

carbon, graphite and grafana

carbon

install carbon-cache service

```
apt-get install graphite-carbon
```

extend retentions

[/etc/carbon/storage-schemas.conf](#)

```
[carbon]
pattern = ^carbon\.
retentions = 60:90d

[default]
pattern = .*
retentions = 30s:7d,5m:30d,10m:1y,1d:10y
```

after changing *schemas.conf older *wsp has to be changed with whisper-resize

```
for i in `find /var/lib/graphite/whisper/ha/ -name "*wsp"`; do
  whisper-resize $i 30s:7d 5m:30d 10m:1y 1d:10y
done
```

graphite-web

web interface

```
apt-get install graphite-web
```

```
# choose user and password
python /usr/lib/python2.7/dist-packages/graphite/manage.py syncdb
```

adjust user and password and SECRET_KEY (with random string) in [/etc/graphite/local_settings.py](#)

nginx

```
apt install uwsgi uwsgi-plugin-python
```

[/etc/uwsgi/apps-enabled/graphite.ini](#)

```
[uwsgi]

vacuum = true
master = true
processes = 2
pidfile = /tmp/uwsgi.pid
socket = /tmp/uwsgi.sock
chmod-socket = 666
gid = _graphite
uid = _graphite
chdir = /usr/share/graphite-web
wsgi-file = graphite.wsgi
pymodule-alias =
graphite.local_settings=/etc/graphite/local_settings.py
buffer-size = 65536
plugin = python
```

```
systemctl restart uwsgi
```

[/etc/nginx/sites-enabled/graphite](#)

```
upstream graphite {
    server unix:///tmp/uwsgi.sock;
}

server {
    listen 9002;
    server_name localhost;

    access_log /var/log/nginx/graphite-access.log;
    error_log /var/log/nginx/graphite-error.log;

    root /usr/share/graphite-web/static;

    location / {
        add_header Access-Control-Allow-Origin "*";
        add_header Access-Control-Allow-Methods "GET, OPTIONS";
        add_header Access-Control-Allow-Headers "origin, authorization,
accept";

        uwsgi_pass graphite;
        include /etc/nginx/uwsgi_params;
    }

    location /media {
        # This makes static media available at the /media/ url. The
        # media will continue to be available during site downtime,
        # allowing you to use styles and images in your maintenance
page.
```

```
alias /usr/lib/python2.7/dist-  
packages/django/contrib/admin/media;  
}  
}
```

grafana

install

```
echo "deb https://packagecloud.io/grafana/stable/debian/ jessie main" >  
/etc/apt/sources.list.d/grafana.list  
curl https://packagecloud.io/gpg.key | sudo apt-key add -  
apt update  
apt install -y grafana  
systemctl enable grafana-server  
systemctl start grafana-server
```

Login to <http://localhost:3000> (admin/admin)

[add datasource](#)

Data Sources

Edit data source

Config | Dashboards

Name	ha	<input type="checkbox"/>	Default	<input type="checkbox"/>
Type	Graphite			

Http settings

Url	http://localhost:9002	<input type="checkbox"/>		
Access	proxy	<input type="checkbox"/>		
Http Auth	Basic Auth	<input type="checkbox"/>	With Credentials	<input type="checkbox"/>

Success
Data source is working

Save & Test **Delete** Cancel

From: <https://wiki.csgalileo.org/> - Galileo Labs

Permanent link: <https://wiki.csgalileo.org/projects/internetofthings/graphite?rev=1488027790>

Last update: 2017/02/25 14:03

