Grafana Loki Monitoring Setup for Kubernetes Cluster

Installation of Loki-Stack

$ helm show values grafana/loki-stack > loki-stack-values.yaml

Modify the required changes and update the values file for loki-stack

test\_pod:

  image: bats/bats:v1.1.0

  pullPolicy: IfNotPresent

loki:

  enabled: true

  isDefault: true

  url: http://{{(include "loki.serviceName" .)}}:{{ .Values.loki.service.port }}

  persistence:

    type: pvc

    enabled: true

    size: 10Gi

    storageClassName: grafana

      #existingClaim: storage-loki-manual

      #existingSecretForConfig: loki

    #config:

    #ruler:

    #storage:

    #   type: local

    #   local:

    #     directory: /data/loki/rules

    # rule\_path: /tmp/loki/rules

    # alertmanager\_url: http://loki-prometheus-alertmanager:80

    # ring:

    #   kvstore:

    #     store: inmemory

    # enable\_api: true

    # enable\_alertmanager\_v2: true

  readinessProbe:

    httpGet:

      path: /ready

      port: http-metrics

    initialDelaySeconds: 45

  livenessProbe:

    httpGet:

      path: /ready

      port: http-metrics

    initialDelaySeconds: 45

  datasource:

    jsonData: {}

    uid: ""

promtail:

  enabled: true

  config:

    logLevel: info

    serverPort: 3101

    clients:

      - url: http://{{ .Release.Name }}:3100/loki/api/v1/push

fluent-bit:

  enabled: false

grafana:

  enabled: true

  sidecar:

    datasources:

      label: ""

      labelValue: ""

      enabled: true

      maxLines: 1000

  image:

    tag: 8.3.5

  admin:

    existingSecret: azure-aad

    userKey: admin-user

    passwordKey: admin-password

  env:

    GF\_SERVER\_DOMAIN: https://grafanadev.com

    GF\_SERVER\_ROOT\_URL: https://grafanadev.com/grafana/

    GF\_SERVER\_SERVE\_FROM\_SUB\_PATH: true

  envValueFrom:

    GF\_AUTH\_AZUREAD\_CLIENT\_ID:

         secretKeyRef:

           name: azure-aad

           key: client\_id

    GF\_AUTH\_AZUREAD\_CLIENT\_SECRET:

         secretKeyRef:

           name: azure-aad

           key: client\_secret

    GF\_SECURITY\_ADMIN\_USER:

         secretKeyRef:

           key: admin-user

           name: azure-aad

    GF\_SECURITY\_ADMIN\_PASSWORD:

         secretKeyRef:

           key: admin-password

           name: azure-aad

  persistence:

    type: pvc

    enabled: true

    size: 10Gi

    storageClassName: grafana

      #existingClaim: loki-grafana-manual

  grafana.ini:

     alerting:

        enabled: true

     auth.azuread:

        name: Azure AD

        enabled: true

        allow\_sign\_up: true

        scopes: openid email profile

        auth\_url:https://login.microsoftonline.com//<TENENT\_ID>/oauth2/v2.0/authorize

        token\_url:https://login.microsoftonline.com/<TENENT\_ID>/oauth2/v2.0/token

        allow\_assign\_grafana\_admin: false

prometheus:

  enabled: true

  isDefault: false

  url: http://{{ include "prometheus.fullname" .}}:{{ .Values.prometheus.server.service.servicePort }}{{ .Values.prometheus.server.prefixURL }}

  datasource:

    jsonData: {}

  alertmanager:

    persistentVolume:

      enabled: true

      size: 10Gi

      storageClass: grafana

  server:

    persistentVolume:

      enabled: true

      size: 10Gi

      storageClass: grafana

Create a **StorageClass** Resource for persistentence

apiVersion: storage.k8s.io/v1

kind: StorageClass

metadata:

  name: grafana

allowVolumeExpansion: true

reclaimPolicy: Retain

provisioner: kubernetes.io/azure-disk

volumeBindingMode: Immediate

Create Kubernetes Secret for Azure AAD client\_id, client\_secret, admin-user and admin-password.

Secret should be created before installation of loki-stack

apiVersion: v1

kind: Secret

metadata:

  name: azure-aad

data:

  client\_id: eflsdkflvlvslmlvxm'msllmv

  client\_secret: FZklafeklvxsglvcelrvsmcxv

  admin-user: YWRtaW4=

  admin-password: ZGV2b3Bz

if you created PV and PVC before the setup of loki-stack you can refer the existing PVC claim name in the values.yaml file

apiVersion: v1

kind: PersistentVolume

metadata:

  name: pv-loki-grafana

spec:

  capacity:

    storage: 10Gi

  accessModes:

    - ReadWriteOnce

  persistentVolumeReclaimPolicy: Retain

  storageClassName: grafana

  csi:

    driver: disk.csi.azure.com

    readOnly: false

    volumeHandle: /subscriptions/<SUBCRIPTION\_ID>/resourceGroups/<RESOURcE\_GROUP\_NAME>/providers/Microsoft.Compute/disks/<DISK\_NAME>

apiVersion: v1

kind: PersistentVolumeClaim

metadata:

  name: loki-grafana-manual

  labels:

    data: grafana-ui

spec:

  accessModes:

    - ReadWriteOnce

  storageClassName: grafana

  volumeMode: Filesystem

  volumeName: pv-loki-grafana

  resources:

    requests:

      storage: 10Gi

Create helm resources:

helm install name grafana/loki-stack –namespace <namespaceName> --values loki-stack-values.yaml

Grafana UI:

create Service type of LoadBalancer

apiVersion: v1

kind: Service

metadata:

  name: grafana-ui

  labels:

    app: grafana-svc

  annotations:

    service.beta.kubernetes.io/azure-dns-label-name: grafanalocal

spec:

  selector:

            app.kubernetes.io/instance: loki

            app.kubernetes.io/name: grafana

  type: LoadBalancer

  ports:

    - port: 80

      name: http

      targetPort: 3000

Go to Browser and hit the Public IP address or DNS name provided by cloud provider for Public IP Address.

Grafana is served as subpath i.e., /grafana

**User Credentials**

**Username**

kubectl get secrets  -n grafana-logging azure-aad -o jsonpath='{.data.admin-user}' | base64 -d

**password**

kubectl get secrets  -n grafana-logging azure-aad -o jsonpath='{.data.admin-password}' | base64 -d

Create App Registration for Grafana:

note down the client\_id and client\_secret

Graphical user interface, text, application, email

Description automatically generated



Choose platform web and Add redirect URI as below

<https://domain.com/grafana/login/azuread>

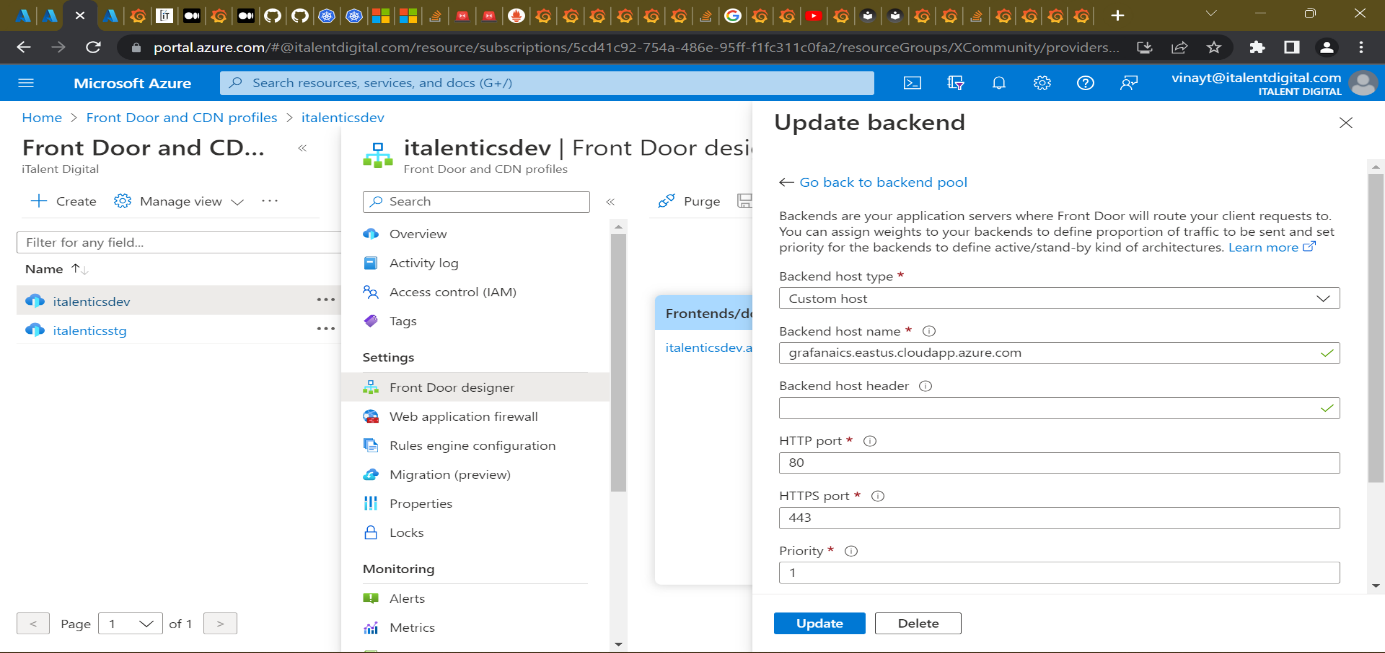
Configuration of Azure Front Door:

Add backend pool for Grafana

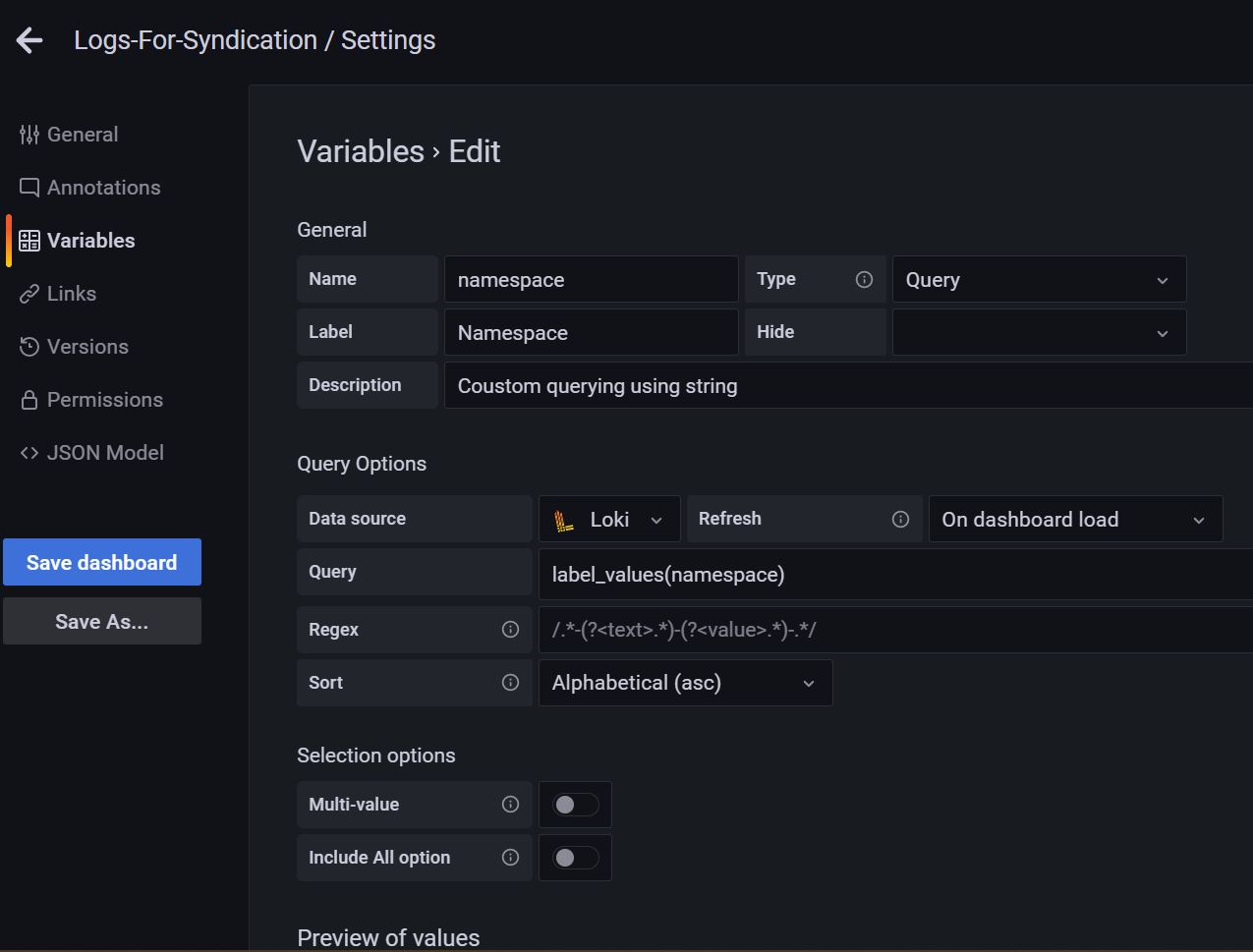
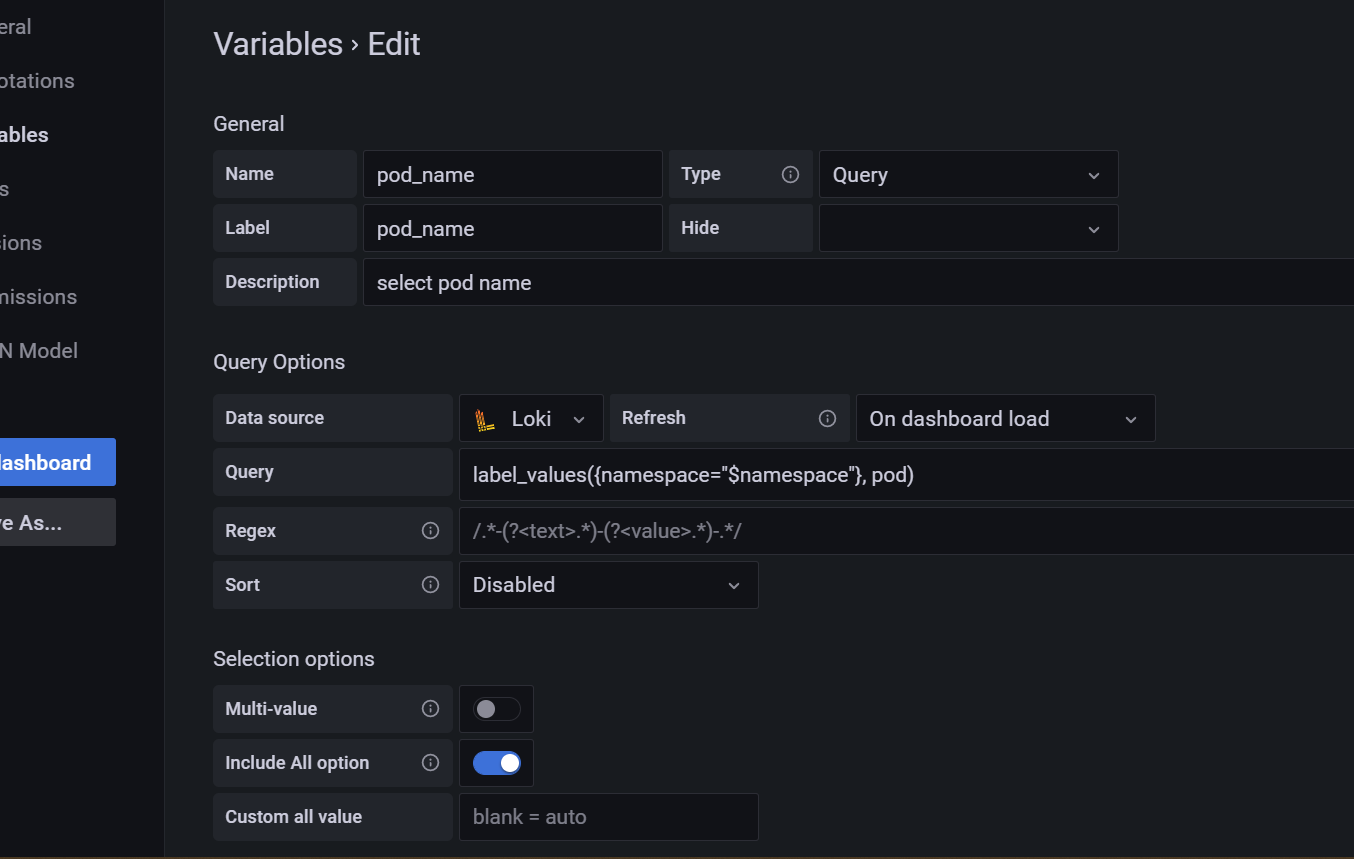
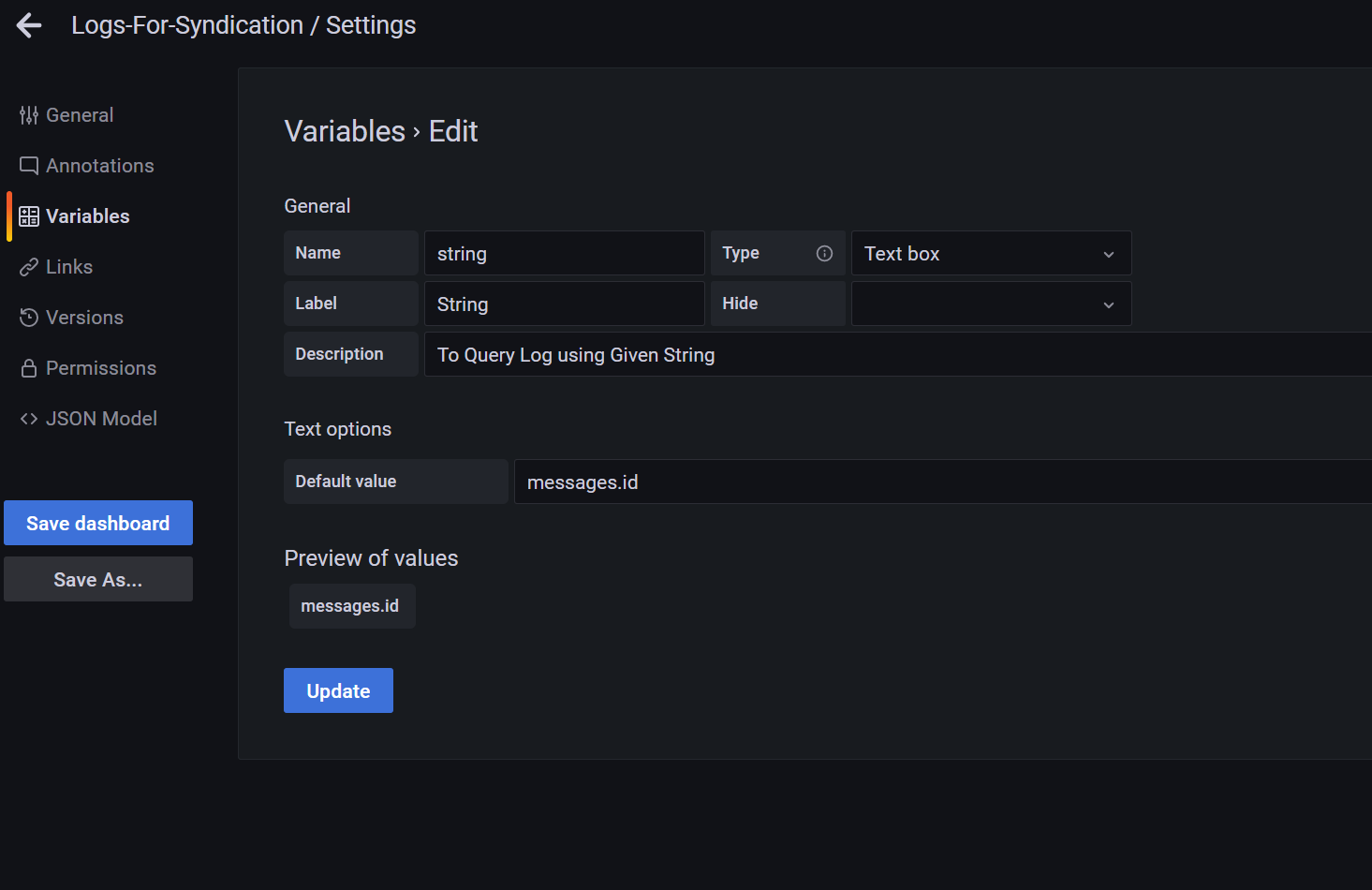
Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generatedGraphical user interface, text, application, email

Description automatically generated



Text

Description automatically generatedText

Description automatically generated