OKdevTV

ELK

- elatic stack 5
- Elasticsearch + Logstash + Kibana
- Elasticsearch는 Apache의 Lucene을 바탕으로 개발한 실시간 분산 검색 엔진이며,
- Logstash는 각종 로그를 가져와 JSON형태로 만들어 Elasticsearch로 전송하고,
- Kibana는 Elasticsearch에 저장된 Data를 사용자에게 Dashboard 형태로 보여주는 솔루션이다.

- http://elastic.co 사이트 오픈소스 제품

장점

- Google Analytics(GA)의 데이터로 사이트 접속 통계를 구할 경우 원하는 대로 데이터를 획득하기 어렵다.
- 자체 서버의 모든 로그를 100% 수집할 수 있기 때문에 데이터에 대한 신뢰성이 높다.
- 파라미터 값별로 통계를 볼 수 있기 때문에 정확한 데이터 분석이 가능하다.
- 검색엔진(lucene)이 포함되어 있어, 빠르게 데이터를 검색할 수 있다.
- 모두 오픈소스이며 자유롭게 사용이 가능하다.

사전 준비

- 로그수집 서버(AWS 추천)
  - aws 접속 key가 있는 경우
  - 윈도우에서 git bash 추천(http://git-scm.com). putty 접속보다 쉬움
- 리눅스 서버 CentOS 또는 Ubuntu
- Java 1.7 이상(esp. logstash는 1.8이상 필요)
- ubuntu 에서는 jdk 설치 필요

```
sudo add-apt-repository ppa:openjdk-r/ppa

sudo apt-get update
sudo apt-get install openjdk-8-jdk -y
```

nginx 설치(샘플용)

- nginx 설치

AWS 포트 설정

- EC2 Security Groups
- 외부 접근 포트 추가(inbound)
  - http(80)
  - elasticsearch(9200)
  - kibana(5601)

설치

- Elasticsearch
- Kibana

- Logstash (FluentD로 대치 가능)

- 버전을 맞춰서 작업하는 것이 좋지만, 최신 버전으로 작업해도 동작함(2016/04/03 현재)

- Elasticsearch와 Kibana는 권장 버전을 맞춰야 함
- 설치 위치 /opt/ 또는 ~/local/ 권장

Elasticsearch 설치

```
mkdir ~/local
cd ~/local
wget https://download.elastic.co/elasticsearch/release/org/elasticsearch/distribution/tar/elasticsearch/2.4.1/elasticsearch-2.4.1.tar.gz
tar xvfz elasticsearch-2.4.1.tar.gz
ln -s elasticsearch-2.4.1 elasticsearch
cd elasticsearch
vi config/elasticsearch.yml
  # `# network.host: 192.168.0.1`의 주석을 풀고 `network.host: 0.0.0.0`으로 변경
  # 모든 IP에서 접근 가능
bin/elasticsearch -d
```

```
# 데몬(백그라운드)로 실행. 옵션 -d를 빼면 터미널 접속해 있는 동안만 실행
```

- 실행 확인

```
curl -i http://localhost:9200/
```

Kibana 설치

```
cd ~/local
wget https://download.elastic.co/kibana/kibana/kibana-4.6.1-linux-x86_64.tar.gz
tar xvfz kibana-4.6.1-linux-x86_64.tar.gz
ln -s kibana-4.6.1-linux-x86_64 kibana
cd kibana
```

```
bin/kibana
# background run
nohup bin/kibana &
```

- 실행 확인 http://아이피:5601

Logstash 설치

```
cd ~/local
wget https://download.elastic.co/logstash/logstash/logstash-2.4.0.tar.gz
tar xvfz logstash-2.4.0.tar.gz
ln -s logstash-2.4.0 logstash
cd logstash
```

- conf 파일 생성

```
mkdir logconf
vi logconf/nginx.conf
```

logconf/nginx.conf

```
input {
    file {
        path => "/var/log/nginx/access.log"
        start_position => beginning
    }
}
filter {
    grok {
        match => { "message" => "%{COMBINEDAPACHELOG}"}
    }
    geoip {
        source => "clientip"
    }
}
output {
    elasticsearch {}
    stdout {}
}
```

- logstash 실행

```
# test
bin/logstash -f logconf/nginx.conf --configtest
# run
bin/logstash -f logconf/nginx.conf
# background run
nohup bin/logstash -f logconf/nginx.conf &
```

Kibana 통계

시각화(Visualize)

- Terms(request.raw, clientip.raw, ...) 또는 Filters(request: "/hello.html", ...) 이용해서 차트 생성
- 테이블, 라인차트, 파이차트, 지도 등 가능
- 만들어진 차트는 저장 가능

대시보드 만들기

- 저장된 차트를 한 화면에서 볼 수 있도록 추가, 레이아웃 가능

part 2

Logstash

- 필드 추가

```
field{
  mutate {
      add_field => {
          "reqs" => "%{request}"
      }
  }
}
```

- 분리

```
field{
  mutate {
      split => ["reqs", "?"]
      add_field => { "uri" => "%{reqs[0]}" }
      add_field => { "req_uri" => "%{reqs[0]}" }
#         add_field => { "querystring" => "%{reqs[1]}" }
  }
}
```

- 필드 제거

```
  mutate {
      remove_field => [
          "reqs",
          "uri"
      ]
  }
```

- 파라미터 필드 만들기

```
filter {
  mutate {
      add_field => {
          "tmp" => "%{request}"
      }
  }
  if [tmp] =~ "W?" {
      mutate {
          split => [
              "tmp", "?"
          ]
          add_field => {
              "params" => "%{[tmp][1]}"
          }
      }
      kv {
          field_split => "&"
          source => "params"
          include_keys => [ "category", "utm_source" ]
          prefix => "param_"
      }
  }
}
```

- 또는

```
  # params
  if [request] =~ "W?" {
      kv {
          field_split => "&"
          source => "querystring"
          include_keys => [ "query", "redirectUrl" ]
          prefix => "param_"
      }
  }
```

- 이미지 제거

```
filter {
  if [message] =~ "^#|W.(css|js|ico|png|xml|jpg|JPG|gif|jpeg|eotW?) " {
      drop {}
  }
}
```

- useragent 파싱

```
  useragent {
      source => "agent"
  }
```

- timestamp 조정(apache log)

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

- https://www.elastic.co/guide/en/logstash/current/plugins-filters-date.html
- urldecode

```
 urldecode {
     field => "params"
 }
```

- to integer

```
  mutate {
      convert => [ "bytes", "integer" ]
  }
```

Kibana

- 질의어 문법(query syntax)
  - Lucene 검색 엔진의 문법 그대로 사용(https://lucene.apache.org/core/2_9_4/queryparsersyntax.html)
- request: "uri"
- 제외 -device : "Spider"

elasticsearch

- 데이터 지우기
  - curl -XDELETE http://localhost:9200/logstash*

Filebeat with logstash

- (Optional)
- logstash forwarder(deprecated) 의 경량(lightweight) 버전

- logstash plugin 설치

```
cd ~/local/logstash
./bin/logstash-plugin install logstash-input-beats
```

- filebeat 설치

```
cd ~/local
wget https://download.elastic.co/beats/filebeat/filebeat-1.3.1-x86_64.tar.gz
ln -s filebeat-1.3.1-x86_64 filebeat
cd filebeat
# elasticsearch 부분 #으로 주석 처리
  # elasticsearch:
    #hosts: ["localhost:9200"]
# logstash 부분 # 주석 해제
  logstash:
    hosts: ["localhost:5044"]

# filebeat.yml 내용 중 로그 위치 변경 `/var/log/nginx/*.log`
```

- logconf/nginx.conf 파일 변경

```
input {
  beats {
    port => 5044
  }
}
filter {
    grok {
        match => [
            "message", "%{COMBINEDAPACHELOG}",
            "message", "%{COMMONAPACHELOG}"
        ]
    }
    geoip {
        source => "clientip"
    }
}
output {
  elasticsearch {
    hosts => "localhost:9200"
    manage_template => false
    index => "%{[@metadata][beat]}-%{+YYYY.MM.dd}"
    document_type => "%{[@metadata][type]}"
  }
```

```
}
```

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

- start shell

```
echo "nohup ./filebeat -e -c filebeat.yml &" > start.sh
chmod +x start.sh
./start.sh
```

ELK with PM2

- 2G짜리 메모리의 인스턴스에서 ELK를 돌리면 OutOfMemory 때문에 종종 Elasticsearch 또는 Kibana가 죽습니다.
- 고육지책으로 Kibana는 node 기반이기 때문에 pm2로 Kibana가 죽으면 자동으로 살리는 방법입니다.
- download from http://nodejs.org and install node.js

```
npm install -g pm2
cd ~/local/kibana
pm2 start bin/cli
```

- check kibana status with pm2 list
- pm2 logs path is placed in ~/.pm2/logs

kibana 인증 with nginx

```
sudo vi /etc/nginx/nginx.conf
```

- server_name: 아래 kibana 프록시 설정

```
        auth_basic "Restricted Access";
        auth_basic_user_file /etc/nginx/htpasswd.users;

        location / {
                proxy_pass http://localhost:5601;
                proxy_http_version 1.1;
                proxy_set_header Upgrade $http_upgrade;
                proxy_set_header Connection 'upgrade';
                proxy_set_header Host $host;
                proxy_cache_bypass $http_upgrade;
        }
```

- nginx 재시작
  - sudo service nginx restart
- 5601 포트는 막고 80으로만 접속

참고

- Logstash grok patterns
  - https://github.com/logstash-plugins/logstash-patterns-core/blob/master/patterns/grok-patterns
- ELKR (ElasticSearch + Logstash + Kibana + Redis) 를 이용한 로그분석 환경 구축하기
  - http://brantiffy.axisj.com/archives/418
- 2016 ELK 스택으로 서울시 지하철 대시보드 만들기 *추천*
  - https://youtu.be/xPjNtd8xUZo
- EMOCON 2015 F/W ELK 스택을 사용한 서울시 지하철 대시보드 만들기
  - https://youtu.be/ec-XzM6_CgU
- ELK 구축하기 1 - LOGSTASH
  - http://linux.systemv.pe.kr/elk-구축하기-1-logstash/
- [Ubuntu] ELK 설치 및 테스트 하기
  - http://digndig.kr/ubuntu/449/
- Splunk 대체 Solution으로서의 ELK Stack
  - http://blog.embian.com/18
- How To Install Elasticsearch, Logstash, and Kibana 4 on Ubuntu 14.04
  - https://www.digitalocean.com/community/tutorials/how-to-install-elasticsearch-logstash-and-kibana-4-on-ubuntu-14-04
- ELK 프로그래밍 방송 영상
  - http://bit.ly/okdevtv-elk

- Logstash Configuration
  - https://www.elastic.co/guide/en/logstash/current/event-dependent-configuration.html
- Elasticsearch(Lucene) Query Syntax
  - https://lucene.apache.org/core/2_9_4/queryparsersyntax.html
- ELK Kibana 사용법
  - https://dl.dropboxusercontent.com/u/2385737/Kibana-basic.pdf
- okky.conf
  - https://okdevtv.com/md/elk/okky.conf

What Else?