# - 오픈소스 Helm 차트 분석 - Grafana 개요

**Helm Chart**

**grafana**

Chart.yaml
values.yaml
README.md

helm install my-grafana .
-f **ci/**with-dashboard-json-values.yaml
-f **ci/**with-image-renderer-values.yaml
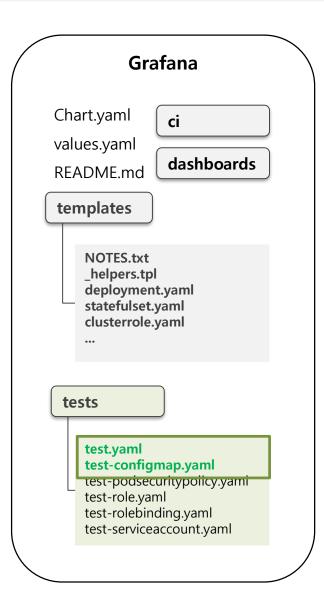
**templates**

helm **test** my-grafana

**ci**

NOTES.txt
_helpers.tpl

**tests**

_pod.tpl
deployment.yaml
statefulset.yaml
clusterrole.yaml
clusterrolebinding.yaml
configmap-dashboard-provider.yaml
configmap.yaml
image-renderer-deployment.yaml
podsecuritypolicy.yaml
pvc.yaml
role.yaml
dashboards-json-configmap.yaml
rolebinding.yaml
secret.yaml
...

test-configmap.yaml
test-podsecuritypolicy.yaml
test-role.yaml
test-rolebinding.yaml
test-serviceaccount.yaml
test.yaml

default-values.yaml
with-dashboard-json-values.yaml
with-dashboard-values.yaml
with-image-renderer-values.yaml

**dashboards**

custom-dashboard.json

{{ **.Files.get** dashboard/custom-dashboard.json}}

**HELM**

**Grafana**

Chart.yaml
values.yaml
README.md

ci

dashboards

templates

NOTES.txt
_helpers.tpl
deployment.yaml
statefulset.yaml
clusterrole.yaml
...

tests

test.yaml
test-configmap.yaml
test-podsecuritypolicy.yaml
test-role.yaml
test-rolebinding.yaml
test-serviceaccount.yaml

NAME: my-grafana
LAST DEPLOYED: Thu..
NAMESPACE: default
STATUS: **deployed**
REVISION: 1
NOTES:
....

**배포**

helm install [release name] .

**배포 테스트**

**helm test [release name]**

NAME: my-grafana
LAST DEPLOYED: Thu..
NAMESPACE: default
STATUS: deployed
REVISION: 1
**TEST SUITE: my-grafana-test**
**Last Started: Thu...**
**Last Completed: Thu..**
**Phase: Succeeded**   | **Phase: Failed**
NOTES:
.....

**Deployment**

**ReplicaSet**

**POD**

Grafana

**POD**

**test**

Terminate
Complete

Terminate
**Error**

**Service
Check**

# - 오픈소스 Helm 차트 분석 - Grafana (helm test)

## Grafana

Chart.yaml
values.yaml
README.md

**ci**

**dashboards**

**templates**

NOTES.txt
_helpers.tpl
deployment.yaml
statefulset.yaml
clusterrole.yaml
...

**tests**

**test.yaml**
**test-configmap.yaml**
test-podsecuritypolicy.yaml
test-role.yaml
test-rolebinding.yaml
test-serviceaccount.yaml

### values.yaml

```
testFramework:
  enabled: true
  image: "bats/bats"
  ...
```

### test.yaml

```
{{- if .Values.testFramework.enabled }}
apiVersion: v1
kind: Pod
metadata:
  name: {{ template "grafana.fullname" . }}-test
  annotations:
    "helm.sh/hook": test-success
    "helm.sh/hook": test
                        grafana.namespace" . }}
spec:
...
  containers:
    - name: {{ .Release.Name }}-test
      image:
      command: ["/opt/bats/bin/bats", "-t", "/tests/run.sh"]
      volumeMounts:
        - mountPath: /tests
          name: tests
          readOnly: true
  volumes:
  - name: tests
    configMap:
      name: {{ template "grafana.fullname" . }}-test
  restartPolicy: Never
{{- end }}
```

### 배포 테스트

**helm test [release name]**

```
NAME: my-grafana
LAST DEPLOYED: Thu May 27 08:58:53 2021
NAMESPACE: default
STATUS: deployed
REVISION: 1
TEST SUITE: my-grafana-test
Last Started: Fri May 28 10:27:21 2021
Last Completed: Fri May 28 10:27:24 2021
Phase: Succeeded
```

### test-configmap.yaml

```
{{- if .Values.testFramework.enabled }}
apiVersion: v1
kind: ConfigMap
metadata:
  name: {{ template "grafana.fullname" . }}-test
data:
  run.sh: |-
    @test "Test Health" {
      url="http://{{ template "grafana.fullname" . }}/api/health"
      code=$(wget --server-response --spider --timeout 10
      --tries 1 ${url} 2>&1 | awk '/^  HTTP/{print $2}')
      [ "$code" == "200" ]
    }
{{- end }}
```

## values.yaml

```
persistence.:
  enabled: true
  type: pvc          [statefulset]
  # existingClaim:
```

### Template

```
.Template.BasePath
.Template.Name
```

### Grafana

```
Chart.yaml
values.yaml
README.md

templates

    NOTES.txt
    _helpers.tpl
    deployment.yaml
    statefulset.yaml
    clusterrole.yaml
    configmap.yaml
```

## deployment.yaml

```
{{ if (or (not .Values.persistence.enabled) (eq .Values.persistence.type "pvc")) }}
apiVersion    [false]                          [true]
kind: Deployment
metadata:
  name: {{ template "grafana.fullname" . }}
...
spec:
...
  template:
    metadata:
    ...
      annotations:                    [grafana/template/configmap.yaml]
        checksum/config: {{ include (print $.Template.BasePath "/configmap.yaml") . | sha256sum }}
        checksum/dashboards-json-config: {{ include (print $.Template.BasePath "/dashboards-json-
configmap.yaml") . | sha256sum }}
        ...
{{- end }}
{{- if .Values.envRenderSecret }}
        checksum/secret-env: {{ include (print $.Template.BasePath "/secret-env.yaml") . | sha256sum }}
{{- end }}
{{- with .Values.podAnnotations }}
{{ toYaml . | indent 8 }}
{{- end }}
    spec:
      {{- include "grafana.pod" . | nindent 6 }}
{{- end }}
```

[grafana/template/deployment.yaml]

## _pod.tpl

```
{{- define "grafana.pod" -}}

serviceAccountName: {{ template
"grafana.serviceAccountName" . }}
{{- if .Values.securityContext }}
securityContext:
{{- if .Values.schedulerName }}
schedulerName: "{{ .Values.schedulerName }}"
{{- end }}
...

{{- end }}
```

```
annotations:
  checksum/config:
e62ba605067cc7f46613c3194e97e9cb6da
0061334e9087e92906da260ff49ef
```

## statefulset.yaml

```
{{- if and .Values.persistence.enabled (not .Values.persistence.existingClaim)
(eq .Values.persistence.type "statefulset")}}
apiVersion: apps/v1
kind: StatefulSet
metadata:
...
{{- end }}
    spec:
      {{- include "grafana.pod" . | nindent 6 }}
  volumeClaimTemplates:
  - metadata:
      name: storage
    spec:
...
```

# - 오픈소스 Helm 차트 분석 - Grafana (configmap.yaml)

## values.yaml

```
plugins:
   - digrich-bubblechart-panel
   - grafana-clock-panel


grafana.ini:
   paths:                              key
      data: /var/lib/grafana/data
      logs: /var/log/grafana
      plugins: /var/lib/grafana/plugins  value
      provisioning: /etc/grafana/provisioning
   analytics:
      check_for_updates: true
   log:
      mode: console
   grafana_net:
      url: https://grafana.net
```

## cm.yaml

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: {{ template "grafana.fullname" . }}
  ..
data:
{{- if .Values.plugins }}
  plugins: {{ join "," .Values.plugins }}
{{- end }}
  grafana.ini: |
{{- range $key, $value := index .Values "grafana.ini" }}
    [{{ $key }}]
   {{- range $elem, $elemVal := $value }}
    {{- if kindIs "invalid" $elemVal }}
    {{ $elem }} =
    {{- else if kindIs "string" $elemVal }}
    {{ $elem }} = {{ tpl $elemVal $ }}
    {{- else }}
    {{ $elem }} = {{ $elemVal }}
    {{- end }}
   {{- end }}
{{- end }}
```

grafana:
   ini:    ◄────    **{{ .Values.grafana.ini }}**

## _helpers.tpl

```
apiVersion: v1
kind: ConfigMap
metadata:
...
data:
  plugins: digrich-bubblechart-panel, grafana-clock-panel

grafana.ini: |
    [paths]
```

## values.yaml

```
plugins:
  - digrich-bubblechart-panel
  - grafana-clock-panel

grafana.ini:
  paths:                    /var/lib/{{ .Release.Name }}/data
    data: /var/lib/grafana/data
    logs: /var/log/grafana
    plugins: /var/lib/grafana/plugins
    provisioning: /etc/grafana/provisioning
  analytics:
    check_for_updates: true
  log:
    mode: console
  grafana_net:
    url: https://grafana.net
```

### Release

[ helm template my-grafana . ]

name

**grafana:** **ini:** ⬅ **{{ .Values.grafana.ini }}**
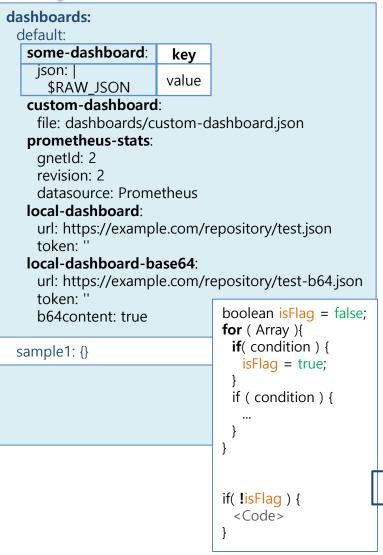
## cm.yaml

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: {{ template "grafana.fullname" . }}
  ..
data:
{{- if .Values.plugins }}
  plugins: {{ join "," .Values.plugins }}
{{- end }}
  grafana.ini: |
{{- range $key, $value := index .Values "grafana.ini" }}
    [{{ $key }}]
    {{- range $elem, $elemVal := $value }}
①  {{- if kindIs "invalid" $elemVal }}
    {{ $elem }} =
②  {{- else if kindIs "string" $elemVal }}
    {{ $elem }} = {{ tpl $elemVal $ }}
③  {{- else }}
    {{ $elem }} = {{ $elemVal }}
    {{- end }}
    {{- end }}
{{- end }}
```

## _helpers.tpl

```
apiVersion: v1
kind: ConfigMap
metadata:
...
data:
  plugins: digrich-bubblechart-panel, grafana-clock-panel

grafana.ini: |
    [paths]
    data =                /var/lib/my-grafana/data
    logs = /var/log/grafana
    plugins = /var/lib/grafana/plugins
    provisioning = /etc/grafana/provisioning

    [analytics]
    check_for_updates = true

    [grafana_net]
    url = https://grafana.net
    [log]
    mode = console
```

# - 오픈소스 Helm 차트 분석 - prometheus (dashboard-json-config.yaml)

## values.yaml

```
dashboards:
  default:
    some-dashboard:        key
      json: |
        $RAW_JSON          value
    custom-dashboard:
      file: dashboards/custom-dashboard.json
    prometheus-stats:
      gnetId: 2
      revision: 2
      datasource: Prometheus
    local-dashboard:
      url: https://example.com/repository/test.json
      token: ''
    local-dashboard-base64:
      url: https://example.com/repository/test-b64.json
      token: ''
      b64content: true

  sample1: {}
```

```
boolean isFlag = false;
for ( Array ){
  if( condition ) {
    isFlag = true;
  }
  if ( condition ) {
    ...
  }
}


if( !isFlag ) {
  <Code>
}
```

## dashboard-json-config.yaml

```
{{- if .Values.dashboards }}
{{ $files := .Files }}
{{- range $provider, $dashboards := .Values.dashboards }}
apiVersion: v1
kind: ConfigMap
metadata:
  name: {{ template "grafana.fullname" $ }}-dashboards-{{ $provider }}
  ...
{{- if $dashboards }}
data:
{{- $dashboardFound := false }}
{{- range $key, $value := $dashboards }}
  {{- if (or (hasKey $value "json") (hasKey $value "file")) }}
    {{- $dashboardFound = true }}
  {{ print $key | indent 2 }}.json:
    {{- if hasKey $value "json" }}
      |-
    {{ $value.json | indent 6 }}
    {{- end }}
    {{- if hasKey $value "file" }}
      {{ toYaml ( $files.Get $value.file ) | indent 4}}
    {{- end }}
  {{- end }}
{{- end }}
{{- if not $dashboardFound }}
  {}
{{- end }}
{{- end }}
---
{{- end }}

{{- end }}
```
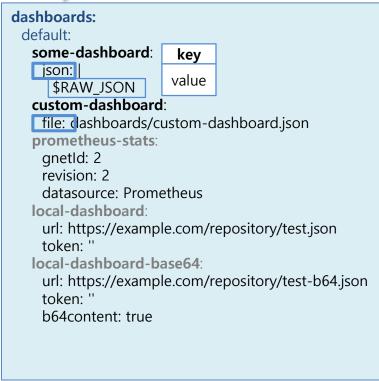
## _helpers.tpl

```
prometheus.fullname
grafana
```

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: grafana-dashboards-default
data:



---
apiVersion: v1
kind: ConfigMap
metadata:
  name: grafana-dashboards-sample1
data:
{}
---
```

## values.yaml

```
dashboards:
  default:
    some-dashboard:
      json:
        $RAW_JSON
    custom-dashboard:
      file: dashboards/custom-dashboard.json
    prometheus-stats:
      gnetId: 2
      revision: 2
      datasource: Prometheus
    local-dashboard:
      url: https://example.com/repository/test.json
      token: ''
    local-dashboard-base64:
      url: https://example.com/repository/test-b64.json
      token: ''
      b64content: true
```

| key |
|-----|
| value |

dashboards
└── custom-dashboard.json
        {}

## dashboard-json-config.yaml

```
{{- if .Values.dashboards }}
{{ $files := .Files }}
{{- range $provider, $dashboards := .Values.dashboards }}
apiVersion: v1
kind: ConfigMap
metadata:
  name: {{ template "grafana.fullname" $ }}-dashboards-{{ $provider }}
  ...
{{- if $dashboards }}
data:
{{- $dashboardFound := false }}
{{- range $key, $value := $dashboards }}
    {{- if (or (hasKey $value "json") (hasKey $value "file")) }}
    {{- $dashboardFound = true }}
    {{ print $key | indent 2 }}.json:
    {{- if hasKey $value "json" }}
        |-
      {{ $value.json | indent 6 }}
    {{- end }}
    {{- if hasKey $value "file" }}
      {{ toYaml ( $files.Get $value.file ) | indent 4}}
    {{- end }}
  {{- end }}
{{- end }}
{{- if not $dashboardFound }}
  {}
{{- end }}
{{- end }}
---
{{- end }}

{{- end }}
```

.Files.Get dashboards/custom-dashboard.json

## _helpers.tpl

```
prometheus.fullname
grafana
```

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: grafana-dashboards-default
data:
  some-dashboard.json:
    |-
    $RAW_JSON
  custom-dashboard.json:
  '{}'
---
```