3. ELK stack Install

ELK Stack Install

ELK Stack 패키지 설치 - ELK서버에서 수행

1. repository 구성

```
$ vi /etc/yum.repos.d/elk.repo


[logstash-7.x]
name=Elastic repository for 7.x packages
baseurl=https://artifacts.elastic.co/packages/7.x/yum
gpgcheck=0
gpgkey=https://artifacts.elastic.co/GPG-KEY-elasticsearch
enabled=1
autorefresh=1
type=rpm-md
```

2. java 설치 (java 설치 버전은 1.8 버전으로 배포 진행)

```
$ yum install java -y
$ java -version
openjdk version "1.8.0_282"
OpenJDK Runtime Environment (build 1.8.0_282-b08)
OpenJDK 64-Bit Server VM (build 25.282-b08, mixed mode)
```

3. Logstash / Elasticsearch / Kibana 설치

```
$ yum install logstash elasticsearch kibana -y
```

ELK Stack Config - ELK서버에서 수행

1. kibana 설정

```
$ vi /etc/kibana/kibana.yml
...
server.host: "0.0.0.0" (외부에서 webui 접근이 0.0.0.0 으로 사용 )
...
elasticsearch.hosts: ["http://localhost:9200"] (Elasticsearch 설치 서버 ip)
...
i18n.locale: "ko-KR"
```

2. Cluster 기반의 Elasticsearch 설정

```
$ vi /etc/elasticsearch/elasticsearch.yml
...
cluster.name: es-cluster # 클러스터링 할 서버는 동일한 cluster.name값으로 설정
node.name: ${HOSTNAME} # 클러스터링할 서버 호스트네임 (노드별로 uniq한 값이어야 함 )
path.data: /data/elasticsearch # Elasticsearch Data경로
path.logs: /var/log/elasticsearch # Elasticsearch 로그경로
network.host: 0.0.0.0 # 외부에서 접속시 설정
discovery.seed_hosts: ["192.168.0.10", "192.168.0.11", "192.168.0.12"] #Elasticsearch Discovery 호스트 설정
cluster.initial_master_nodes: ["192.168.0.10", "192.168.0.11", "192.168.0.12"] #마스터 서버 리스트
...
http.port: 9200 # http 호스트 사용하는 포트
transport.tcp.port: 9300 # 데이터 전송 포트
...
node.master: true # master 노드 역활시 true
node.data: true # data 노드 역활 적용시 true
...
index.number_of_replicase: 1 #각 인덱스를 3개의 replicaset으로 구성
index.number_of_shards: 2 #각 인덱스를 샤딩
...
node.attr.box_type: hot #노드역활 설정 (hot / warm / clod 중 선택 )
```

3. logstash config 설정

```
$ vi /etc/logstash/conf.d/nginx.conf
input {
  beats {
    port => 5044
    host => "0.0.0.0"
  }
}

output {
  elasticsearch {
    hosts => ["http://localhost:9200"]
    index => "nginx-%{+YYYY.MM.dd}"
    #user => "elastic"
    #password => "changeme"
  }
}



#config 참고해서 logstash 구동하도록 설정 (기존 설정값 삭제 후 아래내용 설정 )

$ vi /etc/systemd/system/logstash.service
...
ExecStart=/usr/share/logstash/bin/logstash -f /etc/logstash/conf.d/nginx.conf
...

$ systemctl daemon-reload
```

4. filebeat 설치 - log를 전달할 서버에 설치
    1. repository 구성

```
$ vi /etc/yum.repos.d/elk.repo

[logstash-7.x]
name=Elastic repository for 7.x packages
baseurl=https://artifacts.elastic.co/packages/7.x/yum
gpgcheck=0
gpgkey=https://artifacts.elastic.co/GPG-KEY-elasticsearch
enabled=1
```

```
autorefresh=1
type=rpm-md
```

2. 패키지 설치

```
$ yum install filebeat -y
```

3. filebeat 설정

```
$ vi /etc/filebeat/filebeat.yml
...
filebeat.inputs:
- type: log
  enabled: true   #true로 변경
  paths: #Logstash로 전달할 로그파일 혹은 경로를 설정하면 된다 .
    - /svc/stg/web/logs/access.log
    - /var/log/cmd.log
    - /var/log/kibana/*
...
setup.kibana:
  host: "192.168.0.11:5601" #Kibana 서버 IP
...
#output.elasticsearch: #filebeat -> logstash로 전달할것이기 때문에 elasticsearch는 주석처리
# hosts: ["localhost:9200"]
...
output.logstash:  #주석해제
  hosts: ["192.168.0.11:5044"] #logstash 서버 ip/포트 설정
...
```

프로세스 실행

1. kibana / elasticsearch 프로세스 실행 - ELK 서버에서 수행

```
$ systemctl enable kibana --now
$ systemctl enable elasticsearch --now
$ systemctl enable logstash --now
```

2. filebeat 서비스 구동 - Log 전달할 서버에서 수행

```
$ systemctl enable filebeat --now
```

서비스 작동 확인

1. logstash 기능 확인

```
#logstash 포트 LISTEN 확인
$ netstat -antp| grep 5044 | grep LISTEN
tcp6       0      0 :::5044                 :::*                    LISTEN      6730/java



#filebeat → logstash로 데이터 전송이 되는지 확인 (logstash서버에서 수행 )
$ tcpdump -nn port 5044 -i bond0
14:11:35.759481 IP 192.168.0.11.5044 > 192.168.10.2.34160: Flags [P.], seq 379:385, ack 87143,
win 1432, options [nop,nop,TS val 341934898 ecr 464702009], length 6
14:11:35.760109 IP 192.168.10.2.34160 > 192.168.0.11.5044: Flags [.], ack 385, win 115, options
[nop,nop,TS val 464702013 ecr 341934898], length 0
```

2. Elasticsearch 기능 작동 확인

```
#Elasticsearch 포트 LISTEN 확인
$ netstat -antp | grep 9200 | grep LISTEN
tcp6       0      0 :::9200                 :::*                    LISTEN      11324/java

#logstsh에서 전달한 데이터가 elasticsearch에서 index수집되는지 확인
$ curl --connect-timeout 2 -XGET http://127.0.0.1:9200/_cat/indices?v
health status index                           uuid                   pri rep docs.count docs.deleted store.size
pri.store.size
green  open   .kibana_task_manager_7.12.0_001 jNMZ2LZcRtqYkwCrQqCsdQ   1   1         9
10     92.6kb         73.7kb
green  open   .apm-custom-link                MmzSDfLtSXuQCYwqXoYbFg   1   1          0          0
416b           208b
green  open   .apm-agent-configuration       xbHoMaQ0QUS2WAsOy3Uspw   1   1          0          0
416b           208b
green  open   .async-search                  pMPoD_2OQzue0gJH-vSdig   1   1          1          0
```

```
90.9kb        46.9kb
green  open   .kibana_7.12.0_001            Qmo4u9gjTOmihGVwJlguqQ   1   1       22         0
6.3mb         4.2mb
green  open   .kibana-event-log-7.12.0-000001 VykSos0vR1W_l5F2E5G2pg   1   1        2         0
21.9kb        10.9kb
green  open   .elastichq                   7sr4ATTsSnasGRH4tJhCBA   1   1        1         0
13.7kb         6.8kb
green  open   .tasks                       X2B8PyG5SMCV0dPAo6eH4g   1   1        2         0
15.5kb         7.7kb
```

3. 클러스터 구성 정보 확인

```
$ curl --connect-timeout 2 -XGET http://127.0.0.1:9200/_cluster/health?pretty=true
{
  "cluster_name" : "es-cluster",   #클러스터 이름
  "status" : "green",              #클러스터 상태
  "timed_out" : false,
  "number_of_nodes" : 3,           # 마스터 노드 수
  "number_of_data_nodes" : 3,      # 데이터 노드 수
  "active_primary_shards" : 9,
  "active_shards" : 18,
  "relocating_shards" : 0,
  "initializing_shards" : 0,
  "unassigned_shards" : 0,
  "delayed_unassigned_shards" : 0,
  "number_of_pending_tasks" : 0,
  "number_of_in_flight_fetch" : 0,
  "task_max_waiting_in_queue_millis" : 0,
  "active_shards_percent_as_number" : 100.0
```

4. Kibana 구성정보 확인

```
#kibana 포트   LISTEN 확인
$ netstat -antp | grep 5601 | grep LISTEN
tcp        0      0 0.0.0.0:5601            0.0.0.0:*               LISTEN      13513/node

#kibana 접속   확인
$ curl -IL -XGET http://192.158.0.11:5601/app/home/
```

```
HTTP/1.1 200 OK
content-type: text/html; charset=utf-8
content-security-policy: script-src 'unsafe-eval' 'self'; worker-src blob: 'self'; style-src 'unsafe-inline'
'self'
kbn-name: SKB-DJK-ELK1
kbn-license-sig: 0f6943d9f4b6625724a0d78fe647bbe2f284a6e24fb46f587b17b1b0bec18e34
cache-control: private, no-cache, no-store, must-revalidate
content-length: 127971
vary: accept-encoding
accept-ranges: bytes
Date: Fri, 09 Apr 2021 05:52:47 GMT
Connection: keep-alive
Keep-Alive: timeout=120
```

Kibana Index Pattern 설정

1. WebUI : http://kibanaIP:5601
   image-1658595831958.png

2. Management → Stack Management → Kibana → Index patterns
   image-1658595840322.png

3. {{ index name }}-YYYY.mm.DD 패턴이 보이지 않으면 elasticsearch에서 데이터가 아직 유입되지 않은 상태 .
4. Search에서 등록할 index명 입력 후 Create index pattern 선택
5. Time filed에는 @timestamp 선택 후 Create index pattern 선택
   image-1658595851071.png

6. Analytics → discover 선택하면 유입된 데이터 확인 가능
   image-1658595858282.png

WEB UI를 통한 Elasticsearch 상태 확인

- docker 설치 후 cerebro container 구동

```
$ docker container run -d --name cerebro --restart always -p 9000:9000 -m 512m
lmenezes/cerebro:latest
```

```
8d691f585fa8: Pull complete
3da6fe7ff2ef: Pull complete
e22147996cc0: Pull complete
8df48a2d4467: Pull complete
45e578fea430: Pull complete
Digest: sha256:1cd0765418f1737de3533648d549655437eb550ee0cfad27488c19e620028f2f
```

- WEB UI 로그인 : http://elk서버 ip:9200
  image-1658595875298.png

- Node address에 ELK 설치된 서버 IP입력
- 첫화면 (Overview) : Elastic 서버 & 인덱스 상태확인
  image-1658595879212.png
- Nodes : 노드 상태 확인 (별표에 색깔 칠해진 노드가 master 노드)
  image-1658595901657.png

참고 Site

- *logstash input : https://www.elastic.co/guide/en/logstash/current/input-plugins.html*
- *filebeat log : https://www.elastic.co/guide/en/beats/filebeat/current/configuration-filebeat-options.html#filebeat-input-types*
- *elk intsall: https://www.elastic.co/guide/en/beats/filebeat/current/setup-repositories.html*
- *elk stack 소개 : https://medium.com/naver-cloud-platform/%EB%84%A4%EC%9D%B4%EB%B2%84-%ED%81%B4%EB%9D%BC%EC%9A%B0%EB%93%9C-%ED%94%8C%EB%9E%AB%ED%8F%BC%EC%9D%84-%ED%99%9C%EC%9A%A9%ED%95%B4-elk-elasticsearch-logstash-kibana-%EC%8A%A4%ED%83%9D-%EA%B5%AC%EC%B6%95%ED%95%98%EA%B8%B0-4cbaf5dd4305*
- *logstach / filebeat 비교 : https://velog.io/@deet1107/logstash-filebeat*
- *ElasticSearch 이중화 : https://nesoy.github.io/articles/2019-01/ElasticSearch-System-Architecture*
- *elasticsearch data 구조 : https://koocci-dev.tistory.com/13*

Revision #5

Created 17 July 2022 18:34:21 by artop0420

Updated 11 November 2024 10:33:23 by artop0420