 grafana\_report.md

# Grafana - Monitoring Tool

---

## Version

---

- Docker: 18.6.1-ce
- Grafana: 5.2.4 (Docker Default)

## Grafana with Docker

---

### Docker Command

```
$ docker run -d -p 3000:3000 --name=grafana --restart=always \  
-e "GF_INSTALL_PLUGINS=grafana-piechart-panel" \  
-e "GF_INSTALL_PLUGINS=alexanderzobnin-zabbix-app" \  
-e "GF_SMTP_ENABLED=true" \  
-e "GF_SMTP_HOST=smtp.test.com:25" \  
-e "GF_SMTP_FROM_ADDRESS=admin@grafana.localhost" \  
-e "GF_SMTP_USER=user_mail" \  
-e "GF_SMTP_PASSWORD=user_pw" \  
-e "GF_SMTP_SKIP_VERIFY=true" \  
--privileged grafana/grafana
```

### Run Grafana

- URI: [localhost:3000](http://localhost:3000)
- Admin username: admin
- Admin password: admin

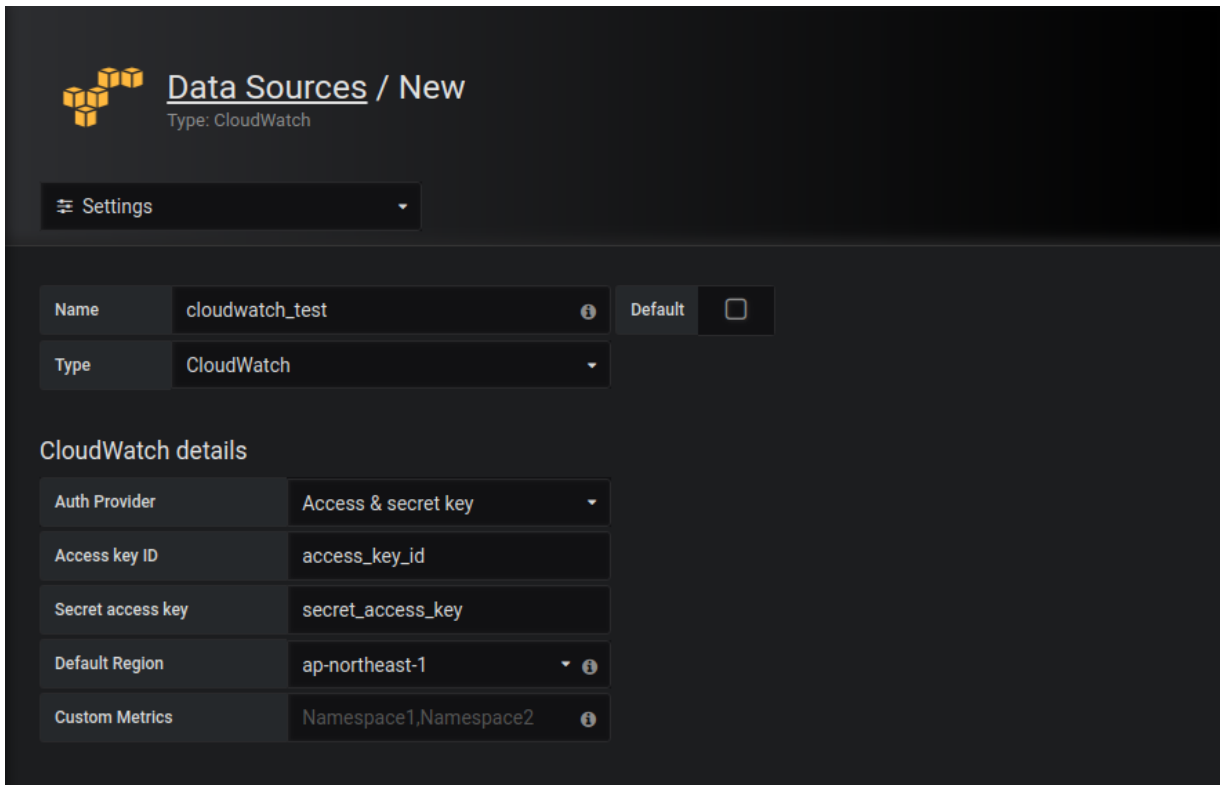
## Grafana

---

### Data Source

#### CloudWatch

- Type: **CloudWatch**
- CloudWatch details
  - Auth Provider: **Access & secret key** (Access key ID & Secret access key 는 AWS에서 부여 받은 Key 정보 입력)
  - Default Region: 모니터링 할 Instance가 있는 Region으로 선택 (Dash Board 그릴 때 다른 Region 선택 가능)
- Save & Test 버튼 누른 후 **Data source is working** 이라는 문구가 뜨면 연동 성공



**Data Sources / New**  
Type: CloudWatch

Settings

Name: cloudwatch\_test ⓘ Default: ☐

Type: CloudWatch

**CloudWatch details**

Auth Provider: Access & secret key

Access key ID: access\_key\_id

Secret access key: secret\_access\_key

Default Region: ap-northeast-1 ⓘ

Custom Metrics: Namespace1, Namespace2 ⓘ

### Elasticsearch

- Type: **Elasticsearch**
- HTTP
  - URL: 모니터링 할 Elasticsearch URI 입력, 반드시 Port 입력
- Elasticsearch details
  - Index name: 모니터링 할 index 입력
  - Pattern: index의 날짜 pattern에 맞게 선택
  - Time field name: 로그의 time field 입력 (default는 @timestamp)
  - Version: 모니터링 하는 Elasticsearch version (default는 5.x)
  - Min time interval: 데이터 수집 주기 (default는 10s)
- Save & Test 버튼 누른 후 **Index OK. Time field name OK.** 이라는 문구가 뜨면 연동 성공

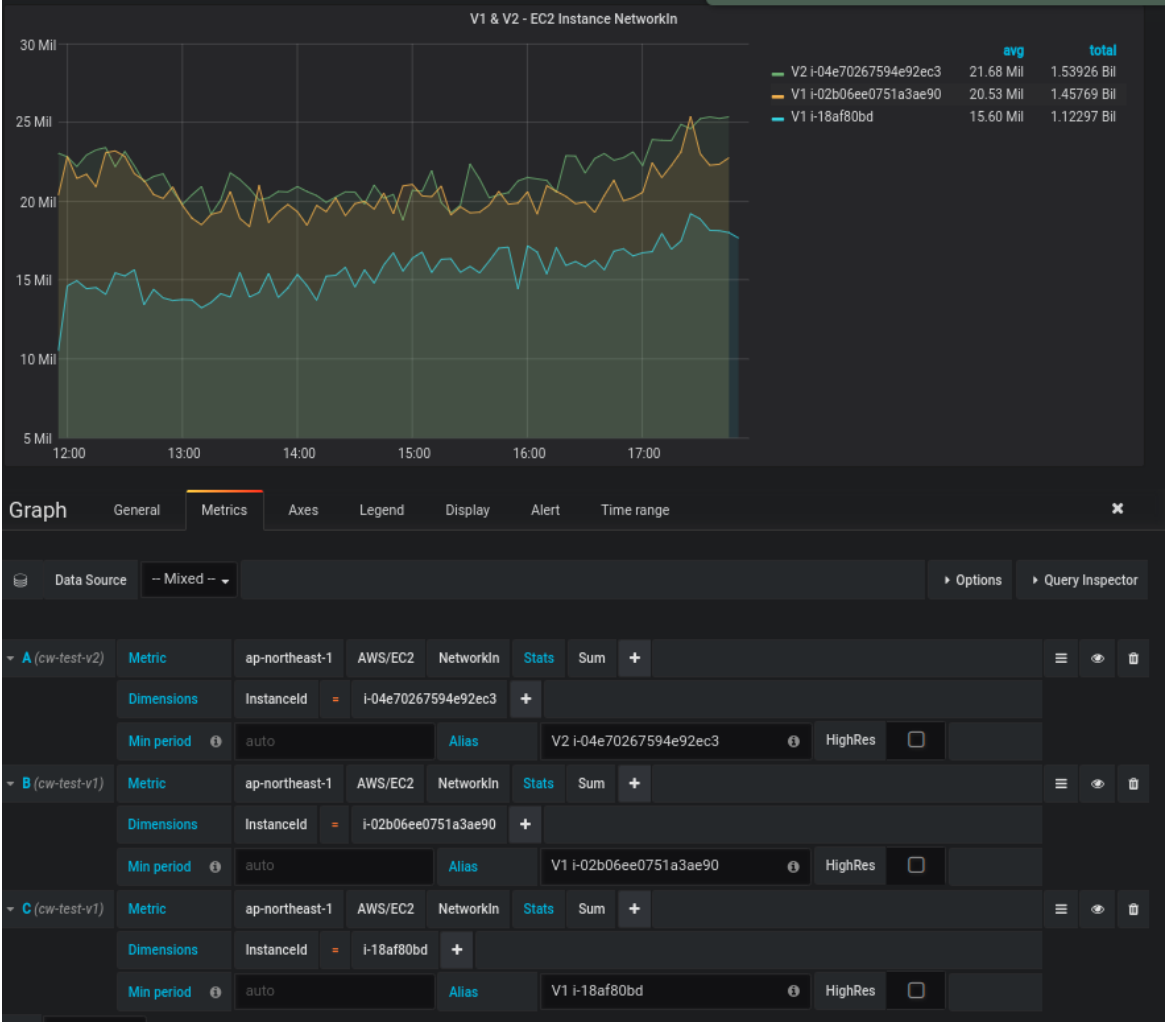
The screenshot shows the 'Data Sources / elasticsearch\_test' configuration page in Grafana. The page is dark-themed and contains several sections for configuring the Elasticsearch data source.

- Settings:** A tab at the top left.
- Name:** 'elasticsearch\_test' with an information icon and a 'Default' checkbox.
- Type:** 'Elasticsearch' with a dropdown arrow.
- HTTP:**
  - URL:** 'http://localhost:9200' with an information icon.
  - Access:** 'Server (Default)' with a dropdown arrow and a 'Help' link.
- Auth:**
  - Basic Auth:** A checkbox, followed by 'With Credentials' (information icon) and another checkbox.
  - TLS Client Auth:** A checkbox, followed by 'With CA Cert' (information icon) and another checkbox.
  - Skip TLS Verification (Insecure):** A checkbox.
- Advanced HTTP Settings:**
  - Whitelisted Cookies:** A text input field with 'Add Name' and an information icon.
- Elasticsearch details:**
  - Index name:** '[index-]YYYY.MM.DD' with a 'Pattern' dropdown set to 'Daily'.
  - Time field name:** '@timestamp'.
  - Version:** '5.6+' with a dropdown arrow.
  - Max concurrent Shard Requests:** '256'.
  - Min time interval:** '10s' with an information icon.

## Dash Board 지원 기능

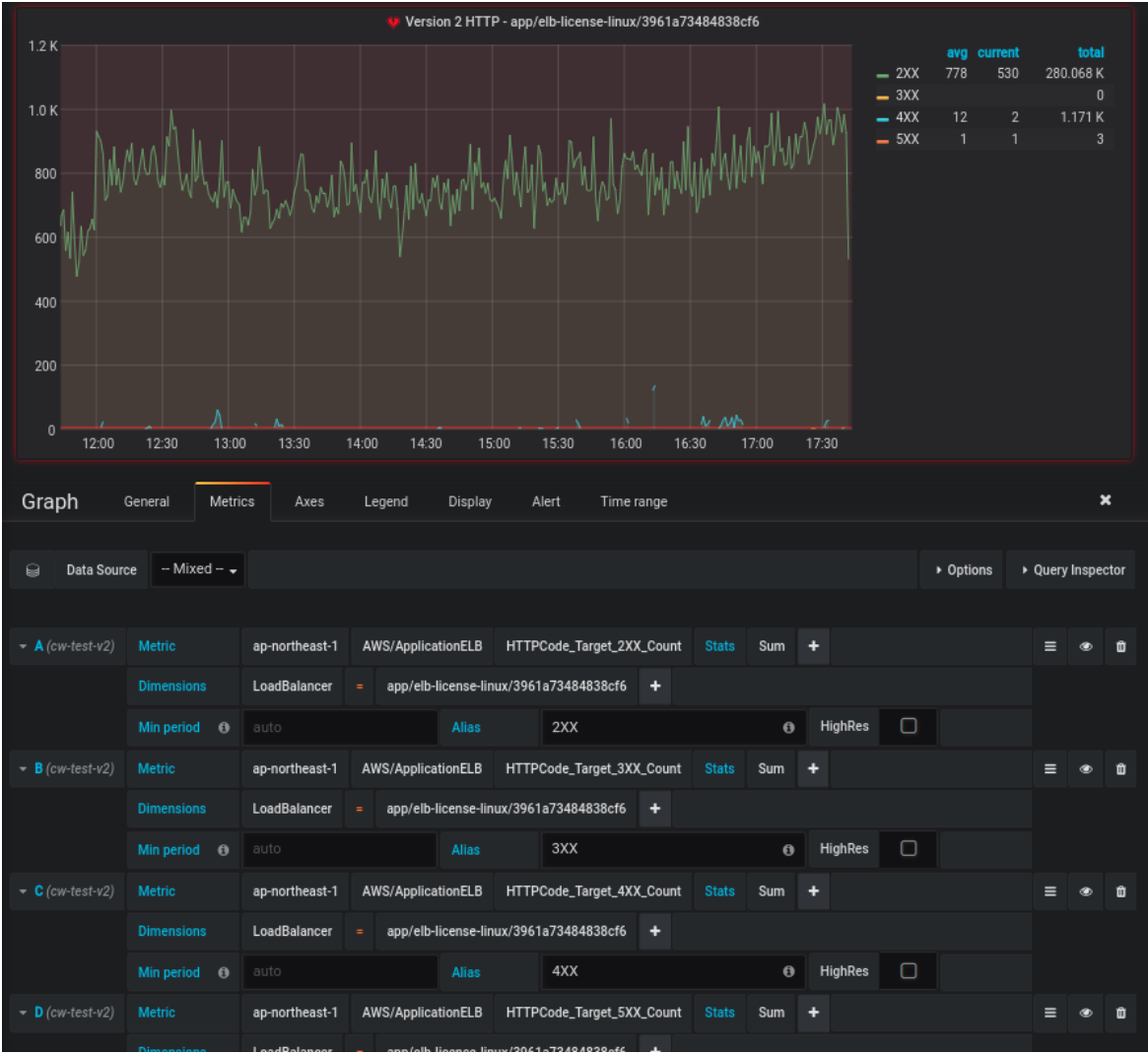
- Dash Board에 생성할 수 있는 Graph 개수 제한 없음 (Loading 및 Dash Board 설정을 바꿀 시 적용 시간이 김)
- Data Source에 대한 계정 정보만 입력하면 해당 계정으로 수집하는 데이터 모두 열람 및 사용 가능
- Plugin
  - Pie-Chart-Panel plugin을 통한 데이터 간 ratio 연산 가능
  - Zabbix plugin을 통한 Zabbix-Grafana 연동 (*현재 Grafana가 Zabbix의 버전을 못해서 모니터링 불가*)
- Panel
  - Graph, Singlestat, Table, Heatmap, Alert List 등
  - Axes, Legend, Display, Alert, Time Range 설정
- Snapshot 기능을 제공하여 Dash Board 백업 지원
- Dash Board Settings
  - Dash Board refresh time custom → 0.5초까지 확인
  - Monitoring time range custom → 1초까지 가능
  - Versions - Dash Board 저장 시점을 기록하여 버전 관리
  - Permissions - 사용자에게 Admin, Editor, Viewer 역할을 부여하여 Dash Board 접근 권한 관리
  - JSON Model - 해당 Dash Board의 settings 관련 데이터는 json 형태로 제공
- Graph panel에서 여러 Data Source, 여러 지표 혼합 가능

**e.g.1.**  
CloudWatch version 1 과 version 2 에 있는 모든 EC2 instance에 대한 NetworkIn 정보 모두를 한 graph에 표시 가능



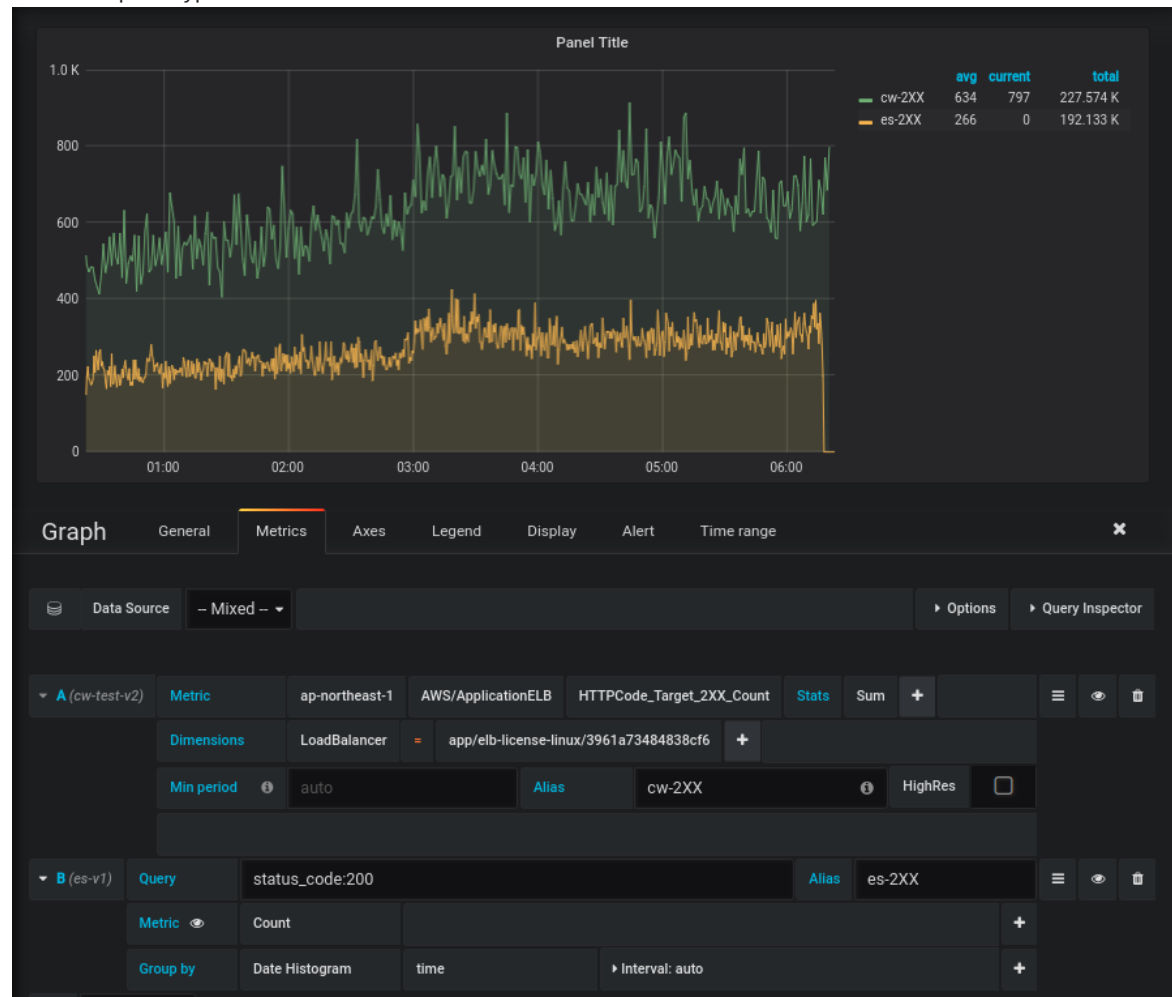
e.g.2.

하나의 LoadBalancer에 대한 모든 지표(HTTPCode\_Target\_2XX, 3XX, 4XX, 5XX 등)를 하나의 graph에 표시 가능



**e.g.3.**

하나의 Graph에 Type이 다른 Data Source 데이터 표시 가능



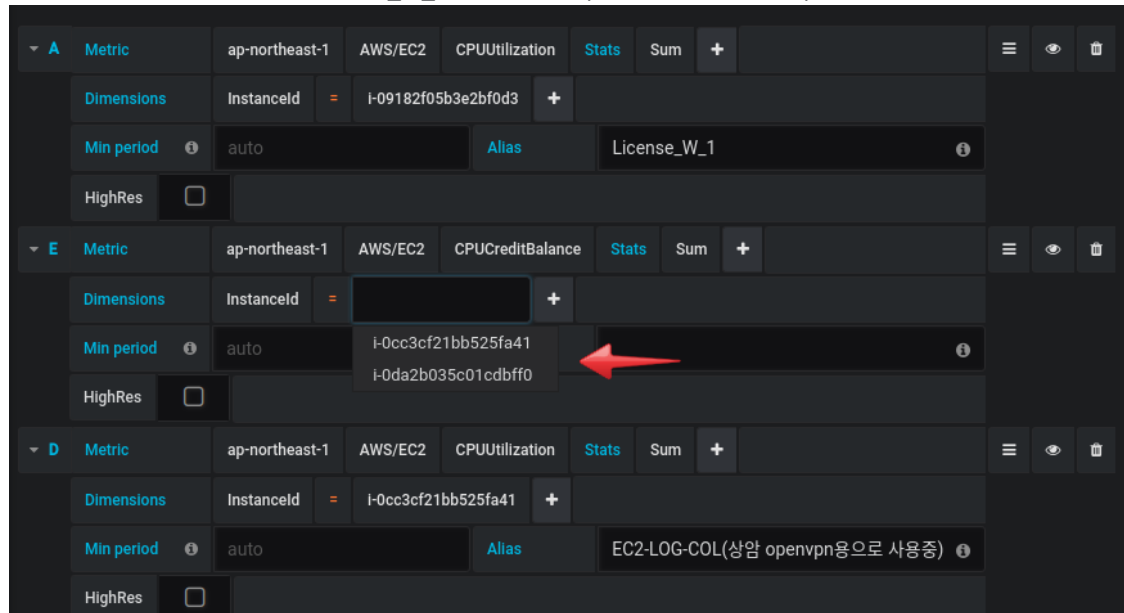
- Playlist를 통해 기존에 있던 여러 Dash Board를 묶어서 한번에 모니터링 가능

**CloudWatch**

- Data Source Type에 맞게 Metric 지표 자동 쿼리
  - region, namespace, metric, Dimensions 순
  - 입력된 Access Key에 상응하는 계정의 CloudWatch가 리포팅 하는 모든 EC2/LoadBalancer의 지표를 사용자에게 모두 표시 → 사용자가 선택한 지표에 대하여 데이터가 없는 EC2/LoadBalancer의 InstanceID/LoadBalancerName는 표시하지 않음

e.g.

CloudWatch Version2 EC2 Instance 중 License\_W\_1 인스턴스가 수집하지 않는 지표인 CPUCreditBalance 지표를 선택 후 InstanceID로 검색하면 License\_W\_1의 InstanceID(i-09182f05b3e2bf0d3)를 표시하지 않음



## Elasticsearch

- 아직까지는 특별한 기능 없음

## 제한 사항

- Docker 기반이기 때문에 플러그인 설치가 Container를 만들 때만 가능
- 지표 및 Key list 받아 오는 기능 없음
- Alert에 대한 메일 및 API Notification 에러
  - SMTP 관련 이슈
    - Docker -v 옵션: 로컬에서 grafana.ini 파일 수정하는 방법 시도했으나 볼륨 설정 옵션을 넣으면 에러
    - Docker -e 옵션: SMTP 값 변경 후 Grafana에서는 메일 발신 완료 뜸 → 알림 메일 수신 안됨
  - Grafana Notification Service
    - Email: Send Test Failed → SMTP 이슈로 추정
    - Slack 연동: Grafana에서 Slack API App Token 입력해도 Send Test Failed

## CloudWatch

- CloudWatch 만의 제한 사항 아직까지 없음

## Elasticsearch

- 데이터에 대한 range 설정이 번거로움

e.g.

status\_code가 200일 경우와 아닌 경우만 확인 가능

4XX 데이터를 확인하고 싶으면 400 to 499 로 지정하거나 403, 404와 같이 지정

- 자동 쿼리 기능 없음
- 쿼리할 때 ", " (쌍따옴표) 인식 못함
- key는 따옴표 없이, value는 type(int, str 등)에 따라 따옴표 사용

## 기타 Grafana 기능

- Alert Rules와 Notification channel에서 설정된 Alert 목록 확인
- Sever Admin 탭에서 사용자 추가 가능