

# dracula - time series graphing server

graf(ana)-dracula .. get it? ;)

- intended to capture time series of performance data during benchmark runs
- grafite and grafana

## setup

### graphite

isntalled in docker container:

install docker

```
apt-get install apt-transport-https ca-certificates curl gnupg-agent software-properties-common
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu \
$(lsb_release -cs) \
stable"
apt update
apt-get install docker-ce docker-ce-cli containerd.io
cat >> /etc/docker/daemon.json <<EOF
{
  "log-driver": "syslog"
}
EOF
systemctl restart docker
docker run hello-world
```

install and run grafite

```
docker run -d --name graphite --restart=always -p 80:80 -p 2003-2004:2003-2004 -p 2023-2024:2023-2024 -p 8125:8125/udp -p 81
```

these are the pors that where mapped:

Host	Container	Service
80	80	nginx
2003	2003	carbon receiver - plaintext
2004	2004	carbon receiver - pickle
2023	2023	carbon aggregator - plaintext
2024	2024	carbon aggregator - pickle
8080	8080	Graphite internal gunicorn port (without Nginx proxying).
8125	8125	statsd
8126	8126	statsd admin

### grafana

```
add-apt-repository "deb https://packages.grafana.com/oss/deb stable main"
wget -q -O - https://packages.grafana.com/gpg.key | sudo apt-key add -
apt-get update
apt-get install grafana
systemctl enable --now grafana-server
systemctl status grafana-server
```

### Telegraf

is a tool from the influxDB people that has gathering plugins and output plugs that can in combination query lots of different data sources and then feed the data to different databases such as Graphite. I used it to query our APC UPS and get the temperature readings from the UPS via SNMP

installation is easy via deb package:

```
wget https://dl.influxdata.com/telegraf/releases/telegraf_1.11.3-1_amd64.deb
dpkg -i telegraf_1.11.3-1_amd64.deb
```

get help via

```
telegraf --help
```

a pretty well explained example config was installed with the deb package and can always be recreated using the `telegraf config` command. I simply edited the file in `/etc/telegraf/telegraf.conf`

now run telegraf without any further options in the foreground or run it with `--test` to check the config first.

a systemd service is also installed with the deb file.

## graphs.dalco.ch

is the publicly reachable site over which we can publish grafana snapshots to customers and suppliers when we are benchmarking or debugging stuff.

originally used for AMD Gigabyte ROME debugging

same setup as dracula, only a bit more secure, grafana is running on port 443 with ssl encryption provided by letsencrypt using the certbot.

port 80 and 443 are open to the outside world, while port 80 is required for certbot in standalone mode, it's not being used otherwise.

at least for the AMD / Gigabyte tests port **2113** was open too (redirected to the default 2003 port for sending data), so that their systems could push metrics to the graphite instance that is running on the server.

dracula.txt · Last modified: 09.01.2020 17:32 by psuter