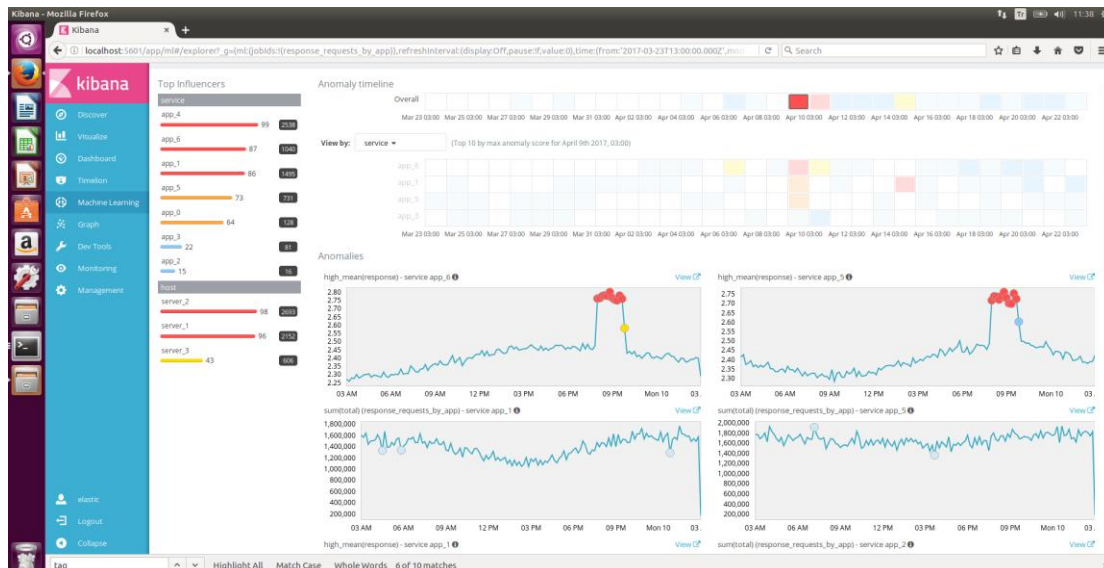
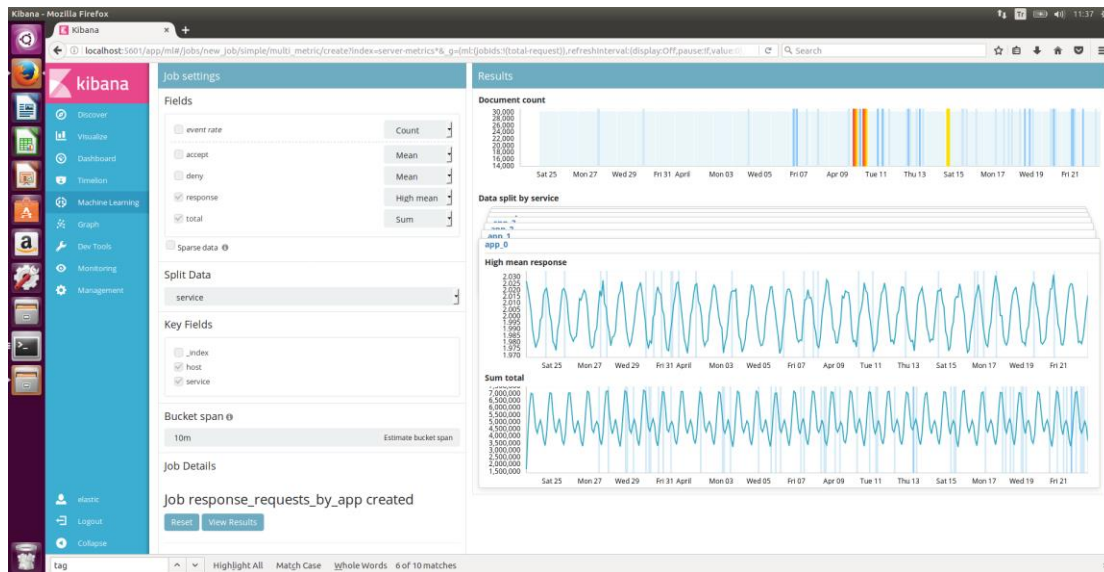


Figure 14

Sometimes monitoring fails after loading more data into elasticsearch as Figure-16. To handle this problem, monitoring status should be fixed from “red” to “green” [12]. For example, if one of monitoring status seen as “red”, all monitoring can be deleted as following by opening terminal in elasticsearch-5.5.1 folder;

```
curl -u elastic:elasticpassword -XDELETE localhost:9200/.monitoring-*
```

Also, even mobilbanka or kredikarti indexes deleted from kibana, it is not deleted from elasticsearch completely because it creates different kredikarti and mobilbanka indexes by time. To handle this issue, indexes fully deleted from elasticsearch and all elasticsearch, logstash, filebeat etc. should be restarted;

```
curl -u elastic:elasticpassword -XDELETE 'localhost:9200/mobilbanka*?pretty'
```

```
curl -u elastic:elasticpassword -XDELETE 'localhost:9200/kredikarti*?pretty'
```

After that following line is seen;

```
{  
  "acknowledged" : true  
}
```

After that, the problem is fixed as Figure-17.

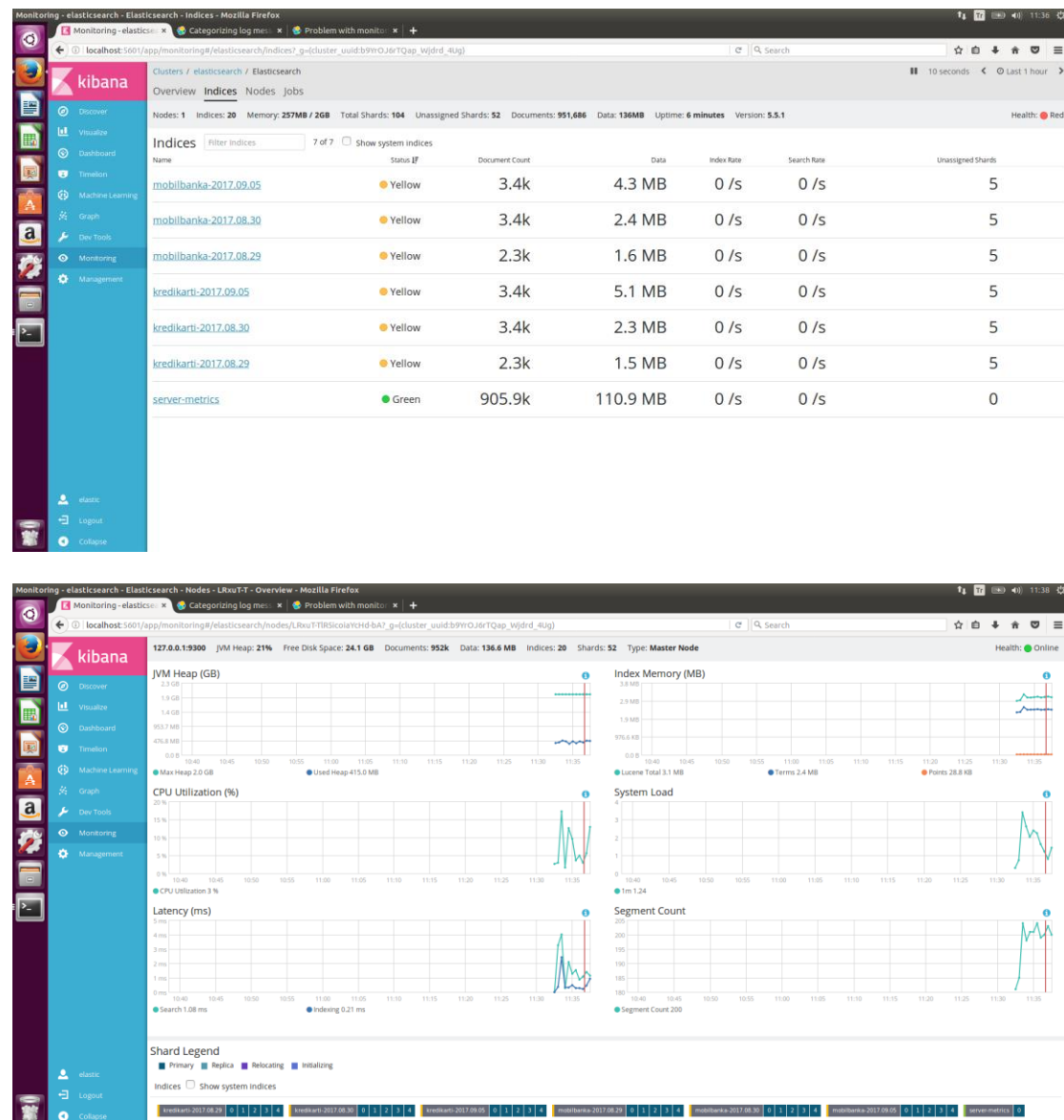


Figure 15

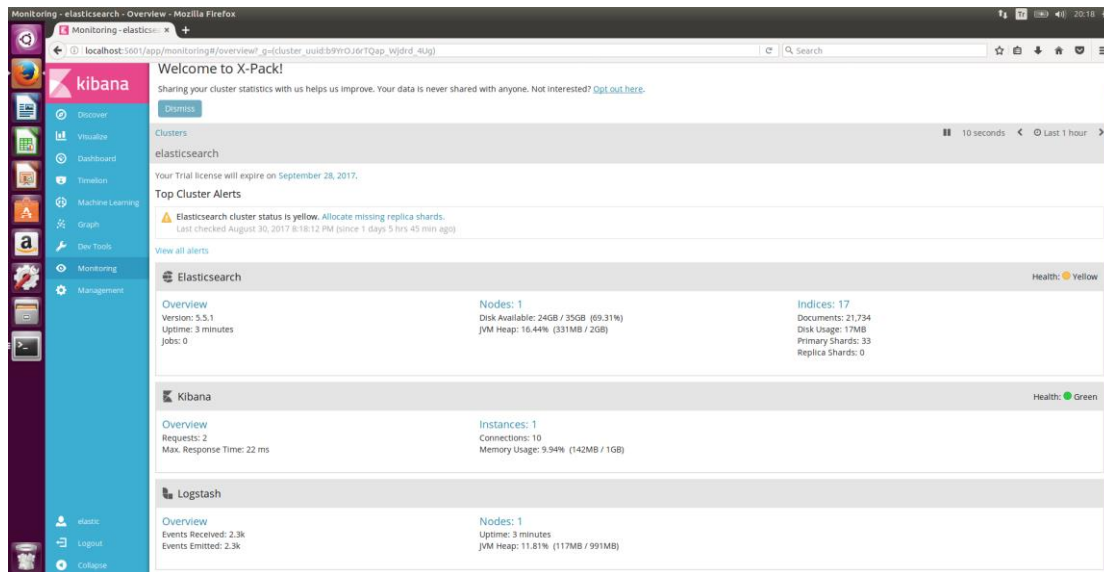


Figure 16

Another important feature is alerting. Sometimes CPU usage is unexpectedly increasing. Application response time is spiking. 503 errors are skyrocketing or elasticsearch indexing rate is has plummeted. To know all kinds of error to fix later, alerting via watches is needed. In this task, alerting is used for machine learning feature. Necessary configuration can be done via Dev Tools feature or Management section [13]. After elasticsearch.conf is regulated properly, watches can be used for anomaly detection as seen in Figure-18. This shows in every one minute, email is sent to specified email address.

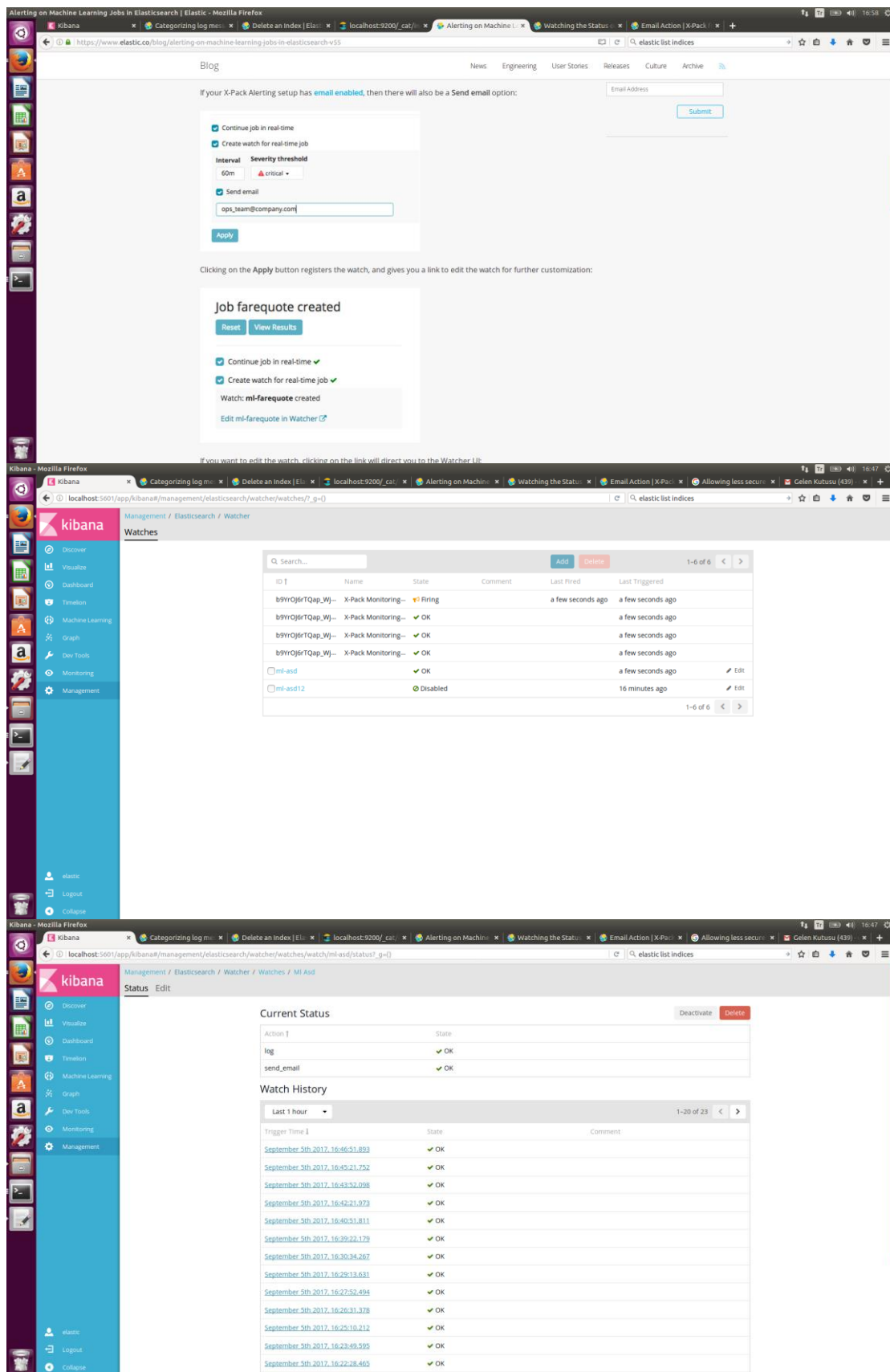


Figure 17

4 Performance and Outcomes

4.1 Applying Knowledge and Skills Learned at Bilkent

I could not use the technical features I learned during the internship because the technologies and information that I used here were different than what I learned at school. However, a very important feature that I learned at school contributed to my study greatly was “studying continuously”. School gave me studying and researching skills to solve the problems and finding solutions. Linux knowledge was the only one that I used before.

4.2 Solving Engineering Problems

I used some of the valuable websites to learn the process and work. Elastic.co and StackOverflow websites were main resources that I was checked while studying. Defining problem statements and using given/finds were related to discussion and conversations about the elastic stack. In some parts, elastic stack did not work well but after doing research and asking questions, problems are solved. In the equation/analysis part, data were compared the previous samples and future data behaviors were analyzed by using previous data. In other words, the defined process of analyzing elastic stack was applied according to previous works that people had done.

4.3 Team Work

During the internship, each intern was assigned to the specific team that each one has different tasks to do. My team has approximately ten members and some of them assigned me tasks about the elastic stack. Sometimes team leader checked my status, and almost every day some of the team members gave me tasks and checked the status regularly. So, I was a part of teamwork but did my job alone, meanwhile, I got help from them.

4.4 Multi-Disciplinary Work

In the IT department, there was no obvious multidisciplinary work with other disciplines. In the department, the majority of the people are graduated from computer engineering, electrical engineering departments etc. and all of them was an engineer, therefore all of the team members from engineer to project manager, they have the same disciplinary. But some guest company engineers worked with IT department of İş Bank.

4.5 Professional and Ethical Issues

Everyone in this company knows how to do their jobs professionally. In-out times, service hours, communication and behaviors between people etc. are proceeding planned and smoothly. Rules and procedures are clear and people are aware of it. This leads the professionalism of the company to the next level. Ethical values and company confidentiality are also very important that company provides a reliable environment for people.

4.6 Impact of Engineering Solutions

When elastic stack security part is done, one of the existing and emergent tasks was completed. Security provided more reliable authentication that only authenticated a user can access the specified data. Alerting and machine learning tasks were important for the future plan that by applying these features into the existing log analysis, they become getting more accurate and useful data feedbacks to detect anomalies in the servers and hosts etc.

4.7 Locating Sources and Self-Learning

Main sources for the elastic stack was elastic.co website. It includes lots of different materials such as readings, videos etc. and also there is a discussion section that provides elastic users to share complaints about their work. StackOverflow was another supportive website. During the self-learning, I read lots of materials and watched videos. I got support from engineers when I could not overcome the problems myself.

4.8 Knowledge about Contemporary Issues

While working, some information about the elastic stack in the official website were incorrect and not updated. Wrong reading material parts caused to make mistakes and delay resolving the problems. The official elastic website needs to be revised and corrected. Also, only some necessary parts were utility but some parts were very redundant for the tasks.

4.9 Using New Tools and Technologies

Elasticsearch is a search engine based on Lucene and it is most popular enterprise engine followed by Apache and developed in Java. Elasticsearch is developed alongside a data-collection and log-parsing engine called Logstash, and analytic and visualization platform called Kibana. The three integrated products refer to Elastic Stack (ELK) that İşbank uses it for its own data to analyze. I worked on the tasks by using these tools (elastic search, log stash, kibana, file beat) during the entire internship.

5 Conclusions

In conclusion, doing internship added value in terms of experiences of training Elastic Stack, the learning process of the institutional company, living Istanbul and meeting new people. Hard-working already became one of my part from the Bilkent, people at İşbank also works really hard because everybody has certain objectives. Being an Intern in this company was one of my best experience and I will use this memory after graduation.

References

- [1] “Türkiye İş Bankası”. <http://www.isbank.com.tr/TR/hakkimizda/bizi-taniyin/vizyon-ve-stratejimiz/Sayfalar/vizyon-ve-stratejimiz.aspx> [Accessed: Aug, 2017].
- [1] “Türkiye İş Bankası”.
<https://en.0wikipedia.org/index.php?q=aHR0cHM6Ly9lbi53aWtpcGVkaWEub3JnL3dpa2kvVMO8cmtpeWVfxLDFn19CYW5rYXPEsQ>. [Accessed: Aug, 2017].
- [2] “Windows WAS IIS”. [https://technet.microsoft.com/en-us/library/cc770745\(v=ws.11\).aspx](https://technet.microsoft.com/en-us/library/cc770745(v=ws.11).aspx). [Accessed: Aug, 2017].
- [3] “Elastic Stack”. <https://www.elastic.co/>. [Accessed: Aug, 2017].
- [4] “Elasticsearch running as service”.
<https://www.elastic.co/guide/en/elasticsearch/reference/current/rpm.html>. [Accessed: Aug 2017].
- [5] “Installing X-pack”. <https://www.elastic.co/guide/en/x-pack/current/installing-xpack.html>. [Accessed: Aug, 2017].
- [6] “X-pack: Securing elasticsearch and kibana”. <https://www.elastic.co/guide/en/x-pack/current/security-getting-started.html>. [Accessed: Aug 2017].
- [7] “Security: Native User Authentication”. <https://www.elastic.co/guide/en/x-pack/current/native-realm.html>. [Accessed: Aug, 2017].
- [8] “X-pack: Machine Learning in the Elastic Stack”.
<https://www.elastic.co/guide/en/x-pack/current/ml-concepts.html>. [Accessed: Aug-Sept, 2017].
- [9] “X-pack: Alerting via Watches”. <https://www.elastic.co/guide/en/x-pack/current/how-watcher-works.html>. [Accessed: Aug-Sept, 2017].
- [10] “Machine Learning: Creating single metric jobs”.
<https://www.elastic.co/guide/en/x-pack/current/ml-gs-jobs.html>. [Accessed: Aug-Sept, 2017].
- [11] “Machine Learning: Exploring multi metric job results”.
<https://www.elastic.co/guide/en/x-pack/current/ml-gs-job2-analyze.html>. [Accessed: Aug-Sept, 2017].
- [12] “X-pack: Monitoring the Elastic Stack”. <https://www.elastic.co/guide/en/x-pack/current/monitoring-details.html>. [Accessed: Sept, 2017].
- [13] “Watches: Monitoring event data”. <https://www.elastic.co/guide/en/x-pack/current/watching-meetup-data.html>. [Accessed: Sept, 2017].

Appendices

1-

```
PUT /_xpack/security/user/elastic/_password
{
  "password" : "elasticpassword"
}
```

```
PUT /_xpack/security/user/kibana/_password
{
  "password" : "kibanapassword"
}
```

```
PUT _xpack/security/user/logstash_system/_password
{
  "password": "elasticpassword"
}
```

```
PUT _xpack/security/user/logstash_system/_enable
```

```
POST /_xpack/security/role/events_admin
{
  "indices" : [
    {
      "names" : [ "events*" ],
      "privileges" : [ "all" ]
    },
    {
      "names" : [ ".kibana*" ],
      "privileges" : [ "manage", "read", "index" ]
    }
  ]
}
```

```
POST /_xpack/security/user/johndoe
{
  "password" : "userpassword",
  "full_name" : "John Doe",
  "email" : "john.doe@anony.mous",
  "roles" : [ "events_admin" ]
}
```

```
POST /_xpack/security/user/jacknich
{
  "password" : "j@rV1s",
  "roles" : [ "admin", "other_role1" ],
  "full_name" : "Jack Nicholson",
  "email" : "jacknich@example.com",
  "metadata" : {
    "intelligence" : 7
  },
  "enabled": true
}
```

```

}

POST /_xpack/security/user/yugi
{
  "password" : "yugioh",
  "full_name" : "Yugioh",
  "roles" : [ "kibana_user", "yugirole" ]
}

DELETE /_xpack/security/user/yugi

GET /kredikarti-*/_stats

DELETE /_xpack/security/role/yugirole

POST /_xpack/security/role/yugirole
{
  "run_as": [ "*" ],
  "cluster": [ "all" ],
  "indices": [
    {
      "names": [ "logstash-*" ],
      "privileges": [ "all" ]
    }
  ]
}

POST _xpack/security/role/logstash_writer
{
  "cluster": [ "manage_index_templates", "monitor" ],
  "indices": [
    {
      "names": [ "logstash-*" ],
      "privileges": [ "write", "delete", "create_index" ]
    }
  ]
}

POST _xpack/security/role/logstash_reader
{
  "indices": [
    {
      "names": [ "logstash-*" ],
      "privileges": [ "read", "view_index_metadata" ]
    }
  ]
}

POST _xpack/security/user/logstash_internal
{
  "password" : "changeme",
  "roles" : [ "logstash_writer" ],
  "full_name" : "Internal Logstash User"
}

```

POST _xpack/security/user/logstash_user

```
{
  "password" : "changeme",
  "roles" : [ "logstash_reader"],
  "full_name" : "Kibana User"
}
```

2-

server-metrics_1.json;

```
{"index": {"_index": "server-metrics", "_type": "metric", "_id": "1178"}}
{"@timestamp": "2017-03-23T13:00:00", "accept": 13324, "deny": 1963, "host": "server_3", "response": 2.0421294904, "service": "app_1", "total": 15287}
{"index": {"_index": "server-metrics", "_type": "metric", "_id": "1179"}}
{"@timestamp": "2017-03-23T13:00:00", "accept": -680, "deny": -96, "host": "server_1", "response": 2.5898591717, "service": "app_6", "total": -776}
{"index": {"_index": "server-metrics", "_type": "metric", "_id": "1180"}}
{"@timestamp": "2017-03-23T13:00:00", "accept": 9691, "deny": 1675, "host": "server_2", "response": 2.1392489997, "service": "app_0", "total": 11366}
{"index": {"_index": "server-metrics", "_type": "metric", "_id": "1181"}}
{"@timestamp": "2017-03-23T13:00:00", "accept": 3180, "deny": 426, "host": "server_3", "response": 1.992834435, "service": "app_0", "total": 3606}
{"index": {"_index": "server-metrics", "_type": "metric", "_id": "1182"}}
{"@timestamp": "2017-03-23T13:00:00", "accept": 18742, "deny": 264, "host": "server_2", "response": 2.0883402695, "service": "app_1", "total": 19006}
{"index": {"_index": "server-metrics", "_type": "metric", "_id": "1183"}}
{"@timestamp": "2017-03-23T13:00:00", "accept": 35106, "deny": 3507, "host": "server_1", "response": 2.6310828948, "service": "app_3", "total": 38613}
{"index": {"_index": "server-metrics", "_type": "metric", "_id": "1184"}}
{"@timestamp": "2017-03-23T13:00:00", "accept": 16912, "deny": 2604, "host": "server_1", "response": 2.3765833854, "service": "app_2", "total": 19516}
{"index": {"_index": "server-metrics", "_type": "metric", "_id": "1185"}}
...
```

Self-Checklist for Your Report

Please check the items here before submitting your report. This signed checklist should be the final page of your report.

- ☐ Did you provide detailed information about the work you did?
- ☐ Is supervisor information included?
- ☐ Did you use the Report Template to prepare your report, so that it has a cover page, the 8 major sections and 13 subsections specified in the Table of Contents, and uses the required section names?
- ☐ Did you follow the style guidelines?
- ☐ Do your report look professionally written?
- ☐ Does your report include all necessary References and proper citations to them in the body?
- ☐ Did you remove all explanations from the Report Template, which are marked with yellow color? Did you modify all text marked with green according to your case?

Signature: _____