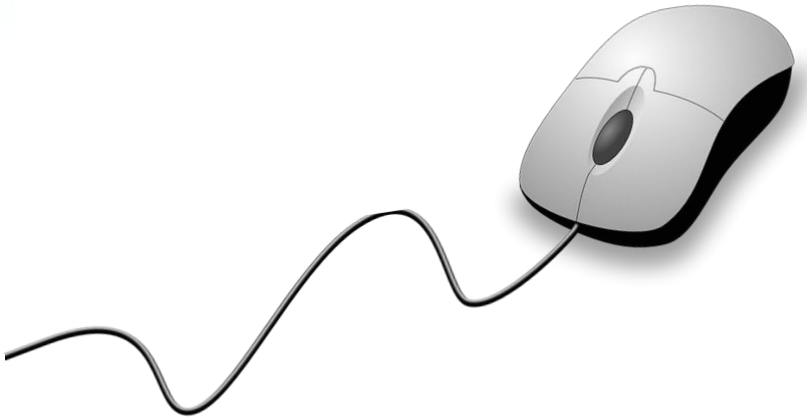
공개SW 솔루션 설치 & 활용 가이드

시스템SW > 데이터관리

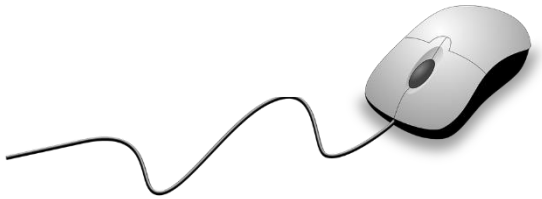


elasticsearch 제대로 배워보자

How to Use Open Source Software

Open Source Software Installation & Application Guide

nipa 공개SW역량프라자



CONTENTS

1. 개요

elasticsearch

소개	• 2012년 샤이 배논(Shay Banon)에 의해 처음 개발된 아파치 루씬(Lucene) 기반의 검색엔진 • 시각화 도구인 Kibana, 수집 도구인 Logstash, Beats 등과 함께 Elastic Stack 으로 구성 • 2017년 7월 기준 1억3천만의 누적 다운로드를 기록중		
주요기능	• 검색 : 커머스 사이트의 상품 검색 또는 위키피디아 등의 문서 검색. • 분석 : 의료, 보안 등의 특수 데이터 분석. • 로그분석 : 웹 로그, 머신 로그 등의 시계열 기반의 데이터 분석.		
대분류	• 시스템 SW	**소분류**	• 데이터관리
라이선스형태	• Apache 2.0	**사전설치 솔루션**	• Java
운영제제	• Linux / MacOS / Windows	**버전**	• 5.5
특징	• 기본적으로 분산 시스템으로 구성되며 스케일 아웃과 데이터 유실을 대비한 고가용성 • JSON 과 REST API 를 이용하여 다양한 클라이언트와 연동되며 유연한 데이터 모델 지원 • 전문(Full Text) 검색을 지원하며 다양한 시계열 데이터와 수치 데이터의 집계 및 연산 가능 • 실시간 데이터 검색, 분석 및 다양한 쿼리 문법 지원		
보안취약점	• 취약점 ID : CVE-2015-4165 • 심각도 : 7.5 HIGH(V3) • 취약점 설명 : Elasticsearch 버전 1.0.0 - 1.5.2는 시스템의 다른 응용 프로그램에 대한 조작된 공격에 취약 • 대응방안 : 1.6.0 이상으로 업그레이드 • 참고 경로 : https://www.securityfocus.com/archive/1/535727/100/0/threaded		
개발회사/커뮤니티	• Elastic		
공식 홈페이지	• https://www.elastic.co		

2. 기능요약

Elasticsearch 는 오픈소스이며 전문검색 기반의 고가용성 실시간 분석시스템이다.

주요기능	지원여부
오픈소스	Apache 2.0
분산 시스템	샤드(Shard) 기반의 데이터 분산 저장
고가용성	복제본을 통한 데이터 유실 방지
문서 기반	json 도큐먼트 기반
RESTFul	http 프로토콜을 통한 REST API 지원
전문(Full Text) 검색	문서 전체를 Term 기반으로 색인
실시간	배치 기반의 사이클 분석이 아닌 데이터를 실시간으로 저장 / 쿼리 가능

3. 실행환경

기본적으로 Java(Oracle, OpenJDK)가 설치된 환경에서는 대부분 실행 가능하다.

https://www.elastic.co/support/matrix 에서 지원 환경의 확인이 가능하다.

	CentOS/RHEL 6.x/7.x	Oracle Enterprise Linux 6/7 with RHEL Kernel only	Ubuntu 14.04	Ubuntu 16.04	SLES 11 SP4**/12	openSUSE Leap 42	Windows Server 2012/R2	Windows Server 2016	Debian 7	Debian 8	Debian 9	Solaris/ SmartOS	Amazon Linux*
Elasticsearch 2.2.x	✔	✔	✔	✖	✔	✔	✔	✖	✔	✖	✖	✖	✔
Elasticsearch 2.3.x	✔	✔	✔	✖	✔	✔	✔	✖	✔	✔	✖	✖	✔
Elasticsearch 2.4.x	✔	✔	✔	✖	✔	✔	✔	✖	✔	✔	✖	✖	✔
Elasticsearch 5.0.x	✔	✔	✔	✔	✔	✔	✔	✖	✔	✔	✖	✖	✔
Elasticsearch 5.1.x	✔	✔	✔	✔	✔	✔	✔	✖	✔	✔	✖	✖	✔
Elasticsearch 5.2.x	✔	✔	✔	✔	✔	✔	✔	✖	✔	✔	✖	✖	✔
Elasticsearch 5.3.x	✔	✔	✔	✔	✔	✔	✔	✖	✔	✔	✖	✖	✔
Elasticsearch 5.4.x	✔	✔	✔	✔	✔	✔	✔	✖	✔	✔	✖	✖	✔
Elasticsearch 5.5.x	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✔	✖	✔

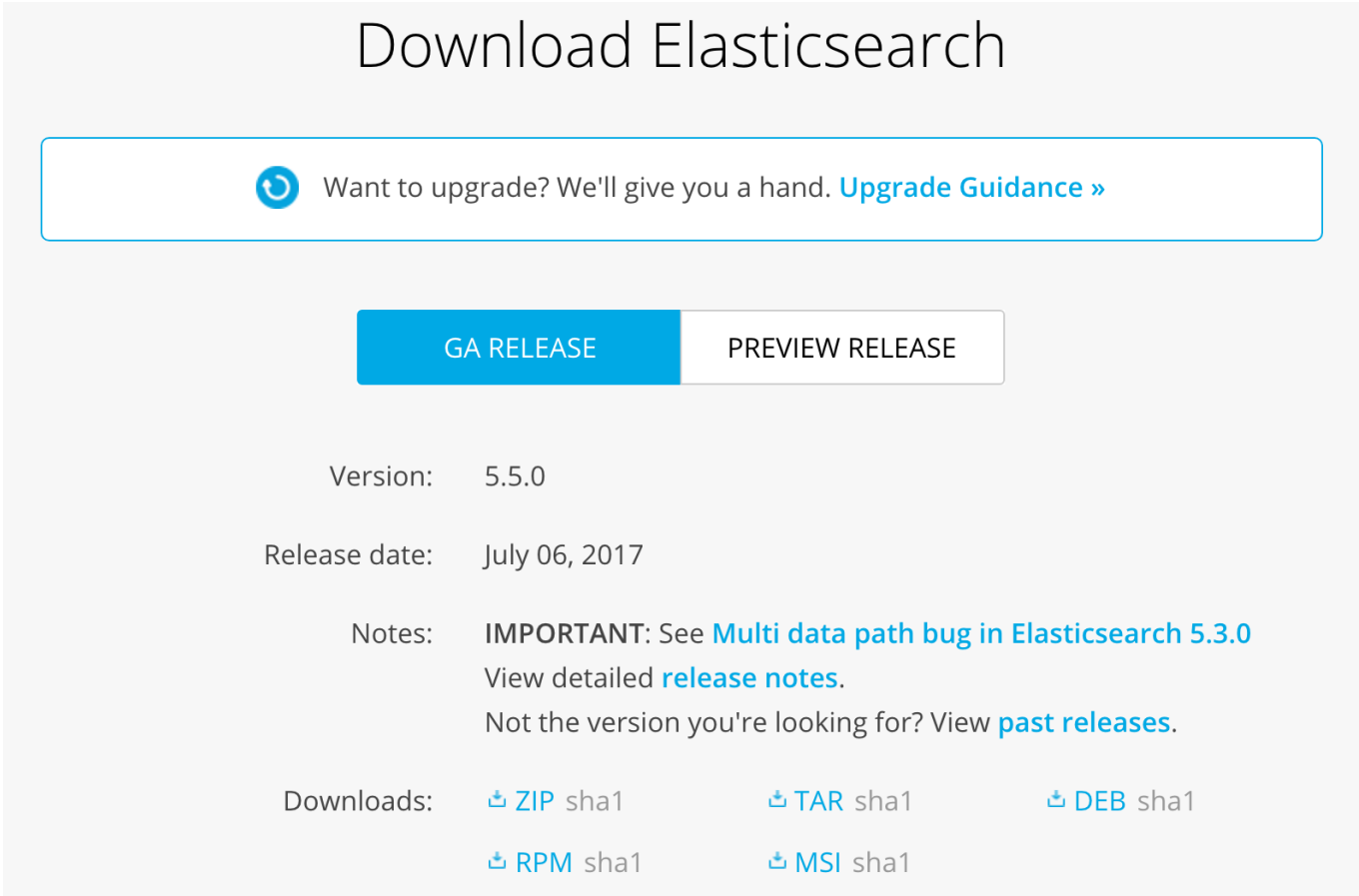
4. 설치 및 실행

세부 목차

4. 설치 및 실행

4.1 압축/설치 파일 다운로드

- https://www.elastic.co/kr/downloads/elasticsearch 페이지에서 운영체제에 맞는 설치 파일을 다운로드 한다. 이 자료에서는 5.5.0 버전을 기준으로 설명한다.



4. 설치 및 실행

4.2 일반 설치 및 실행 - MacOS

- tar 또는 zip 파일을 받아 원하는 경로에 압축을 푼다.

```
$ tar xfz elasticsearch-5.5.0.tar.gz
$ unzip elasticsearch-5.5.0.zip
```

- 압축을 푼 Elasticsearch 홈 디렉토리로 이동한다.

```
$ cd elasticsearch-5.5.0
```

- 홈 디렉토리의 bin 아래에 있는 elasticsearch 파일을 실행한다.

```
$ bin/elasticsearch
[2017-07-24T17:19:53,016][INFO ][o.e.n.Node               ] [] initializing ...
[2017-07-24T17:19:53,103][INFO ][o.e.e.NodeEnvironment    ] [rK2WX9V] heap size [1.9gb],
[2017-07-24T17:19:53,107][INFO ][o.e.n.Node               ] node name [rK2WX9V]
[2017-07-24T17:19:54,261][INFO ][o.e.p.PluginsService     ] [rK2WX9V] loaded module [percolator]
[2017-07-24T17:19:54,261][INFO ][o.e.p.PluginsService     ] [rK2WX9V] loaded module [reindex]
...
[2017-07-24T17:19:56,258][INFO ][o.e.n.Node               ] initialized
[2017-07-24T17:19:56,258][INFO ][o.e.n.Node               ] [rK2WX9V] starting ...
[2017-07-24T17:20:01,482][INFO ][o.e.t.TransportService   ] [rK2WX9V] publish_address {127.0.0.1:9300},
[2017-07-24T17:20:04,564][INFO ][o.e.n.Node               ] [rK2WX9V] started
```

- 백그라운드로 실행하고자 하는 경우에는 –d 옵션을 추가한다.

```
$ bin/elasticsearch -d
```

4. 설치 및 실행

4.3 서비스로 설치 - Redhat Linux

- rpm 파일을 내려받아 설치한다.

```
$ wget https://artifacts.elastic.co/downloads/elasticsearch/elasticsearch-5.5.0.rpm
$ sha1sum elasticsearch-5.5.0.rpm
$ sudo rpm --install elasticsearch-5.5.0.rpm
```

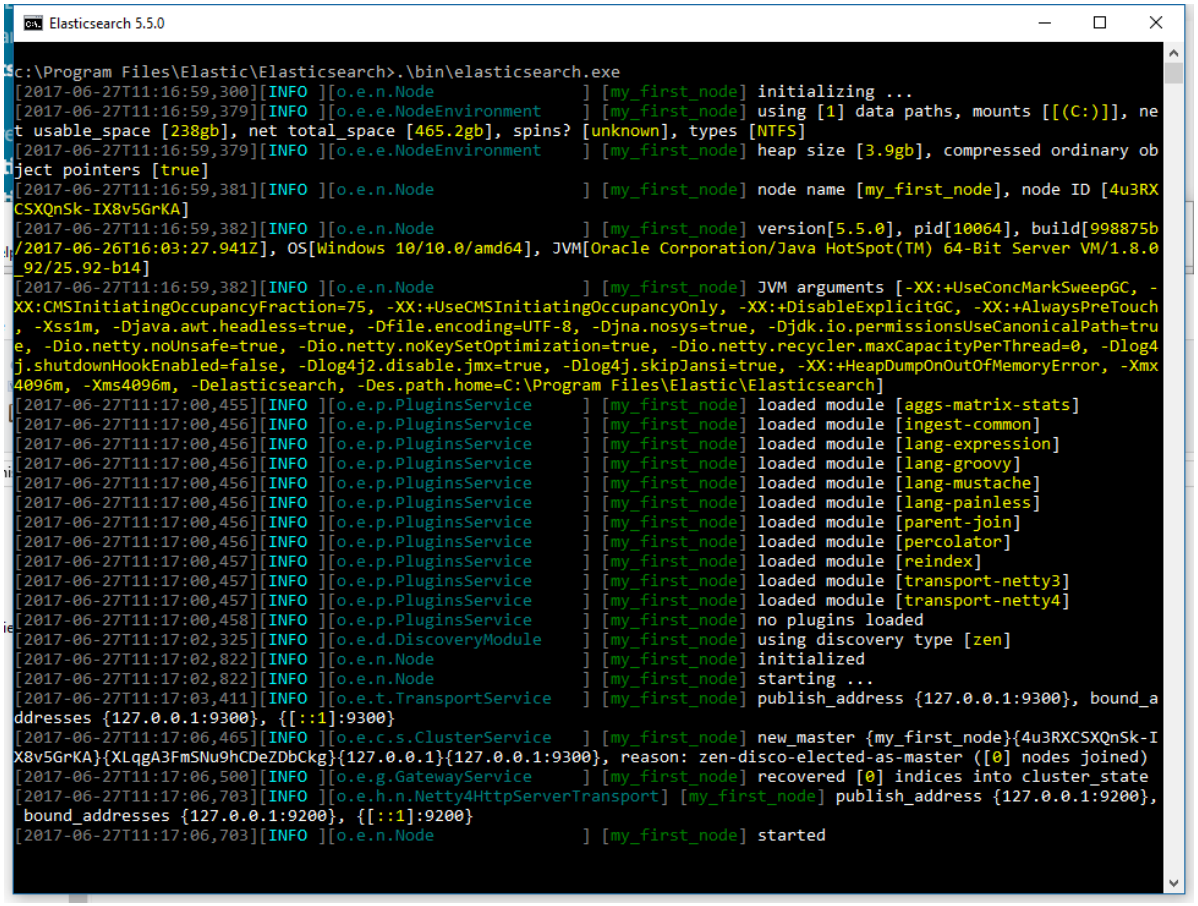
- service 명령을 이용해서 서비스로 Elasticsearch 를 시작, 종료 한다.

```
$ sudo -i service elasticsearch start
$ sudo -i service elasticsearch stop
```

4. 설치 및 실행

4.4 Windows - 커맨드 도구 이용

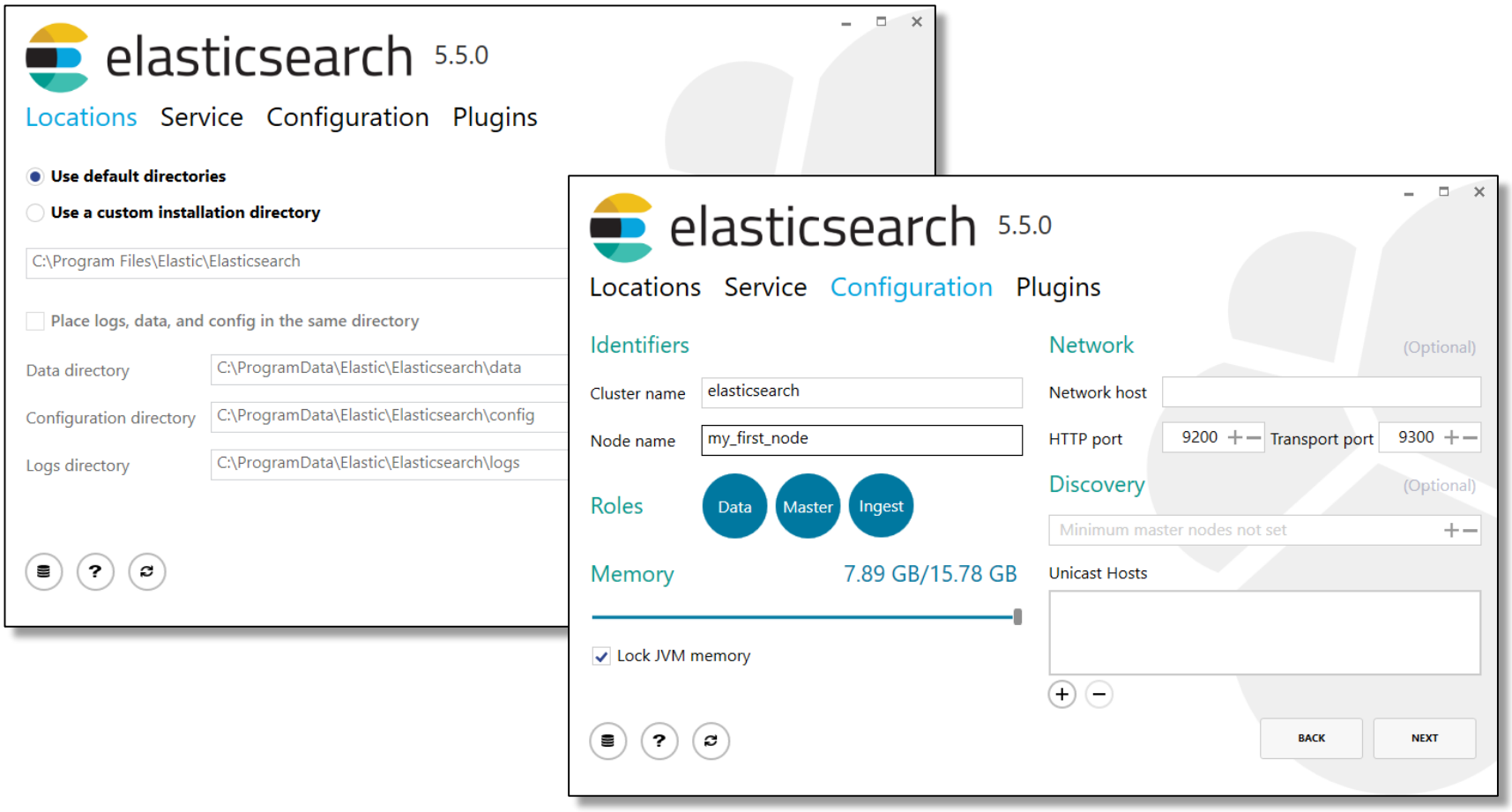
- zip 파일을 받아 원하는 경로에 압축을 푼다.
- 커맨드 도구로 압축을 푼 경로의 ₩bin 디렉토리 아래의 elasticsearch.exe 를 실행시킨다



4. 설치 및 실행

4.5 Windows - MSI 설치(1/2)

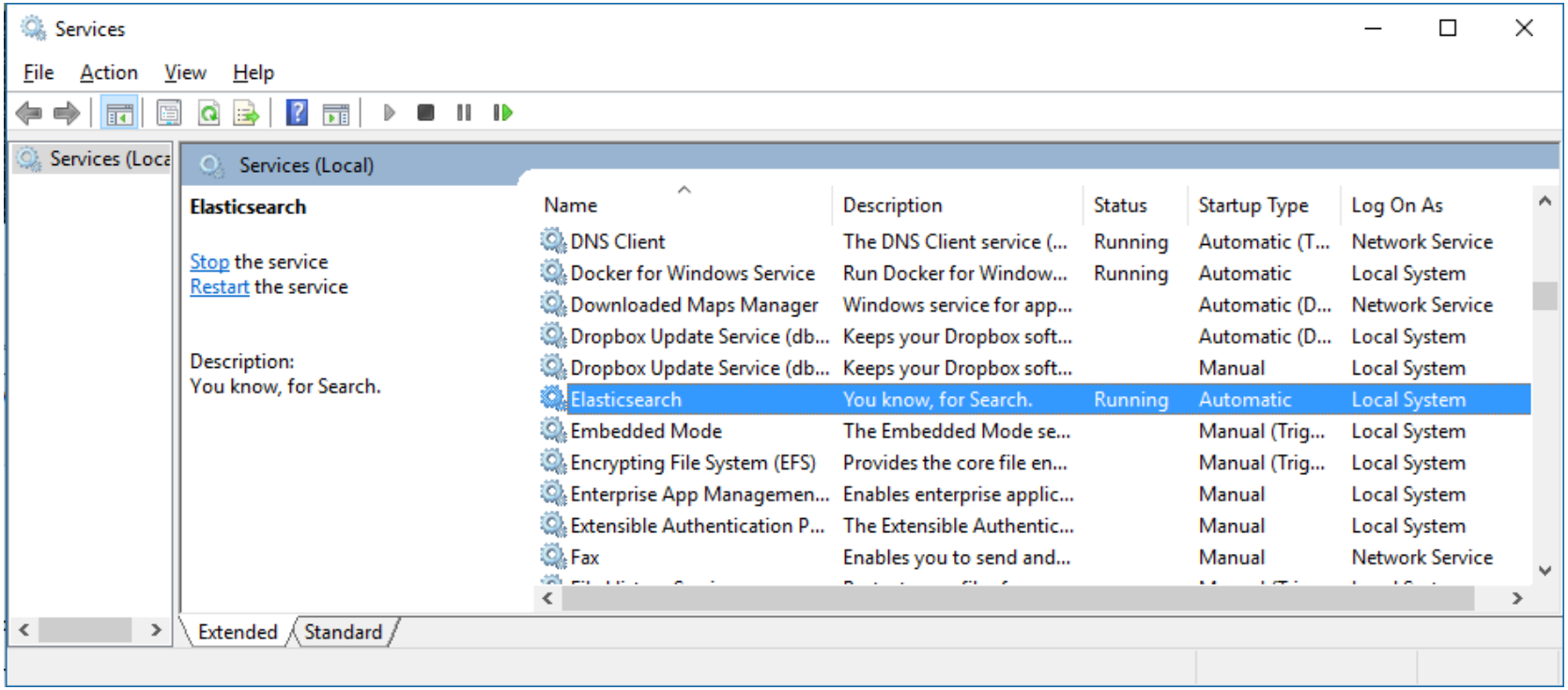
- msi 패키지 프로그램을 다운로드 받아 실행하여 서비스 프로그램으로 설치한다.



4. 설치 및 실행

4.5 Windows - MSI 설치(2/2)

- 프로그램 서비스 목록에서 Elasticsearch 를 찾아 실행, 중지를 시킬 수 있다.



- 세부 설정은 Windows install 메뉴얼 페이지를 참고한다.

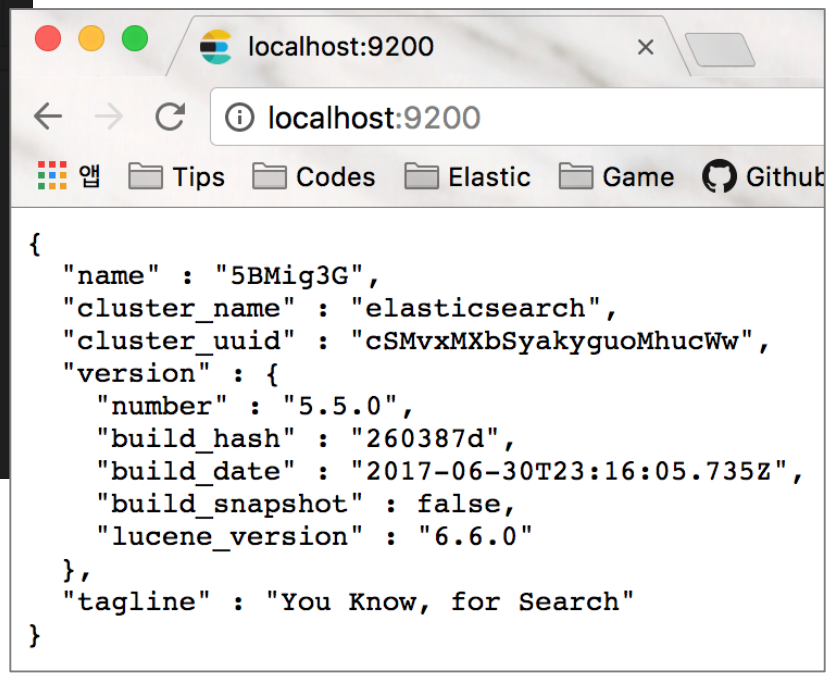
  https://www.elastic.co/guide/en/elasticsearch/reference/current/windows.html

4. 설치 및 실행

4.6 실행 확인

- 유닉스 계열에서 curl 명령을 사용해서 localhost 의 9200번 포트를 확인한다.
- 또는 웹 브라우저에서 http://localhost:9200 로 접속해서도 확인이 가능하다.

```
bash-3.2$ curl localhost:9200
{
  "name" : "5BMig3G",
  "cluster_name" : "elasticsearch",
  "cluster_uuid" : "cSMvxMXbSyakyguoMhucWw",
  "version" : {
    "number" : "5.5.0",
    "build_hash" : "260387d",
    "build_date" : "2017-06-30T23:16:05.735Z",
    "build_snapshot" : false,
    "lucene_version" : "6.6.0"
  },
  "tagline" : "You Know, for Search"
}
```



5. 기능소개

세부 목차

5. 기능소개

5.1 REST API(2/1)

- Elasticsearch 는 REST API 를 이용한 http 통신을 통해 데이터를 처리한다.
- 데이터는 인덱스와 타입 이라고 하는 논리적 영역에 저장되며 JSON 도큐먼트 형식으로 저장된다.
- JSON 도큐먼트의 개별 요소 값을 필드 라고 한다.
- 각 도큐먼트의 접근 URI 는 호스트:포트/인덱스/타입/도큐먼트ID 로 이루어진다.
- http 메소드인 PUT / POST / GET / DELETE 등을 이용하여 문서를 입력/조회/삭제를 한다.

HTTP	CRUD	SQL
GET	Read	Select
PUT	Update	Update
POST	Create	Insert
DELETE	Delete	Delete

관계 DB	Elasticsearch
테이블(Table)	인덱스 (index) / 타입(Type)
열(Row)	도큐먼트 (Document)
행(Column)	필드(Field)
스키마(Schema)	매핑(Mapping)

5. 기능소개

5.1 REST API(2/2)

- 도큐먼트 입력

```
$ curl -XPUT 'http://localhost:9200/books/book/1' -d '
 {
   "title" : "Elasticsearch Guide",
   "author" : "Kim",
   "date" : "2014-05-01",
   "pages" : 250
 }'
{"_index":"books","_type":"book","_id":"1","_version":1,"result":"created","_shards":{"total":2,"successful":1,"failed":0},"created":
true}
```

- 도큐먼트 조회

```
$ curl -XGET 'http://localhost:9200/books/book/1'
{"_index":"books","_type":"book","_id":"1","_version":1,"found":true,"_source":
{  "title" : "Elasticsearch Guide",
  "author" : "Kim",
  "date" : "2014-05-01",
  "pages" : 250
}}
```

- 도큐먼트 삭제

```
$ curl -XDELETE 'http://localhost:9200/books/book/1'
{"found":true,"_index":"books","_type":"book","_id":"1","_version":2,"result":"deleted","_shards":{"total":2,"successful":1,"failed
":0}}
```

5.2.1 클러스터 (Cluster)

- 엘라스틱서치 시스템의 가장 큰 단위
- 하나의 클러스터는 다수의 노드로 구성
- 하나의 클러스터를 다수의 서버로 바인딩 해서 운영, 또는 역으로 하나의 서버에서 다수의 클러스터 운용 가능
- Elasticsearch 설치 경로의 config 디렉토리 아래에 있는 elasticsearch.yml 파일, 또는 실행 시 -E 커맨드 옵션으로 설정 가능

config/elasticsearch.yml

```
cluster.name: elasticsearch
```

```
$ bin/elasticsearch -E cluster.name=elasticsearch
```

5.5.2 노드 (Node) (1/3)

- 엘라스틱서치를 구성하는 하나의 단위 프로세스
- 다수의 샤드로 구성됨
- 같은 클러스터명을 가진 노드들은 자동으로 바인딩 됨
- Elasticsearch 설치 경로의 config 디렉토리 아래에 있는 elasticsearch.yml 파일, 또는 실행 시 -E 커맨드 옵션으로 설정 가능

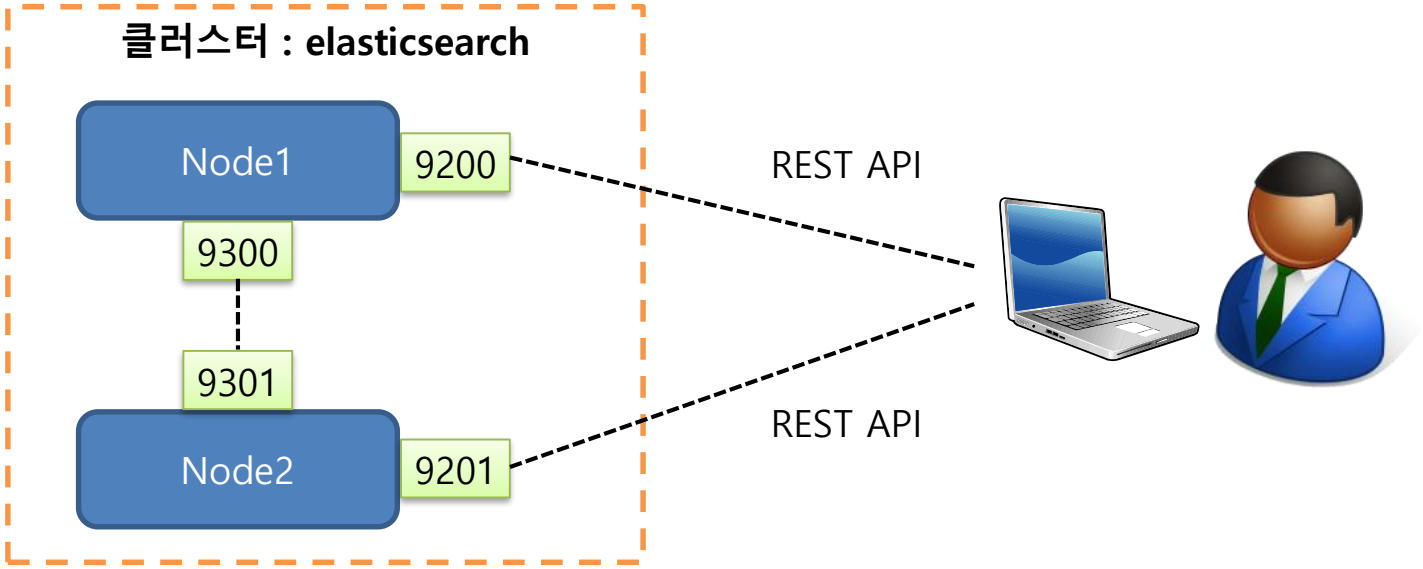
config/elasticsearch.yml

```
node.name: "Node1"
```

```
$ bin/elasticsearch -E node.name=Node1
```

5.2.2 노드 (Node) (2/3)

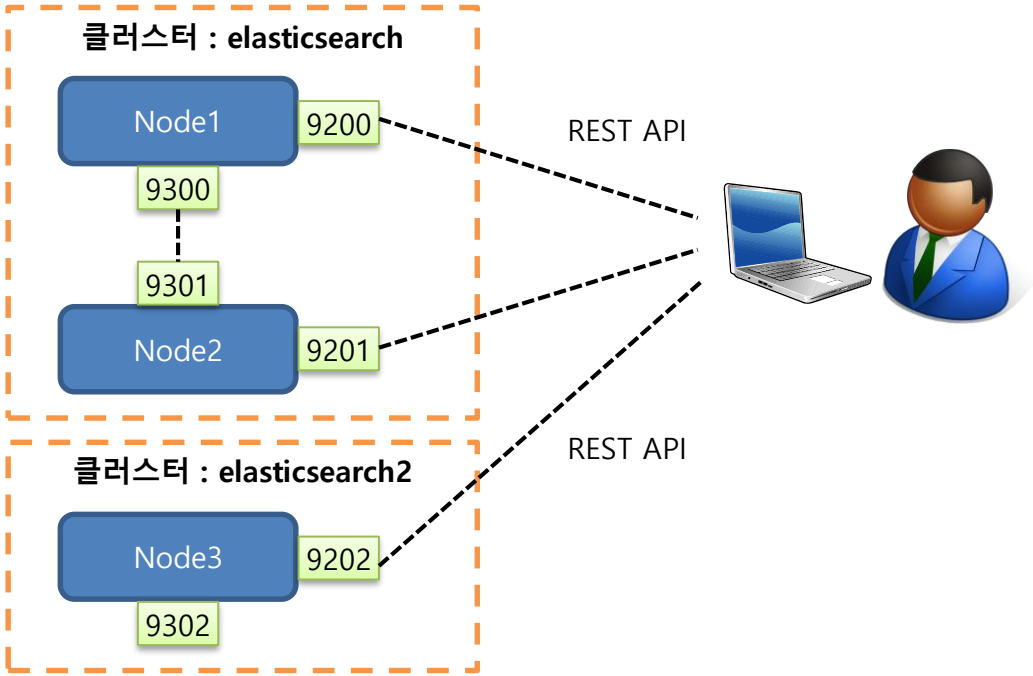
- http 통신 포트 : 9200~ 부터 차례대로 증가
- 노드 간 데이터 교환 포트 : 9300~ 부터 차례대로 증가



5.2.2 노드 (Node) (3/3)

- 같은 클러스터명을 가진 노드들은 자동으로 한 클러스터 안에서 바인딩 됨
- 클러스터명이 다르면 같은 서버 또는 네트워크 내에서도 다른 클러스터로 구성됨

```
$ bin/elasticsearch -E cluster.name=elasticsearch -E node.name=Node1
$ bin/elasticsearch -E cluster.name=elasticsearch -E node.name=Node2
$ bin/elasticsearch -E cluster.name=elasticsearch2 -E node.name=Node3
```



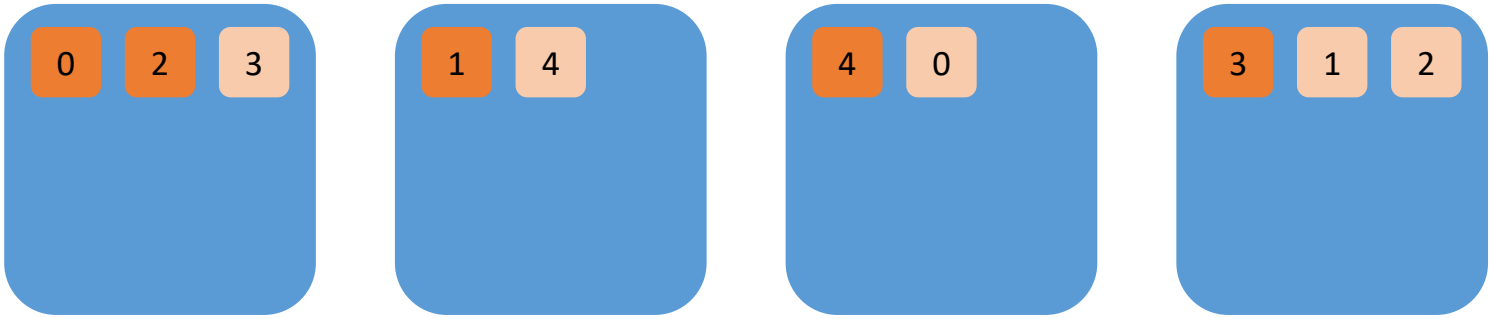
5.5.3 샤드 (Shard) & 레플리카 (Replica) (1/2)

- 샤드 : 데이터 검색 단위 루씬 인스턴스
- 레플리카 : 샤드의 복사본
- 인덱스 별로 설정. 기본값은 샤드-5, 레플리카-1.
- REST API의 PUT 메소드를 이용해서 인덱스 안의 settings 값으로 설정

```
$ curl -XPUT localhost:9200/books -d '
{
  "settings" : {
    "number_of_shards" : 5,
    "number_of_replicas" : 1
  }
}'
```

5.5.3 샤드 (Shard) & 레플리카 (Replica) (2/2)

- 각 노드 별로 샤드가 분배되어 저장됨
- 동일한 샤드와 레플리카는 항상 서로 다른 노드에 저장됨
- 일부 노드가 중지되더라도 샤드와 레플리카 중 최소 1개가 살아 있으면 클러스터는 정상적으로 동작



5.3.1 URI 검색

- 검색은 _search API 사용. 인덱스 단위 및 logs-* 와 같은 멀티테넌시 검색 가능.
- 호출 URI 에 q= 파라메터로 검색 쿼리 삽입.

```
$ curl 'http://localhost:9200/books/_search?q=william'
{ ... 중략 ...
},
  "hits" : {
    "total" : 3,
    "max_score" : 1.0,
    "hits" : [ {
... 중략 ...
       "_source":
{ "title": "Romeo and Juliet", "author": "William Shakespeare", "category":"Tragedies", "written": "1562-12-01T20:40:00",
"pages" : 125 }
    },
... 중략 ...
```

5.3.2 Request Body 검색

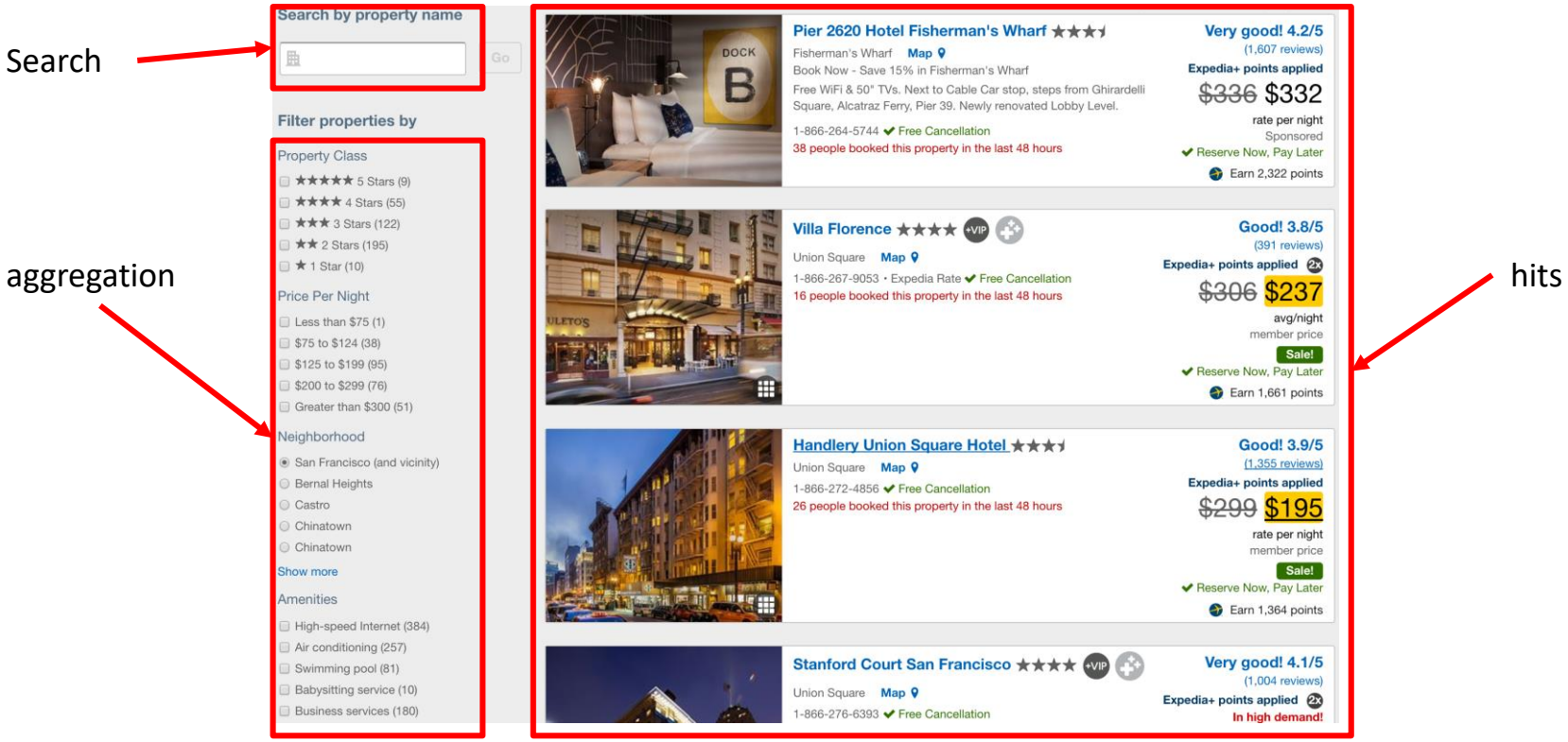
- 검색 쿼리를 JSON 형태의 데이터로 전송

```
$ curl 'http://localhost:9200/books/_search?pretty=true' -d '
{
  "query" : {
    "match" : {
      "author" : "william"
    }
  }
}'
{ ... 중략 ...
},
  "hits" : {
    "total" : 3,
    "max_score" : 1.0,
    "hits" : [ {
... 중략 ...
      "_source":
{ "title": "Romeo and Juliet", "author": "William Shakespeare", "category":"Tragedies", "written": "1562-12-01T20:40:00",
"pages" : 125 }
    },
... 중략 ...
```

5. 기능소개

5.4 집계 - Aggregation(1/2)

- Elasticsearch 는 검색 결과 도큐먼트 내용 외에도 각 필드의 값들을 집계 한 결과를 가져올 수 있다. 이 집계 기능을 aggregation 이라고 한다.
- 검색된 도큐먼트는 검색 결과의 hits[ ], 집계는 aggregation[ ] 배열 안에 들어간다.



- Aggregation 은 _search 로 검색 시 query 문과 함께 사용한다.

```
$ curl 'localhost:9200/_search' -d '
{
  "query" : {
    // query
  },
  "aggregations" : {    // or “aggs”
    “aggs_name” : {
      // a set of aggregation
    }
  }
}'
```

6. 활용예제

세부 목차

6. 활용예제

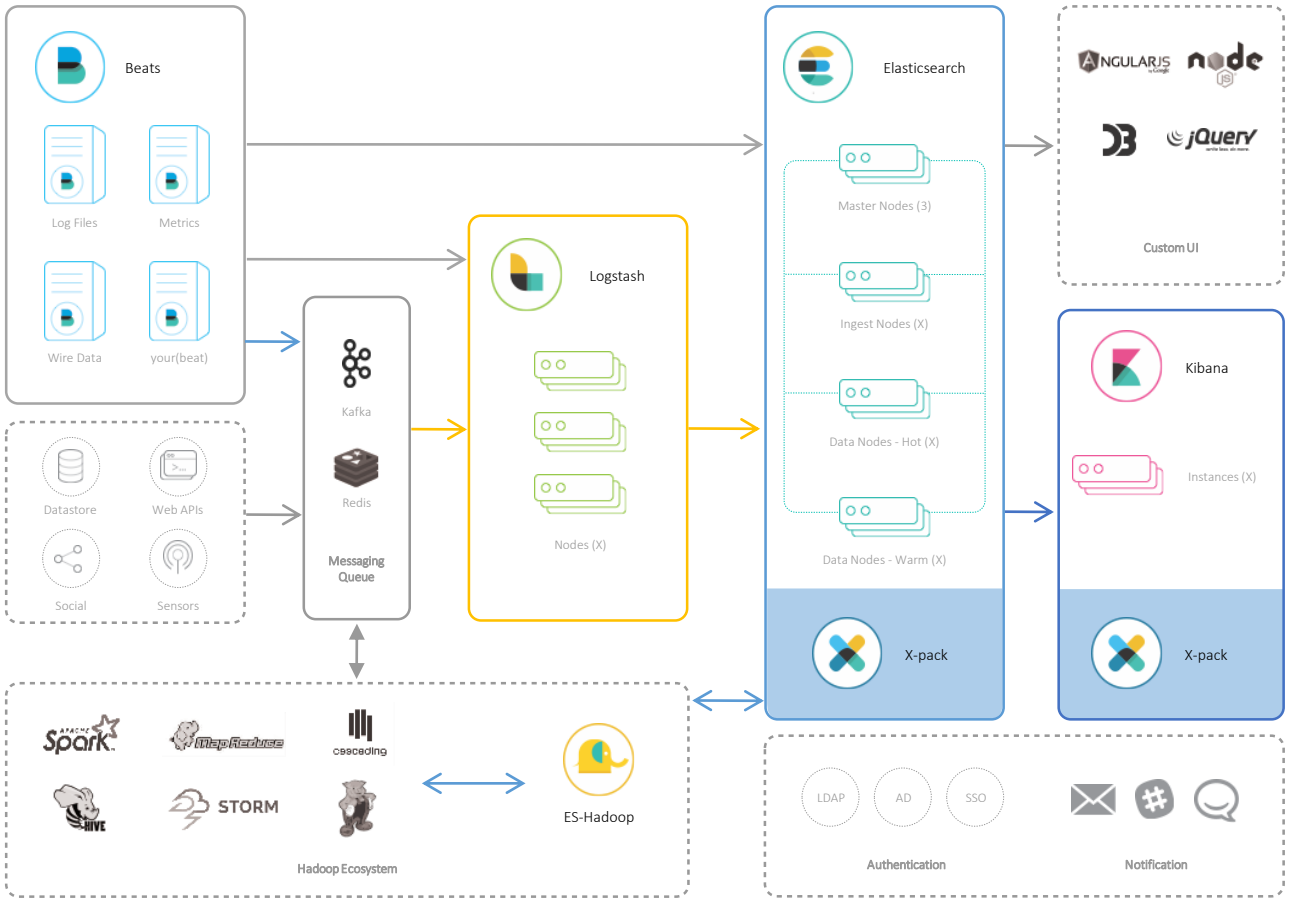
6.1 Elastic Stack 아키텍처(1/2)

- 모든 데이터는 Elasticsearch 안에 색인되어 저장되고, 조회, 집계된다.
- REST API를 지원하는 애플리케이션이라면 모두 데이터 수집이 가능하지만, Elastic 에서는 데이터 수집기인 Logstash 와 Beats 를 지원하고 있다.
  - Logstash 는 다양한 입/출력 파이프라인 및 데이터 변조, 필터링을 지원한다.
  - Beats 는 수집한 데이터를 Elasticsearch 또는 Logstash 로 전송하는 기능만 있으며 대신 가볍고 빠르다.
- 시각화 툴인 Kibana 를 이용해서 Elasticsearch 에 저장된 데이터를 다양한 도표로 시각화가 가능하다.
- Elasticsearch, Logstash, Beats, Kibana 를 묶어 Elastic Stack 이라고 부르며 이 스택에 해당되는 제품들은 모두 오픈소스 (Apache 2.0) 이다.
- 추가로 Elastic 사 에서는 X-Pack 이라는 상용 플러그인을 배포중이며 X-Pack은 보안, 알람, 모니터링, 그래프, 머신러닝 등의 추가 기능들을 제공한다.
- ES-Hadoop 이라는 제품을 통해 하둡 시스템으로부터 데이터 수집이 가능하다

6. 활용예제

6.1 Elastic Stack 아키텍처(2/2)

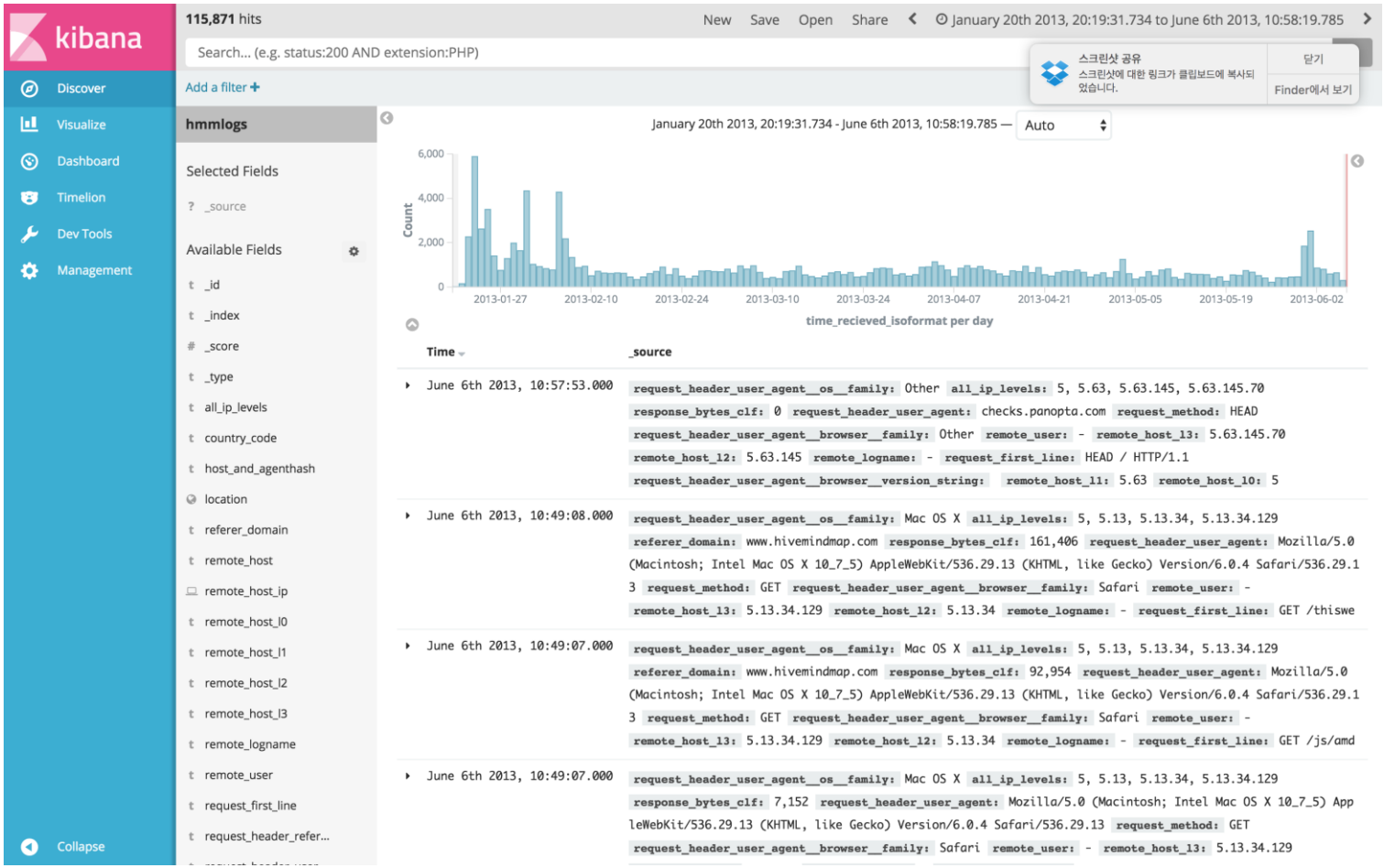
- Beats 및 다양한 소스로부터 수집된 데이터가 Logstash를 거쳐 Elasticsearch 클러스터에 저장되고 Kibana 및 다른 애플리케이션들을 통해 검색 및 조회가 된다.



6. 활용예제

6.2 Kibana(1/3)

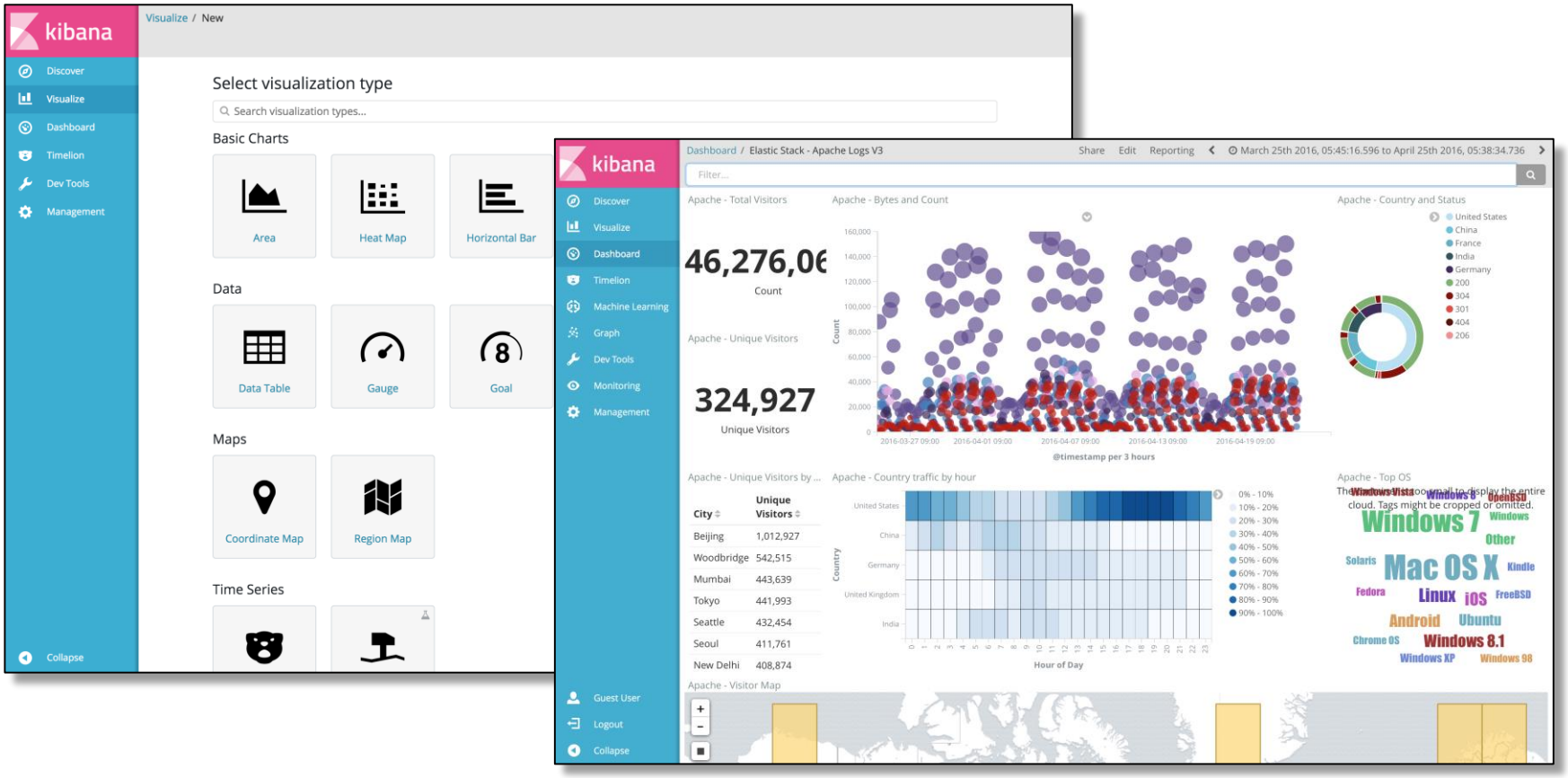
- Kibana를 이용해 Elasticsearch 안에 있는 데이터를 쉽게 시각화가 가능하다.



6. 활용예제

6.2 Kibana(2/3)

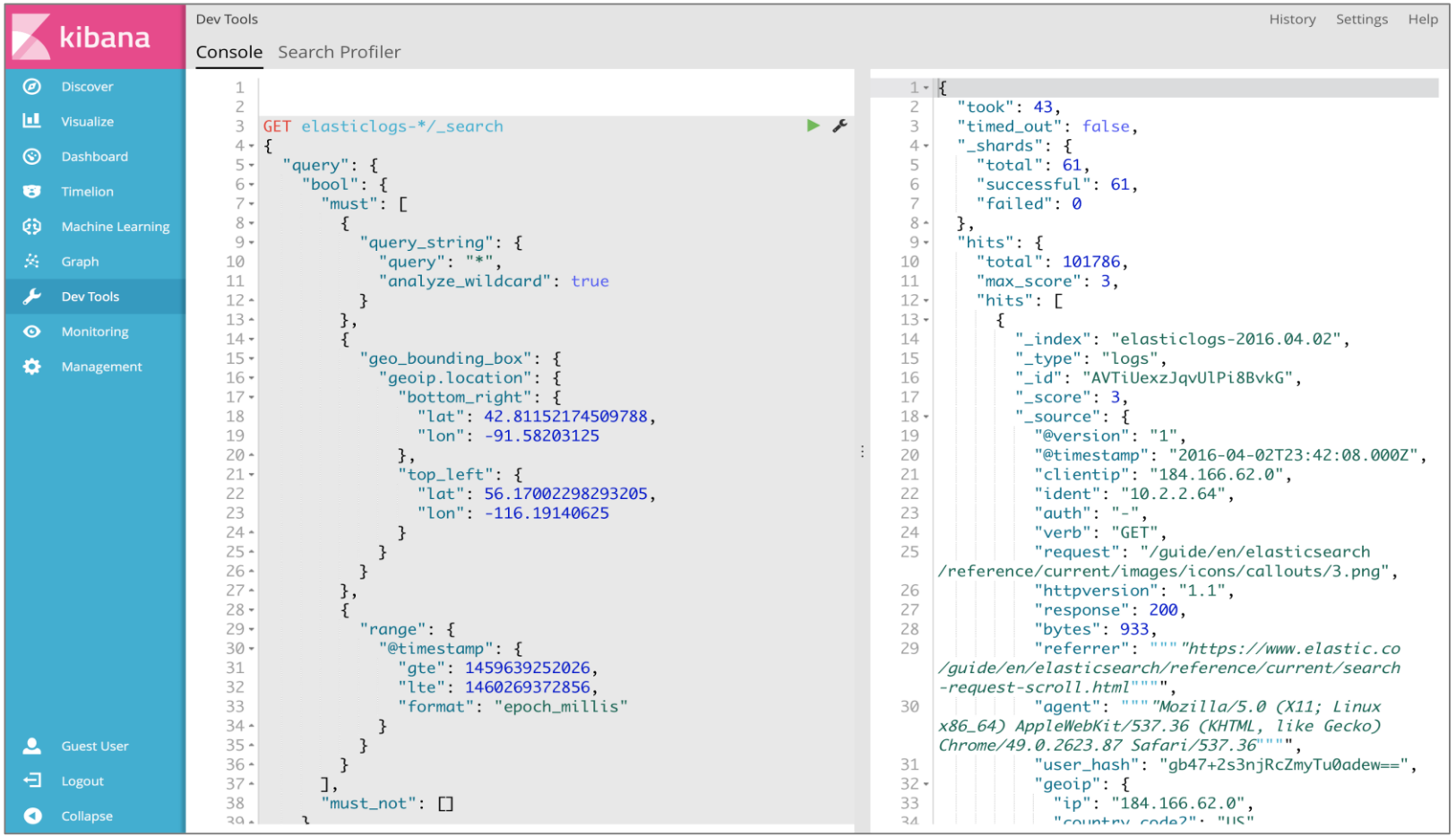
- Visualize 메뉴에서 시각화 모듈들을 먼저 만들고 Dashboard 메뉴에서 모듈들을 구성하여 대시보드를 완성한다.



6. 활용예제

6.2 Kibana(3/3)

- Kibana에 있는 Dev Tools 메뉴에서 Elasticsearch 쿼리의 자동완성 등을 지원하여 편리하게 REST API 사용이 가능하다.



7. FAQ

Q Elasticsearch는 완전 무료인가요?

A Elasticsearch 를 비롯한 Kibana, Logstash, Beats 등 모든 Elastic Stack 제품은 Apache 2.0 라이센스를 따르는 오픈소스이며 기능과 사용 범위에 아무런 제한이 없고 비용이 들지 않습니다.

Q Elasticsearch는 주로 어디에 쓰이나요?

A Elasticsearch는 검색엔진이지만 대부분 형태의 데이터를 모두 처리할 수 있어 로그분석, 위치정보 분석 등 다양한 용도로 활용됩니다. 금융, 보안, 게임, 쇼핑, 의료 등 거의 모든 산업 분야에서 사용되고 있으며, 자세한 사례는 https://www.elastic.co/use-cases 에서 확인이 가능합니다.

7. FAQ

Q **Elasticsearch 에서 한글로도 검색이 가능한가요?**

A 한글을 활용하기 위해서는 한글 형태소 분석기를 별도로 설치해야 합니다. 아리랑, 은전한닢 등의 한글 형태소 분석기의 사용이 가능하며 해당 홈페이지 또는 커뮤니티를 통해 제공받아 설치해야 합니다.

Q **데이터 조인이 가능한가요?**

A Elasticsearch는 데이터를 역색인(Inverted Index) 구조로 저장하기 때문에 기본적으로 조인 기능은 지원하지 않습니다. 하지만 문서를 Nested 구조로 저장하거나 Parent / Child 구조를 이용해서 검색 시 다른 도큐먼트를 참조해서 검색할 수 있습니다.

8. 용어정리

elasticsearch

용어	설 명
REST API	Representational safe transfer API 리소스, URI, http 메소드로 이루어진 시스템 아키텍트
JSON	JavaScript Object Notation. 사람과 기계가 모두 이해하기 용이한 경량 데이터 교환 방식
Query DSL	Query Domain Specific Language Elasticsearch의 검색에 사용되는 질의 문법

Open Source Software Installation & Application Guide

nipa 공개SW역량프라자