



Elastic Stack Hands On Workshop

김종민 (Jongmin Kim)
Developer Evangelist @Elastic
jongmin.kim@elastic.co

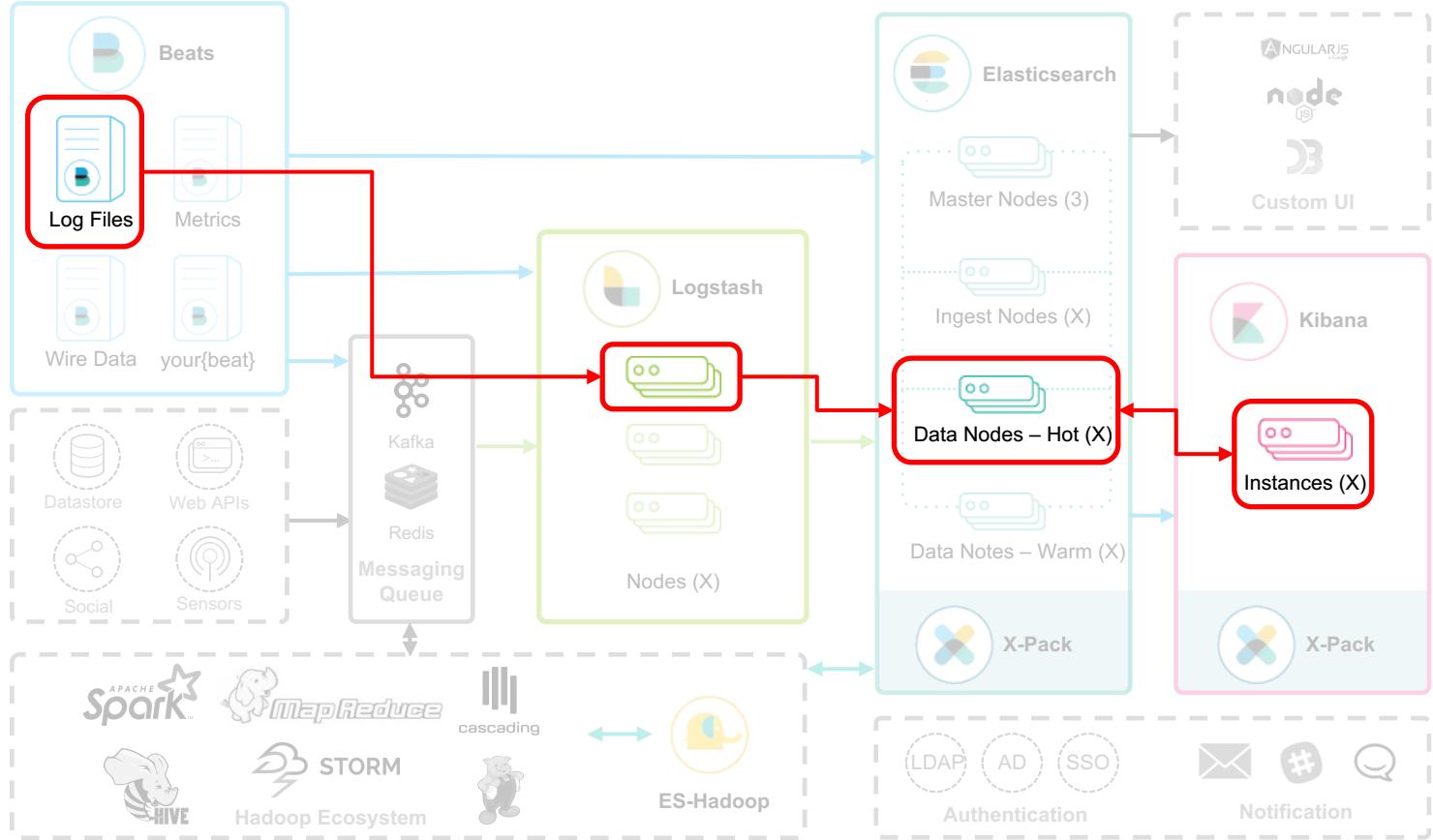


Agenda

Apache Web Log 수집 및 분석

- Elasticsearch 설치 및 실행
- Logstash를 이용한 Apache Log 파싱 및 확장
- FileBeat를 이용한 Apache Log 파일 수집
- Kibana 대시보드 만들기

Architecture



Java 설치

<https://www.java.com/ko/download/>

- 최신 Java 를 다운로드해서 설치합니다.
- Windows 운영체제의 경우 환경변수에 JAVA_HOME 을 지정 해 주어야 합니다.

실습 데이터 파일 다운로드

<https://s3.ap-northeast-2.amazonaws.com/kr.elastic.co/sample-data/weblog-sample.log.zip>

- 위 링크의 Apache Web Log 실습 파일을 다운로드 한 뒤 압축을 풀어 놓습니다.

Elastic Stack 다운로드

<https://www.elastic.co/downloads>

- Elasticsearch, Kibana, Logstash, Filebeat 를 운영체제에 맞게 다운로드 합니다.



Elasticsearch

Distributed, RESTful search and analytics.

[Download](#)



Kibana

Visualize your data. Navigate the Stack.

[Download](#)



Beats

Collect, parse, and ship in a lightweight fashion.

[Download](#)



Logstash

Ingest, transform, enrich, and output.

[Download](#)



Filebeat

Real-time insight into log data.

[Download](#)

Elasticsearch 실행 파일 생성 (Mac, Linux)

Elasticsearch 실행 / 종료 파일 생성

- 내려받은 Elasticsearch의 압축을 푼 뒤 압축을 푼 디렉토리로 이동합니다.

```
cd elasticsearch-6.0.0
```

- 실행파일 start.sh, stop.sh를 생성합니다.

```
echo 'bin/elasticsearch -d -p es.pid' > start.sh  
echo 'kill `cat es.pid`' > stop.sh  
chmod 755 start.sh stop.sh
```

- Elasticsearch 시작하고 프로세스가 떠 있는지 확인합니다.

```
./start.sh  
ps -ef | grep elasticsearch  
curl localhost:9200
```

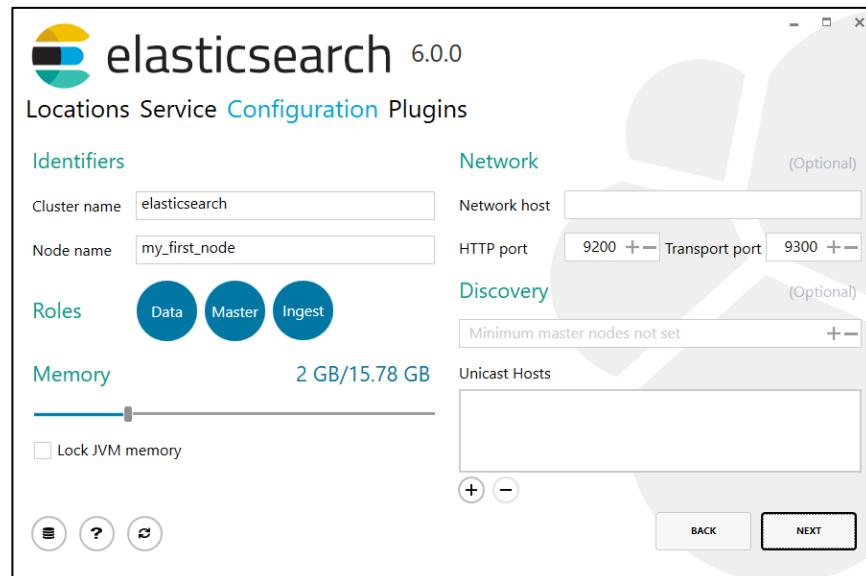
Elasticsearch 실행 (Windows)

방법 1) 개별 프로그램으로 실행

- 내려받은 파일의 압축을 푼 뒤 cmd 프로그램을 이용해 압축을 푼 디렉토리의 bin 아래에 있는 elasticsearch.bat 파일을 실행시킵니다.

방법 2) 서비스로 실행

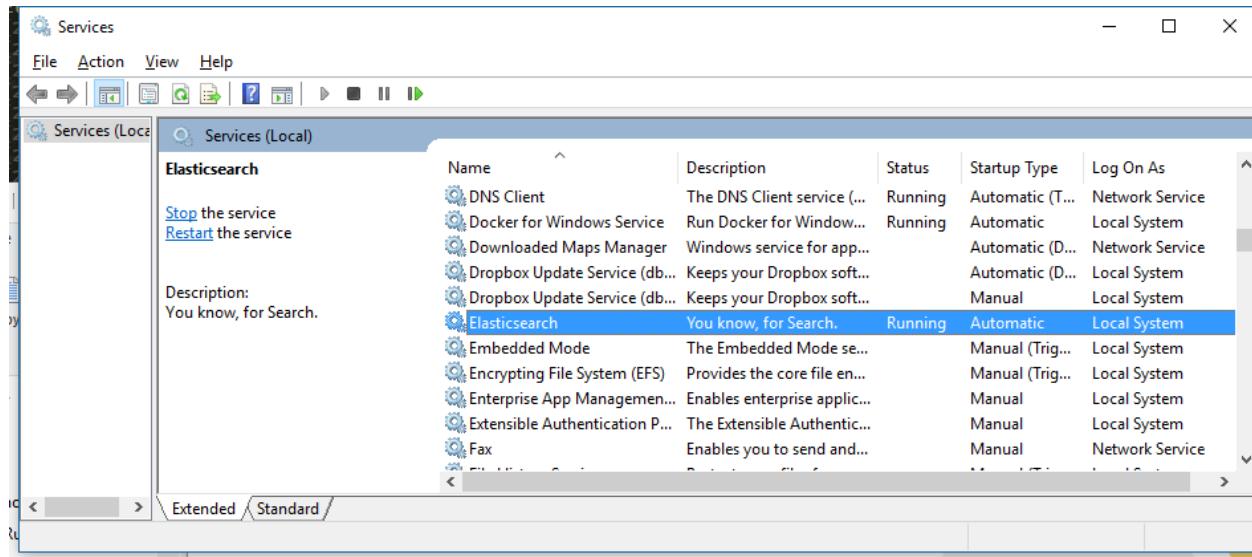
- Elasticsearch 다운로드 페이지의 MSI 파일을 내려받아 설치하면 서비스로 실행이 가능합니다.



Elasticsearch 실행 (Windows)

방법 2) 서비스로 실행 - 계속

- 서비스로 설치한 경우 [프로그램] - [서비스] 목록에서 Elasticsearch 를 찾아 실행, 중지를 시킬 수 있습니다.



Kibana 설치

<https://www.elastic.co/downloads/kibana>

- 다운로드 페이지에서 운영체제에 맞는 버전을 내려 받습니다.
- Windows의 경우 탐색기 프로그램이 아닌 반디집 등을 이용해서 압축을 푸는 것이 좋습니다.

Kibana 실행

<http://localhost:5601>

- 압축을 푼 Kibana 디렉토리로 이동 후 bin 아래의 kibana 를 실행합니다.

```
bin/kibana
```

- Windows 의 경우 cmd 에서 bin 아래의 kibana.bat 파일을 실행합니다.

```
cd bin  
kibana.bat
```

Kibana 실행

웹브라우저로 <http://localhost:5601>에 접속해서 Kibana 화면을 확인합니다.

The screenshot shows the Kibana Management interface. On the left, there is a sidebar with the Kibana logo and links to Discover, Visualize, Dashboard, Timelion, Dev Tools, and Management. The Management link is highlighted. At the bottom of the sidebar is a 'Collapse' button. The main content area has a header 'Management / Kibana' with tabs for Index Patterns, Saved Objects, and Advanced Settings. A warning message states: 'No default index pattern. You must select or create one to continue.' Below this, a section titled 'Configure an index pattern' explains that at least one index pattern must be configured to use Kibana. It shows an input field with 'logstash-*' and a warning message: '⚠ Unable to fetch mapping. Do you have indices matching the pattern?'. It also shows a dropdown for 'Time Filter field name' with a placeholder 'Time Filter field name is required'.

Logstash 설정

config/logstash.yml 설정

- 내려받은 파일의 압축을 푼 뒤 압축을 푼 디렉토리로 이동합니다.
- config 디렉토리 아래의 logstash.yml 파일을 열어 config.reload.automatic 부분의 주석을 해제하고 값을 true로 합니다.

```
config.reload.automatic: true
```

- 이 때 설정값 앞에 빈 칸 띄어쓰기가 없어야 합니다.

Logstash 실행 및 테스트 (Mac, Linux)

weblog.conf 파일 생성

- weblog.conf 파일을 생성하여 다음과 같은 내용을 입력합니다.

```
input {  
  tcp {  
    port => 9900  
  }  
}  
  
output {  
  stdout { }  
}
```

- weblog.conf 파일을 설정으로 Logstash를 실행합니다.

```
bin/logstash -f weblog.conf
```

Logstash 실행 및 테스트 (Mac, Linux)

input / output 테스트

- netcat 을 이용해서 Logstash로 메시지를 테스트 해 봅니다.

```
echo 'Hello World' | nc localhost 9900
```

- 출력 코덱을 rubydebug 형식으로 바꿔서 다시 테스트 해 봅니다.

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

Logstash 실행 및 테스트 (Mac, Linux)

Apache Web Log 테스트

- netcat 을 이용해서 Logstash로 Apache Web Log 실습 데이터를 입력해 봅니다.

```
echo '14.49.42.25 -- [12/Mar/2015:01:24:44 +0000] "GET /articles/ppp-over-ssh/ HTTP/1.1" 200 18586 "-"  
"Mozilla/5.0 (Windows; U; Windows NT 6.1; en-US; rv:1.9.2b1) Gecko/20091014 Firefox/3.6b1 GTB5"' | nc  
localhost 9900
```

Logstash Filter 설정 – GROK

grok 필터 설정

- grok 필터를 이용해서 Apache Web Log를 파싱할 수 있습니다.

```
Input ...
```

```
filter {  
  grok {  
    match => { "message" => "%{COMBINEDAPACHELOG}" }  
  }  
}
```

```
output ...
```

- 필터를 추가한 후 테스트 데이터를 다시 입력해서 메시지가 파싱된 것을 확인합니다.

```
echo '14.49.42.25 - - [12/Mar/2015:01:24:44 +0000] "GET /articles/ppp-over-ssh/ HTTP/1.1" 200 18586 "-"  
"Mozilla/5.0 (Windows; U; Windows NT 6.1; en-US; rv:1.9.2b1) Gecko/20091014 Firefox/3.6b1 GTB5"' | nc  
localhost 9900
```

Logstash Filter 설정 – geoip

geoip 설정

- geoip 필터를 이용해서 ip 주소로 부터 위치정보 값을 가져올 수 있습니다.
- grok 뒤에 geoip 필터 정보를 입력합니다.

```
filter {  
    grok...  
  
    geoip {  
        source => "clientip"  
    }  
}
```

- 필터를 추가한 후 테스트 데이터를 다시 입력합니다.

Logstash Filter 설정 – useragent

useragent 설정

- useragent 필터를 이용해서 접속한 클라이언트의 브라우저, 운영체제 정보 등을 가져올 수 있습니다.
- geoip 뒤에 useragent 필터 정보를 입력합니다.

```
geoip...  
  
useragent {  
    source => "agent"  
    target => "useragent"  
}
```

- 필터를 추가한 후 테스트 데이터를 다시 입력합니다.

Logstash Filter 설정 – mutate : convert

mutate : convert 설정

- 출력 결과 중 bytes 필드는 문자열이 아닌 숫자값으로 되어야 합니다.
mutate / convert 옵션을 이용해서 bytes 필드를 integer 값으로 변경합니다.
- mutate 필터 정보를 입력합니다.

```
useragent ...
```

```
mutate {  
    convert => {  
        "bytes" => "integer"  
    }  
}
```

- 필터를 추가한 후 테스트 데이터를 다시 입력합니다.

Logstash Filter 설정 – date

date 설정

- @timestamp 필드가 실제 데이터 생성 시간이 아닌 로그스태시의 색인 시간이므로 date 필터를 이용하여 timestamp 필드의 값이 @timestamp 필드에 저장되도록 합니다.
- mutate 필터 정보를 입력합니다.

```
mutate...  
  
date {  
    match => [ "timestamp", "dd/MMM/yyyy:HH:mm:ss Z" ]  
}
```

- 필터를 추가한 후 테스트 데이터를 다시 입력합니다.

Logstash – Elasticsearch로 데이터 색인

output : elasticsearch 설정

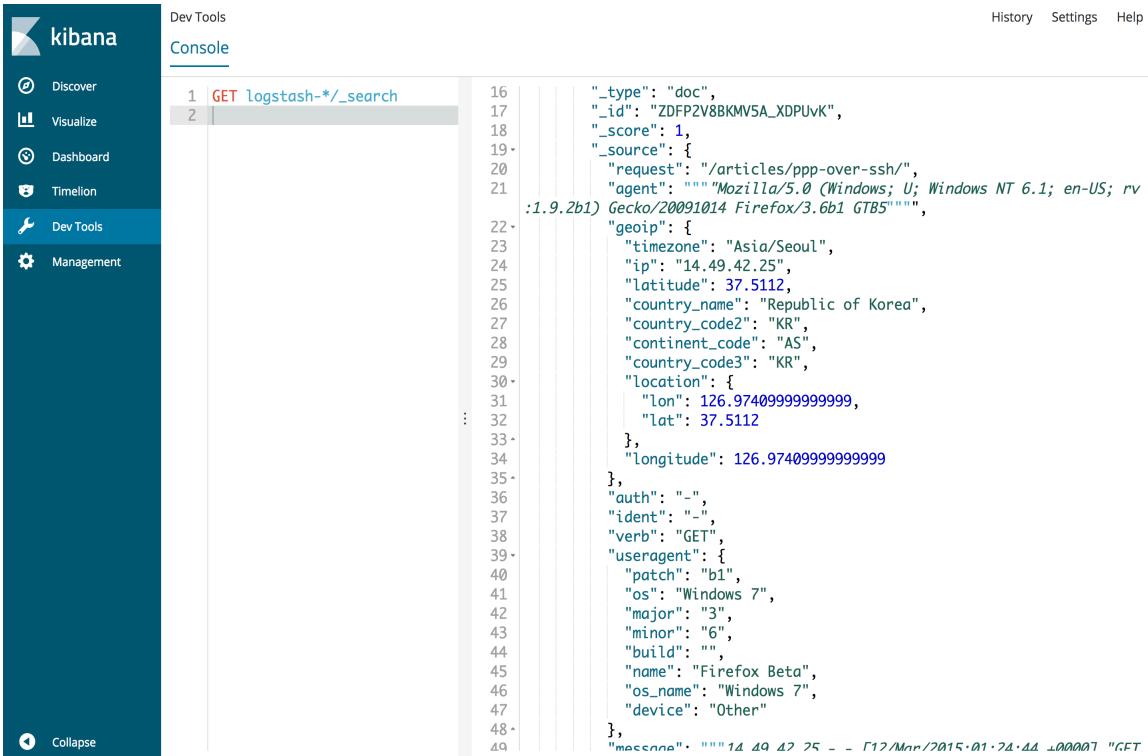
- stdout – 콘솔로 되어 있는 출력을 elasticsearch 로 변경합니다.
- 별도의 설정을 하지 않으면 localhost:9200 서버에 logstash-yyyy-MM-dd 형식으로 저장됩니다.

```
output {  
#   stdout {  
#     codec => "rubydebug"  
#   }  
  
  elasticsearch {}  
  
}
```

- output을 elasticsearch로 변경한 후 테스트 데이터를 다시 입력합니다.

Elasticsearch 입력 데이터 확인

Kibana에서 **GET logstash-*/_search** 명령으로 입력된 데이터 확인합니다.



The screenshot shows the Kibana interface with the Dev Tools tab selected. In the console, a search query is run against the logstash-* index pattern:

```
1 | GET logstash-*/_search
2 |
```

The response shows a single document with the following details:

```
16 |     "_type": "doc",
17 |     "_id": "ZDFP2V8BKMV5A_XDPUvK",
18 |     "_score": 1,
19 |     "_source": {
20 |       "request": "/articles/ppp-over-ssh/",
21 |       "agent": """Mozilla/5.0 (Windows; U; Windows NT 6.1; en-US; rv:
22 | :1.9.2b1) Gecko/20091014 Firefox/3.6b1 GTB5""",
23 |       "geoip": {
24 |         "timezone": "Asia/Seoul",
25 |         "ip": "14.49.42.25",
26 |         "latitude": 37.5112,
27 |         "country_name": "Republic of Korea",
28 |         "country_code2": "KR",
29 |         "continent_code": "AS",
30 |         "country_code3": "KR",
31 |         "location": {
32 |           "lon": 126.97409999999999,
33 |           "lat": 37.5112
34 |         },
35 |         "longitude": 126.97409999999999
36 |       },
37 |       "auth": "-",
38 |       "ident": "-",
39 |       "verb": "GET",
40 |       "useragent": {
41 |         "patch": "b1",
42 |         "os": "Windows 7",
43 |         "major": "3",
44 |         "minor": "6",
45 |         "build": "",
46 |         "name": "Firefox Beta",
47 |         "os_name": "Windows 7",
48 |         "device": "Other"
49 |       },
50 |       "messsage": """14 49 42 25 - 17/Mar/2015:01:24:44 +000007 "GET
51 |       "messsage": """14 49 42 25 - 17/Mar/2015:01:24:44 +000007 "GET
52 |     }
```

Filebeat 설정

filebeat.yml 설정

- 내려받은 파일의 압축을 푼 뒤 압축을 푼 디렉토리로 이동합니다.
- filebeat.yml 파일을 열어 **type** 이하 부분에
 - `enabled: true` 로 입력합니다.
 - `paths` 부분에 우리가 읽어들일 Weblog 예제 파일을 지정합니다. 와일드카드 (*) 의 사용이 가능합니다.
 - 띄어쓰기를 정확히 지켜야 하며 `paths:` 는 앞에서 두칸, `- /...` 부분은 네칸의 공백을 띄어야 합니다.
 - 상대 경로가 아닌 절대 경로를 입력해야 합니다.

```
- type: log  
  enabled: true  
  paths:  
    - /elastic/data/*.log
```

Filebeat 실행

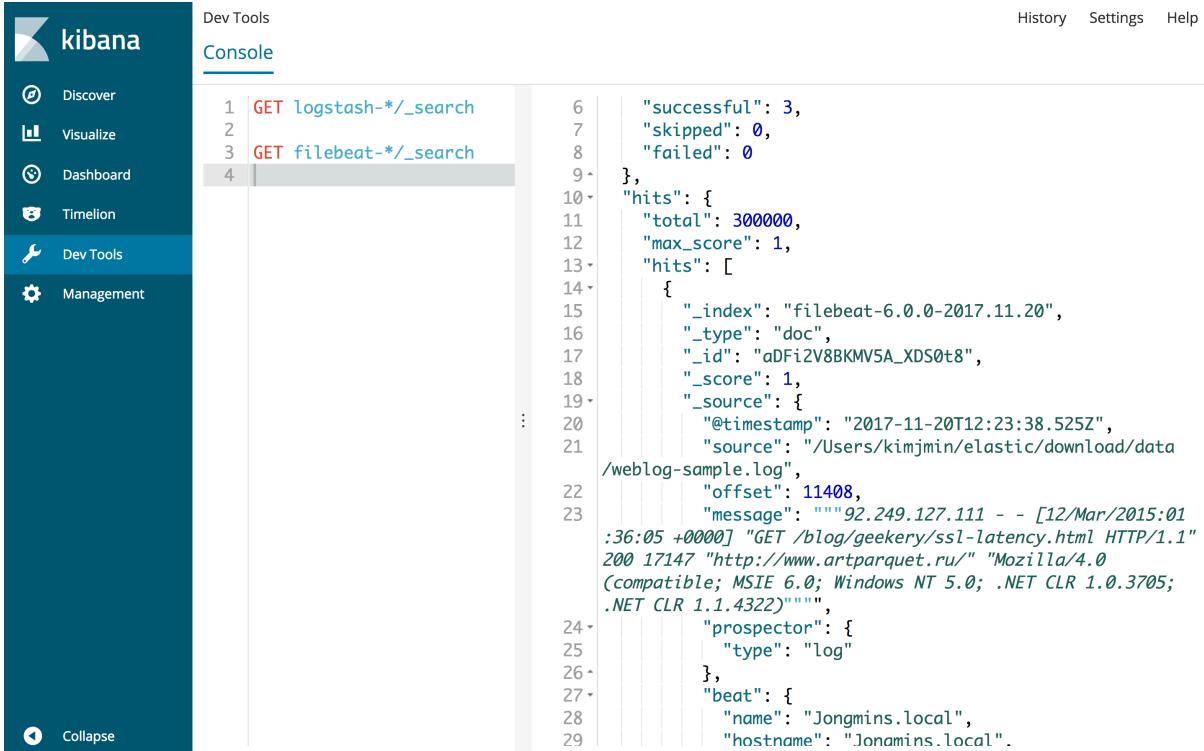
Filebeat 를 이용해서 Elasticsearch 로 데이터 입력

- Mac, Linux는 **filebeat**, Windows는 **filebeat.exe** 를 실행합니다.
- 옵션을 입력하지 않으면 콘솔에 아무런 출력이 나타나지 않습니다.
 - -e 옵션을 입력하면 콘솔에 메시지가 출력됩니다.
 - -c 옵션을 이용하면 현재 디렉토리가 아닌 다른 경로에 있는 filebeat.yml 설정 파일의 사용이 가능합니다.

```
./filebeat -e -c filebeat.yml
```

Elasticsearch 입력 데이터 확인

Kibana에서 **GET filebeat-*/_search** 명령으로 입력된 데이터 확인합니다.



The screenshot shows the Kibana interface with the Dev Tools tab selected. In the console, two search queries are listed:

```
1 GET logstash-*/_search
2
3 GET filebeat-*/_search
4
```

The third query, "GET filebeat-*/_search", is expanded to show its results. The output is a JSON object with the following structure:

```
6   "successful": 3,
7   "skipped": 0,
8   "failed": 0
9 },
10 "hits": {
11   "total": 300000,
12   "max_score": 1,
13   "hits": [
14     {
15       "_index": "filebeat-6.0.0-2017.11.20",
16       "_type": "doc",
17       "_id": "aDFi2V8BKMV5A_XDS0t8",
18       "_score": 1,
19       "_source": {
20         "@timestamp": "2017-11-20T12:23:38.525Z",
21         "source": "/Users/kimjmin/elasticsearch/download/data
22         /weblog-sample.log",
23         "offset": 11408,
24         "message": """92.249.127.111 - - [12/Mar/2015:01
25           :36:05 +0000] "GET /blog/geekery/ssl-latency.html HTTP/1.1"
26           200 17147 "http://www.artparquet.ru/" "Mozilla/4.0
27           (compatible; MSIE 6.0; Windows NT 5.0; .NET CLR 1.0.3705;
28           .NET CLR 1.1.4322)""",
29         "prospector": {
30           "type": "log"
31         },
32         "beat": {
33           "name": "Jongmins.local",
34           "hostname": "Jongmins.local".
35         }
36       }
37     }
38   ]
39 }
```

Filebeat → Logstash → Elasticsearch 설정

filebeat.yml 파일의 출력을 elasticsearch에서 logstash로 변경

- Filebeat의 filebeat.yml 파일을 열어 === Outputs === 부분에
 - output.elasticsearch: 이하 부분을 삭제 또는 주석처리 합니다.
 - output.logstash.hosts: ["localhost:5044"] 부분의 주석을 해제합니다.
- 띄어쓰기를 정확히 지켜야 하며
 - output.logstash: 는 띄어쓰기 없이,
 - hosts: ["localhost:5044"] 부분은 앞에서 두칸을 띄어씁니다.

```
#output.elasticsearch:  
#  hosts: ["localhost:9200"]
```

```
output.logstash:  
  hosts: ["localhost:5044"]
```

Filebeat → Logstash → Elasticsearch 설정

logstash.yml 파일의 입력을 beats로 변경

- Logstash의 logstash.yml 파일을 열어 input 부분을 tcp에서 beat로 변경합니다.

```
input {  
#  tcp { port => 9900 }  
  
  beats {  
    port => 5044  
  }  
}
```

Filebeat → Logstash → Elasticsearch 설정

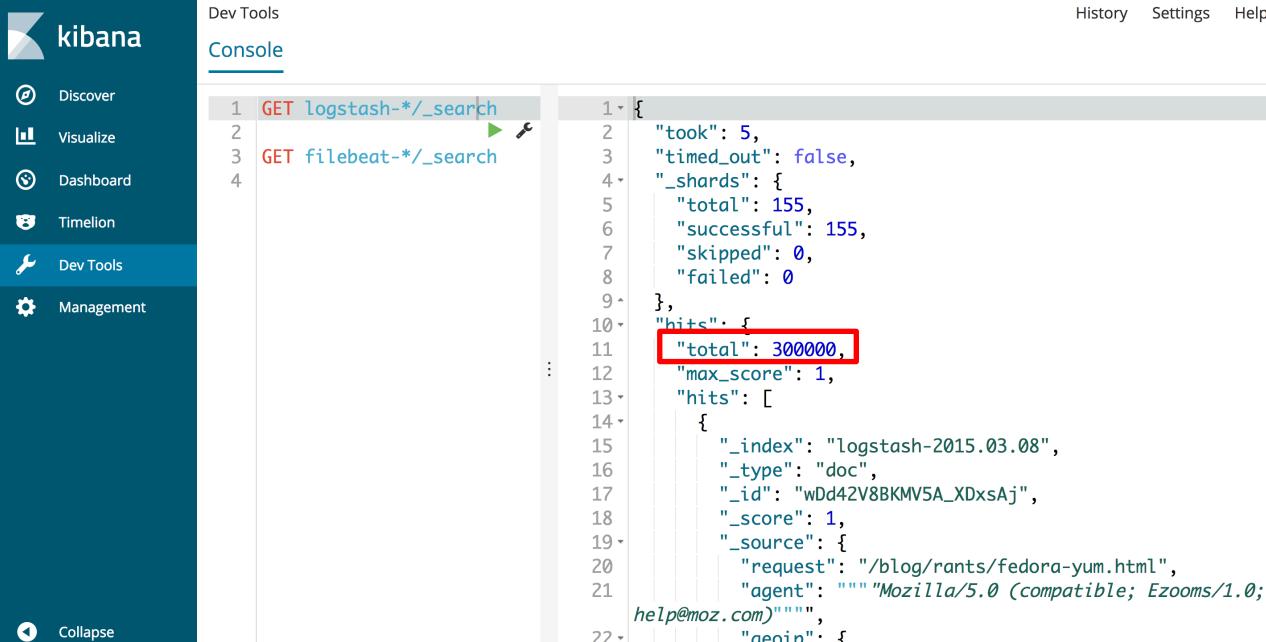
Filebeat 실행

- Filebeat은 한번 읽어들인 데이터는 다시 읽지 않기 때문에 data/registry 파일을 먼저 삭제해야 합니다.
- registry 파일 삭제 후 Filebeat 을 다시 실행합니다.

```
rm data/registry  
./filebeat -e -c filebeat.yml
```

Elasticsearch 입력 데이터 확인

Kibana에서 **GET logstash-*/_search** 명령으로 입력된 데이터 확인합니다.
테스트 데이터의 도큐먼트 수는 총 300,000 입니다.



The screenshot shows the Kibana interface with the Dev Tools section selected. In the console, two queries are run:

```
1 GET logstash-*/_search
2
3 GET filebeat-*/_search
4
```

The results for the first query are displayed as JSON. A red box highlights the "total": 300000 field, indicating the total number of documents found.

```
1 {
2   "took": 5,
3   "timed_out": false,
4   "_shards": {
5     "total": 155,
6     "successful": 155,
7     "skipped": 0,
8     "failed": 0
9   },
10  "hits": {
11    "total": 300000, // Red box here
12    "max_score": 1,
13    "hits": [
14      {
15        "_index": "logstash-2015.03.08",
16        "_type": "doc",
17        "_id": "wDd42V8BKMV5A_XDxsAj",
18        "_score": 1,
19        "_source": {
20          "request": "/blog/rants/fedora-yum.html",
21          "agent": """Mozilla/5.0 (compatible; Ezooms/1.0;
help@moz.com)""",
22          "agent": """
23        }
24      }
25    ]
26  }
27 }
```

Kibana 인덱스 패턴 생성

- Management → Index Patterns 로 이동합니다.

- Index pattern 에 **logstash-*** , Time Filter field name 에 **@timestamp** 를 선택하고 **Create** 버튼을 누릅니다.

Management / Kibana

Index Patterns Saved Objects Advanced Settings

Warning
No default index pattern.
You must select or create one to continue.

Configure an index pattern

In order to use Kibana you must configure at least one index pattern. Index patterns are used to identify the Elasticsearch index to run search and analytics against. They are also used to configure fields.

Index pattern [advanced options](#)
logstash-*

Patterns allow you to define dynamic index names using * as a wildcard. Example: logstash-*

Time Filter field name [refresh fields](#)
@timestamp

Create

Kibana 인덱스 패턴 생성

- logstash-* 인덱스 패턴이 생성되었습니다.

The screenshot shows the Kibana Management interface with the 'Index Patterns' tab selected. A prominent green star icon next to the pattern name indicates it is the active index pattern. Below the pattern name, a note specifies the 'Time Filter field name: @timestamp'. The main table lists core fields with their types and various configuration options like format, searchable, aggregatable, and excluded status. Fields listed include @timestamp (date), @version (string), _id (string), _index (string), _score (number), and _source (_source). Each row in the table includes a pencil icon for editing controls.

name	type	format	searchable	aggregatable	excluded	controls
@timestamp	date		✓	✓		
@version	string		✓	✓		
_id	string		✓	✓		
_index	string		✓	✓		
_score	number					
_source	_source					

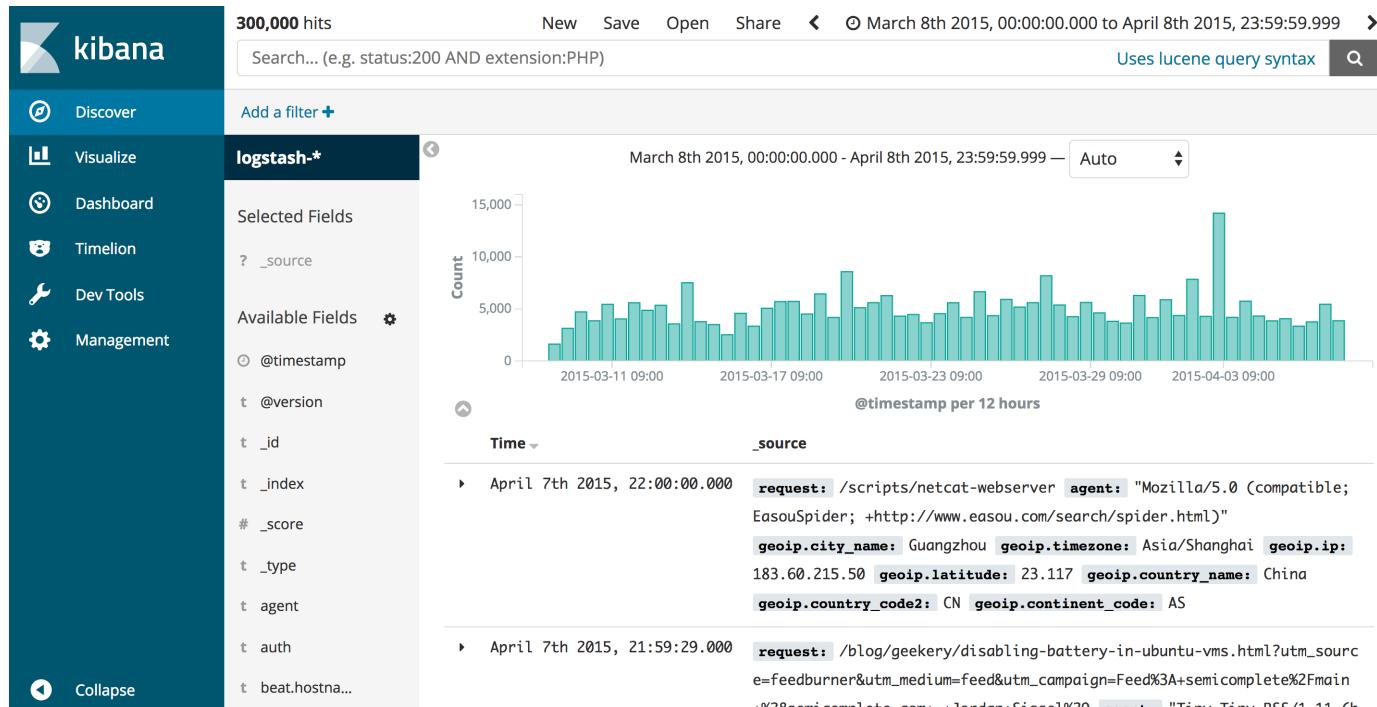
Kibana Discover 데이터 확인

- Discover 메뉴에서 데이터를 확인합니다.
 - 우측 상단의 Time Range 조절 옵션을 클릭하여 날짜를 2015년 3월 ~ 4월로 선택합니다.

The screenshot shows the Kibana Discover interface. On the left is a sidebar with icons for Discover, Visualize, Dashboard, Timelion, Dev Tools, and Management. The main area displays a search bar with placeholder text "Search... (e.g. status:200 AND extension:PHP)" and a "Go" button. Below the search bar is a "Add a filter +". At the bottom, there is a dark bar with the text "logstash-*". The central part of the screen features a "Time Range" section with "From" and "To" fields set to "2015-04-08 00:00:00.000" and "2015-04-08 23:59:59.999" respectively. Below these fields is a date picker for April 2015, showing the days from 01 to 30. The day "08" is highlighted in a blue box. At the top right of the interface, there are buttons for "New", "Save", "Open", "Share", "Auto-refresh", and a dropdown menu currently set to "Last 15 minutes".

Kibana Discover 데이터 확인

- 날짜 변경 후 데이터 확인이 가능합니다.



Kibana Visualize

- Visualize 메뉴에서 각 시각화 도구들을 만들 수 있습니다.

The screenshot shows the Kibana Visualize interface. On the left is a sidebar with icons for Discover, Visualize (selected), Dashboard, Timeline, Dev Tools, and Management. The main area has a title "Select visualization type" with a search bar. It is organized into sections: "Basic Charts" (Area, Heat Map, Horizontal Bar, Line, Pie, Vertical Bar), "Data" (DataTable, Gauge, Goal, Metric), and "Maps" (Coordinate Map, Region Map).

Select visualization type

Search visualization types...

Basic Charts

- Area
- Heat Map
- Horizontal Bar
- Line
- Pie
- Vertical Bar

Data

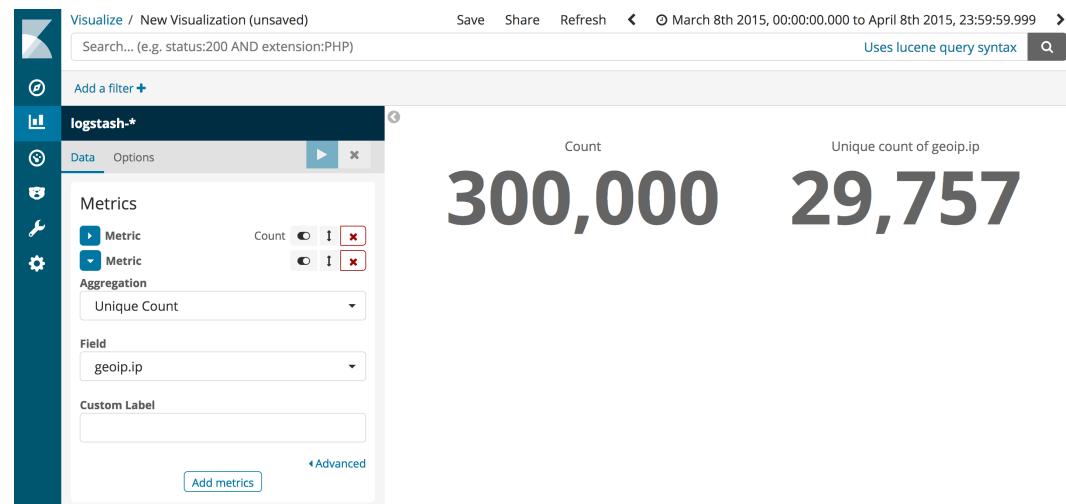
- DataTable
- Gauge
- Goal
- Metric

Maps

- Coordinate Map
- Region Map

Kibana Visualize – Metric

- 전체 문서 수와 개별 IP 수를 집계하는 통계를 만들도록 하겠습니다.
 - Metric을 선택한 후 logstash-* 패턴을 선택하면 전체 문서 수가 나타납니다.
 - Add metrics 를 클릭한 뒤 Metric 을 선택합니다.
 - Aggregation 에서 Unique Count를 선택한 뒤 Field 에서 geoip.ip 를 선택합니다.
 - 변경 내역을 반영하려면 옵션 옆의 버튼을 눌러야 반영이 됩니다.
 - 상단의 Save 눌러 완성된 시각화 도구를 저장합니다.



Kibana Visualize – Bar Chart

- 시간대 별 응답 코드 Bar Chart 를 만들도록 하겠습니다.

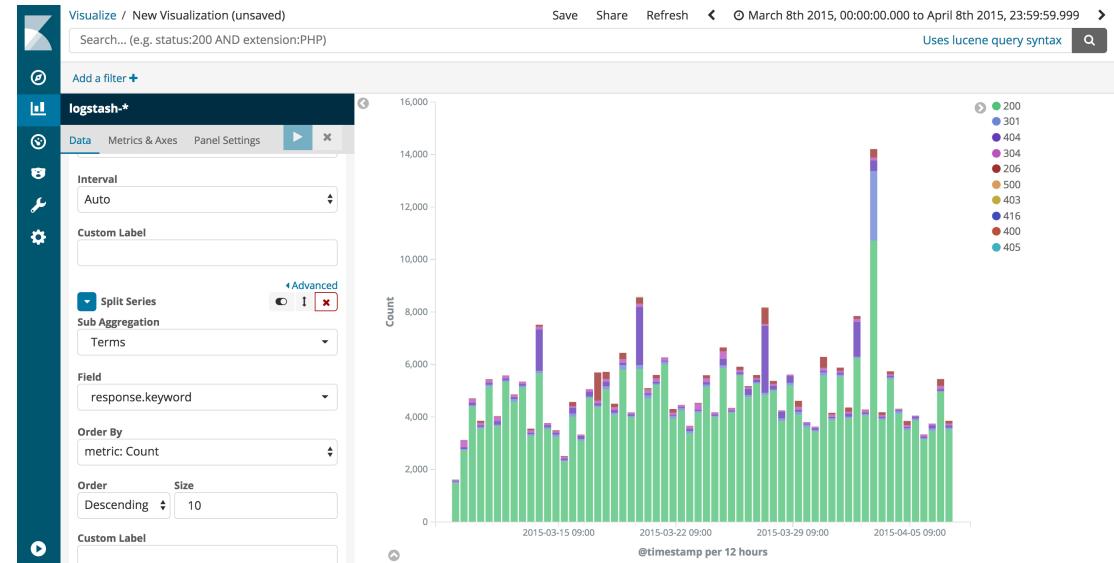
- Vertical Bar를 선택한 후 logstash-* 패턴을 선택합니다.

- X-Axis 를 클릭,
Aggregation은 Date Histogram,
Field는 @timestamp를
선택합니다.

- Add sub-bucket 을 클릭하고
Split Series 를 선택합니다.

- Sub Aggregation 은 Terms,
Field는 response.keyword
를 선택합니다.

- 상단의 Save 눌러
완성된 시각화 도구를
저장합니다.



Kibana Visualize – Table

- 접속 브라우저 순위 테이블을 만들어 보도록 하겠습니다.
 - Data Table을 선택한 후 logstash-* 인덱스 패턴을 선택합니다.
 - Split Rows를 선택하고, Aggregation 은 Terms, Field는 useragent.name.keyword 를 선택합니다.
 - 변경 내역을 반영하려면 옵션 옆의 버튼을 눌러야 반영이 됩니다.
 - 상단의 Save 눌러 완성된 시각화 도구를 저장합니다.

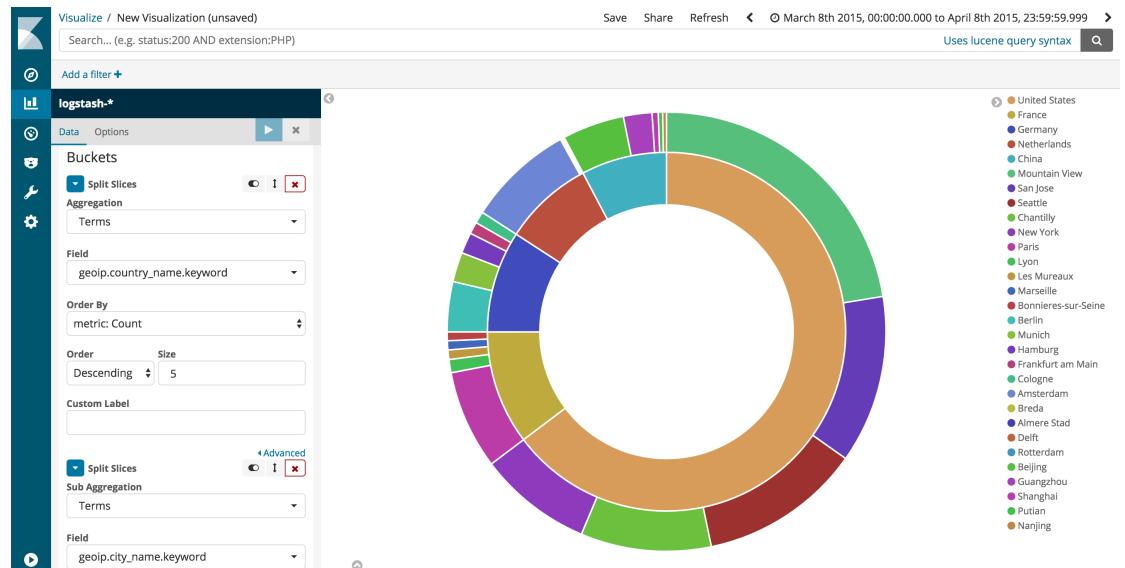
The screenshot shows the Kibana Visualize interface for creating a new visualization. The left sidebar has a 'Data' tab selected. The main area displays a table titled 'useragent.name.keyword: Descending' with the following data:

useragent.name.keyword	Count
Chrome	71,217
Firefox	58,566
Other	50,448
IE	14,476
Googlebot	12,857
Safari	8,379
Java	8,158
MJ12bot	4,930
Tiny Tiny RSS	4,530
AhrefsBot	4,437

Below the table are 'Export' options for 'Raw' and 'Formatted' data.

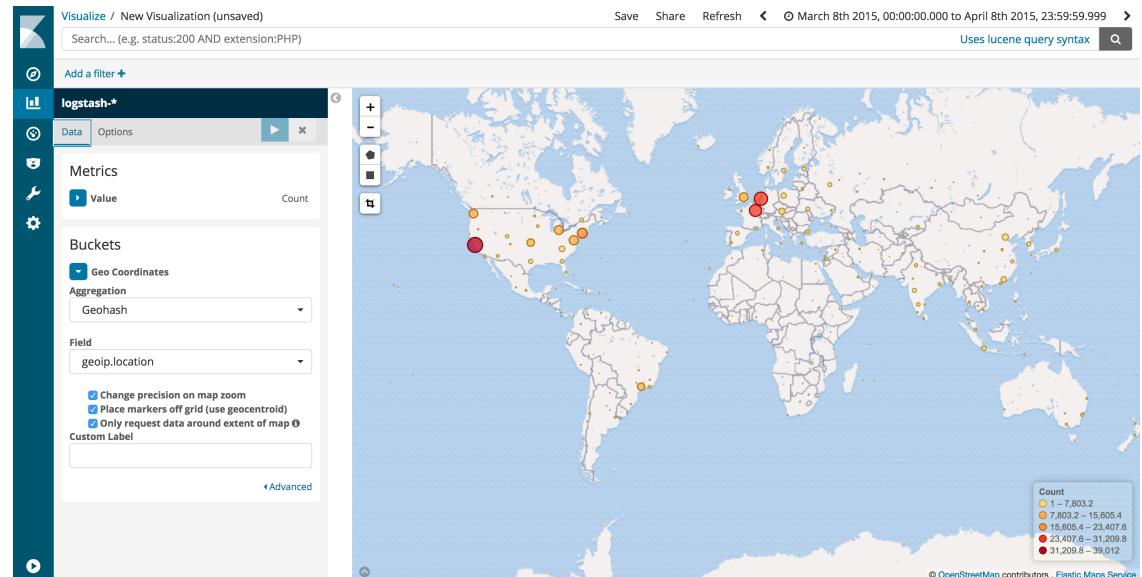
Kibana Visualize – Pie Chart

- 접속자의 국가, 도시별 파이 차트를 만들어 보도록 하겠습니다.
 - Pie 을 선택한 후 logstash-* 인덱스 패턴을 선택합니다.
 - Split Slices를 선택하고, Aggregation 은 Terms, Field는 geoip.country_name.keyword 를 선택합니다.
 - Add sub-buckets 를 눌러 Split Slices를 선택하고, Aggregation 은 Terms, Field는 geoip.city_name.keyword 를 선택합니다.
 - 상단의 Save 를 눌러 완성된 시각화 도구를 저장합니다.



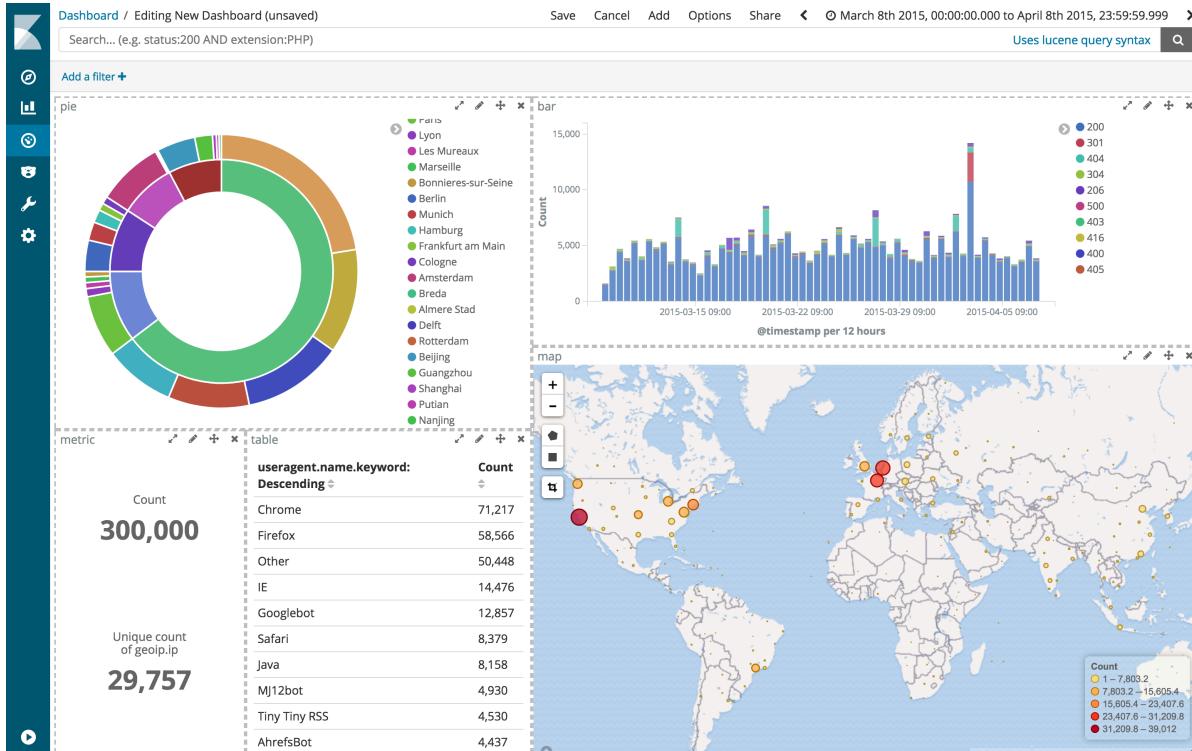
Kibana Visualize – Map

- 접속자 위치를 나타내는 지도를 만들어 보도록 하겠습니다.
 - Coordinate Map을 선택한 후 logstash-* 인덱스 패턴을 선택합니다.
 - Geo Coordinates를 클릭하고, Aggregation 은 GeoHash, Field는 geoip.location 을 선택합니다.
 - Option의 Map Type에서 마음에 드는 종류로 선택해 바꿔봅니다.
 - 상단의 Save 눌러 완성된 시각화 도구를 저장합니다.



Kibana Dashboard

- 이제 Dashboard에서 상단의 add를 눌러 지금까지 만든 시각화 도구들을 배치하고 저장합니다.





감사합니다.

<https://www.elastic.co/kr/>

<https://www.facebook.com/groups/elasticsearch.kr>

