

# Free Prometheus

Docker + Heroku

# Hi I'm Edward Or Wenlock on GitHub.com



Back in the office

**Edward Raigosa**

wenlock

 Software Engineer ,  UofSC  
Gamecock - BS EE, Loves all things  
Containers  , Agile & DevOps  
practices  , Volleyball  ,  
Swim  , & Camping 

 GitHub **STAFF**

 Roseville, CA

 <https://twitter.com/wenlockCA>

**UCDAVIS**

Continuing and  
Professional Education



Coming Soon - DevOps  
Continuous Integration  
classes on Coursera

# 2 repos

<https://github.com/ucdavis>

## WEB1066-probot-hello

UC Davis Continuing and Professional Education WEB1066 Class 

 JavaScript  4  ISC Updated on Dec 7, 2018

## WEB1066-prom

Metrics and Analytics with Prometheus

 Shell  1  ISC Updated on Dec 8, 2018

Repo 1 -  
WEB1066-probot-hello

Check the  
**master\_docker** branch

Repo 2 - WEB1066-prom

Check the  
**master\_docker** branch

# The sequence of commands to publish

```
heroku container:login
```

```
heroku container:push web
```

```
heroku container:release web
```

Deployment method



Heroku Git  
Use Heroku CLI



GitHub  
Connect to GitHub



Container Registry  
Use Heroku CLI

# The Dockerfile

25 lines (18 sloc) | 634 Bytes

Raw

Blame

History

```
1 FROM prom/prometheus
2
3 USER root
4 ADD bin/curl-7.30.0.ermine.tar.bz2 .
5
6 RUN mv curl-7.30.0.ermine/curl.ermine /bin/curl \
7     && rm -rf curl-7.30.0.ermine
8
9 # Capture SIGTERM from an orchestrator so we can reliably use entrypoint.sh
10 # Using tini https://github.com/krallin/tini
11 ENV TINI_VERSION v0.18.0
12 ADD https://github.com/krallin/tini/releases/download/${TINI_VERSION}/tini-static /bin/tini
13 RUN chmod +x /bin/tini
14
15 USER nobody
16 ADD config/prometheus.yml /etc/prometheus/
17 ADD src/entrypoint.sh /bin/entrypoint.sh
18
19 ENTRYPOINT ["/bin/tini", \
20             "--", \
21             "sh", \
22             "/bin/entrypoint.sh" ]
23
24 CMD [ "" ]
```



# Prompted to login to Heroku

```
Edwards-Mac:WEB1066-prom wenlock$ heroku login  
heroku: Press any key to open up the browser to login or q to exit: █
```

# Start a Heroku Project

```
Edwards-Mac:WEB1066-prom wenlock$ heroku apps:create wenlockca-prom  
Creating ● wenlockca-prom... done  
https://wenlockca-prom.herokuapp.com/ | https://git.heroku.com/wenlockca-prom.git  
Edwards-Mac:WEB1066-prom wenlock$ █
```

# Use env values for config settings


```
$ heroku config:set PROBOT_HOST=probot-wenlockca.herokuapp.com
```

# Replace the setting on startup

```
Edwards-Mac:WEB1066-prom wenlock$ cat src/entrypoint.sh
#!/bin/sh
#
# we use an entrypoint script here with PORT setting so that
# we can run this on heroku using containers
export SERVER_PORT="${PORT:-8080}"
echo "Arguments are $*"
echo "Working with ${SERVER_PORT}"


#
# This is the probot host that is collecting our metrics
echo "Configure PROBOT_HOST => ${PROBOT_HOST}"
sed -i "s/{{PROBOT_HOST}}/${PROBOT_HOST}/g" /etc/prometheus/prometheus.yml

echo "Starting up server"
"/bin/prometheus" \
  --storage.tsdb.path=/prometheus \
  --web.console.libraries=/etc/prometheus/console_libraries \
  --web.console.templates=/etc/prometheus/consoles \
  --config.file=/etc/prometheus/prometheus.yml \
  --web.listen-address=0.0.0.0:${SERVER_PORT}
```



# Running: heroku container:push web

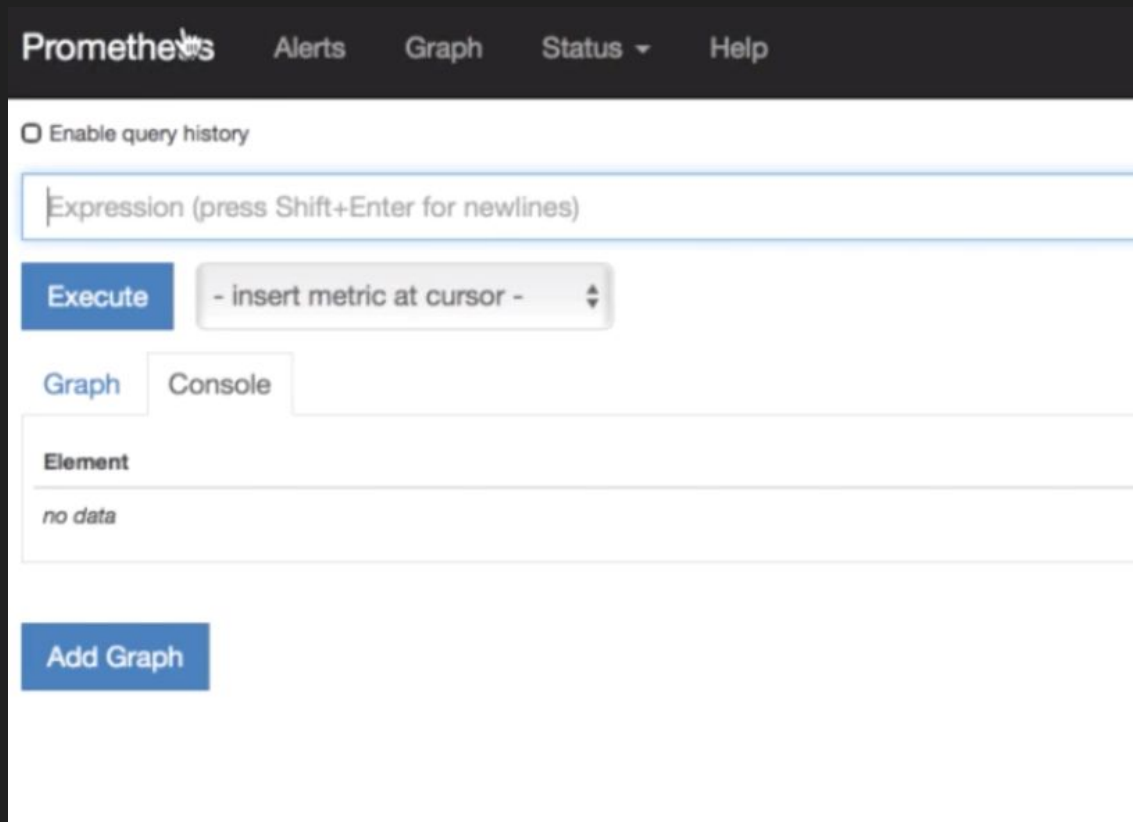
```
---> bb20c6a8b6b6
Step 10/12 : ADD src/entrypoint.sh /bin/entrypoint.sh
---> Using cache
---> 44b5e147347e
Step 11/12 : ENTRYPOINT ["/bin/tini",          "--",          "sh",
" ]
---> Using cache
---> 2272d90879e1
Step 12/12 : CMD          [ "" ]
---> Using cache
---> ffb7e970bfa8
Successfully built ffb7e970bfa8
Successfully tagged registry.heroku.com/wenlockca-prom/web:latest
=== Pushing web (/Users/wenlock/Documents/GitHub/WEB1066-prom/Dockerfile)
The push refers to repository [registry.heroku.com/wenlockca-prom/web]
ba745fcd814: Preparing
aff7d17b7b38: Preparing
56589773f643: Preparing
eaa7f16a9747: Preparing
14a87a1309da: Preparing
```



# Release: heroku container:release web

```
d3a2b908d5ad: Pushed
8a788232037e: Pushed
latest: digest: sha256:015c8fc88b21241a357f4edf8de5e21b1b6de252d5049107a2b7fc27a5c5084d size: 3454
Your image has been successfully pushed. You can now release it with the 'container:release' command.
Releasing images web to wenlockca-prom... done
Edwards-Mac:WEB1066-prom wenlock$
```

# Tada, free Prometheus



The screenshot displays the Prometheus web interface. At the top, a dark navigation bar contains the 'Prometheus' logo and links for 'Alerts', 'Graph', 'Status', and 'Help'. Below this, a white panel features a toggle for 'Enable query history' and a text input field for the query expression. The input field contains the placeholder text 'Expression (press Shift+Enter for newlines)'. To the left of the input is a blue 'Execute' button, and to the right is a dropdown menu with the text '- insert metric at cursor -'. Below the input field, there are two tabs: 'Graph' (which is selected) and 'Console'. Under the 'Graph' tab, there is a section labeled 'Element' which currently displays 'no data'. At the bottom left of the interface is a blue button labeled 'Add Graph'.

Prometheus Alerts Graph Status Help

☐ Enable query history

Expression (press Shift+Enter for newlines)

Execute - insert metric at cursor -

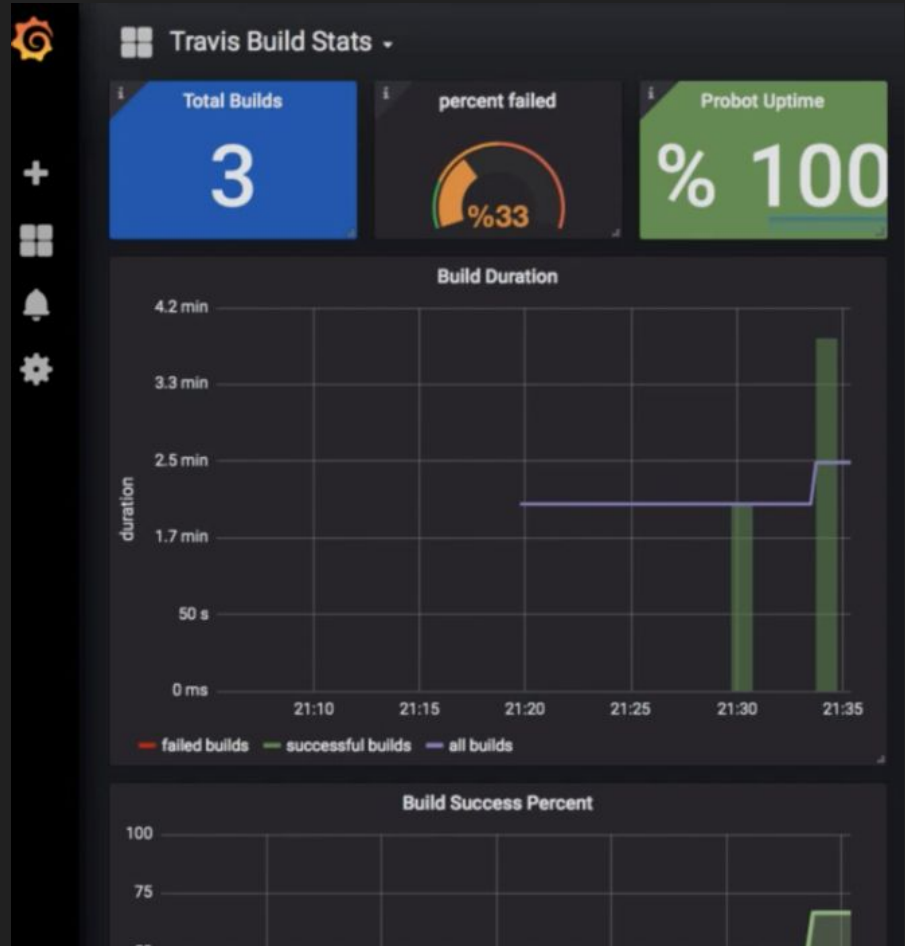
Graph Console

Element

no data

Add Graph

# Putting it together





Thank You