

# Cara Deploy Redash di GCP

Berikut catatan deployment Redash dari referensi

<https://redash.io/help/open-source/setup#-Google-Compute-Engine>

Buat image redash v8.0.0

```
gcloud compute images create "redash-8-0-0" \  
  --source-uri gs://redash-images/redash.8.0.0-b32245-1.tar.gz \  
  --project=project-name
```

Buat VM dengan image redash tersebut

```
gcloud compute instances create redash-vm \  
  --machine-type=e2-medium \  
  --zone=asia-southeast2-b \  
  --image redash-8-0-0 \  
  --project=project-name
```

## Aktifkan SSL Redash

Login dengan ssh ke VM

```
gcloud compute ssh redash-vm --project=project-name
```

- Switch to the root user (`sudo su`).
- Create a folder named `nginx` in `/opt/redash`.
- Create in the `nginx` folder two additional folders: `certs` and `certs-data`.
- Create the file `/opt/redash/nginx/nginx.conf` and place the following in it:

```
upstream redash {  
    server redash:5000;  
}  
  
server {  
    listen      80;  
    #listen [::]:80;  
    server_name redash.mydomain.com;  
  
    location ^~ /ping {  
        proxy_set_header Host $http_host;  
        proxy_set_header X-Real-IP $remote_addr;  
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;  
        proxy_set_header X-Forwarded-Proto $http_x_forwarded_proto;  
  
        proxy_pass      http://redash;  
    }  
}
```

```

location / {
    rewrite ^ https://$host$request_uri? permanent;
}

location ^~ /.well-known {
    allow all;
    root /data/letsencrypt/;
}
}

```

- Edit `/opt/redash/docker-compose.yml` and update the nginx service to look like the following:

```

nginx:
  image: nginx:latest
  ports:
    - "80:80"
    - "443:443"
  depends_on:
    - server
  links:
    - server:redash
  volumes:
    - /opt/redash/nginx/nginx.conf:/etc/nginx/conf.d/default.conf
    - /opt/redash/nginx/certs:/etc/letsencrypt
    - /opt/redash/nginx/certs-data:/data/letsencrypt
  restart: always

```

- Update Docker Compose: `docker-compose up -d`.
- Generate certificates: (remember to change the domain name)

```

docker run -it --rm \
  -v /opt/redash/nginx/certs:/etc/letsencrypt \
  -v /opt/redash/nginx/certs-data:/data/letsencrypt \
  deliverous/certbot \
  certonly \
  --webroot --webroot-path=/data/letsencrypt \
  -d redash.mydomain.com

```

- Assuming the previous step was succesful, update the nginx config to include the SSL configuration:

```

upstream redash {
    server redash:5000;
}

server {
    listen 80;
    listen [::]:80;
    server_name redash.mydomain.com;

    location ^~ /ping {

```

```
    proxy_set_header Host $http_host;
    proxy_set_header X-Real-IP $remote_addr;
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    proxy_set_header X-Forwarded-Proto $scheme;

    proxy_pass      http://redash;
}

location / {
    rewrite ^ https://$host$request_uri? permanent;
}

location ^~ /.well-known {
    allow all;
    root /data/letsencrypt/;
}
}

server {
    listen      443          ssl http2;
    listen [::]:443      ssl http2;
    server_name      redash.mydomain.com;

    add_header      Strict-Transport-Security "max-age=31536000"
always;

    ssl_session_cache      shared:SSL:20m;
    ssl_session_timeout    10m;

    ssl_protocols          TLSv1 TLSv1.1 TLSv1.2;
    ssl_prefer_server_ciphers on;
    ssl_ciphers
"ECDH+AESGCM:ECDH+AES256:ECDH+AES128:!ADH:!AECDH:!MD5";

    ssl_stapling           on;
    ssl_stapling_verify    on;
    resolver                8.8.8.8 8.8.4.4;

    ssl_certificate
/etc/letsencrypt/live/redash.mydomain.com/fullchain.pem;
    ssl_certificate_key
/etc/letsencrypt/live/redash.mydomain.com/privkey.pem;
    ssl_trusted_certificate
/etc/letsencrypt/live/redash.mydomain.com/chain.pem;

    access_log            /dev/stdout;
    error_log              /dev/stderr info;

    # other configs

    location / {
```

```
proxy_set_header Host $http_host;
proxy_set_header X-Real-IP $remote_addr;
proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
proxy_set_header X-Forwarded-Proto $scheme;

proxy_pass http://redash;
}
}
```

- Restart nginx: `docker-compose restart nginx`.
- All done, your Redash instance should be available via HTTPS now.

To renew the certificate in the future, you can use the following command:

```
$ docker run -t --rm -v /opt/redash/nginx/certs:/etc/letsencrypt \
-v /opt/redash/nginx/certs-data:/data/letsencrypt \
deliverous/certbot renew --webroot --webroot-
path=/data/letsencrypt

$ docker-compose kill -s HUP nginx
```

From:

<https://wiki.samsul.web.id/> - **Samsul Maarif**

Permanent link:

<https://wiki.samsul.web.id/linux/Cara.Deploy.Redash.di.GCP>

Last update: **2022/12/28 16:44**

