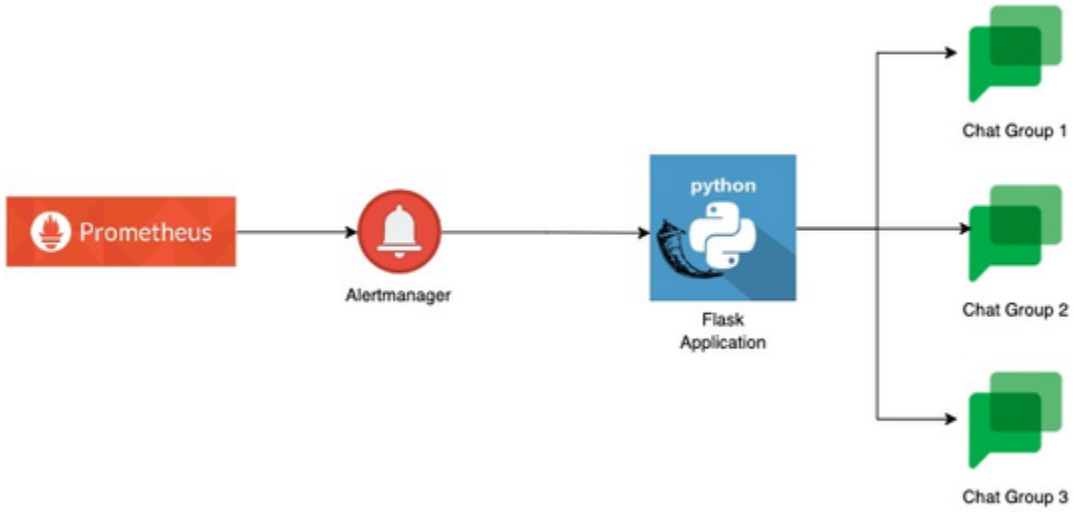
Alert manager alerts to GChat



1. Alerting rules in Prometheus servers send alerts to an Alertmanager
2. The Alertmanager Pushes notifications (webhook) to Python Flask Application.
3. In python application will receive the alerts in json format. Will parse the json alerts, categorise, send the alerts to respective group and assign to respective person.

Prometheus Configurations

Ex: - prometheus.yml

```
global:
  scrape_interval: 15s # Set the scrape interval to every 15 seconds.
Default is every 1 minute.
  evaluation_interval: 15s # Evaluate rules every 15 seconds. The
default is every 1 minute.

# Alertmanager configuration
alerting:
  alertmanagers:
    - static_configs:
        - targets:
           - 192.168.1.2:9093
rule_files:
    - "first_rules.yml"

scrape_configs:
  - job_name: node_10005
    static_configs:
      - targets: ['10.0.0.5:9100']
```

Ex: first_rules.yml

```
groups:
- name: node-exporter.rules
  rules:
  - alert: SaaS-Prod1(Common) WARNING Node Exporter Service down
    expr: up{job=~"node_.*"} == 0
    for: 4m
    labels:
      severity: "WARNING"
    annotations:
      summary: "Grafana URL http://10.0.0.15:3000/"
```

Alertmanager Configurations

Ex: alertmanager.yml

```
route:
  group_by: ['alertname']
  group_wait: 30s
  group_interval: 5m
  repeat_interval: 1h
  receiver: 'web.hook'
receivers:
  - name: 'web.hook'
    webhook_configs:
      - url: 'http://192.168.1.2:5001/'
inhibit_rules:
  - source_match:
      severity: 'critical'
    target_match:
      severity: 'warning'
    equal: ['alertname', 'dev', 'instance']
```

Python Dependencies

```
pip3 install flask gevent requests
```

Sample Flask Application

```
import json
from flask import Flask, request
from gevent.pywsgi import WSGIServer
from collections import namedtuple
import requests
import os

app = Flask(__name__)

@app.route('/', methods=['POST'])

def send():
    try:
        if request.method == 'POST':
            alerts = request.get_json()
            print(alerts)
            print ("********************")
    except Exception as e:
        return "Error", 500

    return "OK", 200

if __name__ == '__main__':
        app.run(host='0.0.0.0', port=80)
```

Run Flask application:

flask run --host=0.0.0.0 --port=5001

or

nohup python3 app.py > log.txt 2>&1 &

Ex: Gchat.py

```
#https://developers.google.com/chat/quickstart/incoming-bot-python

from json import dumps

from httplib2 import Http


def main():
    """Hangouts Chat incoming webhook quickstart."""
    url = 'https://chat.googleapis.com/v1/spaces/AAAAqIKsCCQ/messages?
key=AIzaSyDdI0hCZtE6vySjMm-
WEfRq3CPzqKqqsHI&token=VRQETmeboYrrz9zGdtKensL-Nlr51c0vUlG0HTLr7iA%3D'
    bot_message = {
        'text' : '```Hello from a Python script!```'}

    message_headers = {'Content-Type': 'application/json; charset=UTF-
8'}

    http_obj = Http()

    response = http_obj.request(
        uri=url,
        method='POST',
        headers=message_headers,
        body=dumps(bot_message),
    )

    print(response)

if __name__ == '__main__':
    main()
```

Other Tasks and Actions

1. Create different google chat groups
2. Create Incoming webhooks in gchat group
3. Storing the Alerts in Database for reporting / analysing purpose.
4. snooze button from in google chat group.