

# carbon, graphite and grafana

## carbon

install carbon-cache service

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

enable boot startup in `/etc/default/graphite-carbon`

enable `ENABLE_LOGROTATION` in `/etc/carbon/carbon.conf`

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`

```
bash> 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

```
bash> apt-get install graphite-web
```

adjust `SECRETKEY` (with random string) and `TIMEZONE` in `/etc/graphite/local_settings.py`

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

==== nginx ====

```
bash> apt install uwsgi uwsgi-plugin-python
```

```
ini /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.localsettings=/etc/graphite/local_settings.py
buffer-size = 65536 plugin = python
```

```
bash> systemctl restart uwsgi
```

```
txt /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
```

```
bash> 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=1489132197>

Last update: **2017/03/10 08:49**

