



Fiche technique

Telegraf sur Raspberry Pi 5

Table des matières

1	Présentation	1
2	Matériel Requis	1
3	Installation de Telegraf	2
4	Configuration de Telegraf	3
5	Installation d’InfluxDB	3
6	Configuration d’InfluxDB pour Telegraf	4
7	Installation de Grafana	4
8	Configuration de Grafana	5
9	Première vizualisation graphique	6

Présentation

Cette partie détaille l'installation et la configuration de **Telegraf**, **InfluxDB** et **Grafana** sur une Raspberry Pi, afin d'avoir un aperçu des performances de celle-ci.

Matériel Requis

- Raspberry Pi (modèle récent recommandé)
- Carte microSD (16GB ou plus) avec Raspberry Pi OS
- Connexion internet (requis)

Installation de Telegraf

1. Télécharger la clé GPG :

```
wget https://repos.influxdata.com/influxdata-archive_compat.key -O influxdata.key
```

2. Convertir la clé au format GPG :

```
gpg --dearmor influxdata.key
```

3. Déplacer la clé :

```
sudo mv influxdata.key.gpg /usr/share/keyrings/influxdata-archive_compat.gpg
```

4. Ajouter le dépôt InfluxData :

```
echo "deb [signed-by=/usr/share/keyrings/influxdata-archive_compat.gpg] https://repos.influxdata.com/debian $(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/influxdata.list
```

5. Mettre à jour les paquets :

```
sudo apt-get update
```

6. Installer Telegraf :

```
sudo apt-get install telegraf
```

7. Vérifier la version :

```
telegraf --version
```

8. Démarrer et vérifier que Telegraf fonctionne correctement :

```
sudo systemctl start telegraf
sudo systemctl enable telegraf
sudo systemctl status telegraf
```

On obtient :

```
pi@raspberrypi:~$ sudo systemctl status telegraf
● telegraf.service - Telegraf
   Loaded: loaded (/lib/systemd/system/telegraf.service; enabled; preset: enabled)
   Active: active (running) since Thu 2025-04-24 17:38:34 CEST; 22h ago
     Docs: https://github.com/influxdata/telegraf
   Main PID: 801 (telegraf)
      Tasks: 23 (limit: 1579)
         CPU: 52.21s
    CGroup: /system.slice/telegraf.service
            └─ 801 /usr/bin/telegraf -config /etc/telegraf/telegraf.conf -config-directory /etc/telegraf/telegraf.d
               1200 /usr/bin/dbus-daemon -syslog -fork -print-pid 4 -print-address 0 --session

Apr 25 15:48:37 raspberrypi telegraf[801]: mem,host=raspberrypi,write_bytes=1024001,total=10350457281,committed_size=47796633601,inactive=44809429801,swap_total=5388668161,vmalloc_chunk=0,page_tables=105057281,shared=1040588801,vmalloc_total=
Apr 25 15:48:37 raspberrypi telegraf[801]: diskio,host=raspberrypi,name=mmcblk0 io_write=42.58022033989394,io_write=3.032203398305087,reads=223801,write=206041,read_bytes=1480888841,io_time=2000781,weighted_io_time=42142261,ioops_in=0
Apr 25 15:48:37 raspberrypi telegraf[801]: diskio,host=raspberrypi,name=mmcblk0p1 read_time=15401,write_time=5041,weighted_io_time=21051,ioops_in_progress=01,merged_writes=01,reads=2371,io_time=17441,merged_reads=10701,io_util=0,writes=31
Apr 25 15:48:37 raspberrypi telegraf[801]: diskio,host=raspberrypi,name=mmcblk0p2 read_bytes=14706254081,write_bytes=472219302,read_time=3046071,io_time=2030401,merged_reads=93251,io_util=4.04,reads=229321,write_time=39058
Apr 25 15:48:37 raspberrypi telegraf[801]: system,host=raspberrypi load1l=0.17,n_cpu=4,n_users=21,n_unique_users=11,load1l=0.21,load5l=0.22,1745588910000000000
Apr 25 15:48:37 raspberrypi telegraf[801]: system,host=raspberrypi uptime_format=" 1.33" 1745588910000000000
Apr 25 15:48:37 raspberrypi telegraf[801]: swap,host=raspberrypi used_percent=1.02483463161874,total=5388668161,used=64020401,free=5304647681 1745588910000000000
Apr 25 15:48:37 raspberrypi telegraf[801]: swap,host=raspberrypi io=3056241,io=42718241 1745588910000000000
Apr 25 15:48:37 raspberrypi telegraf[801]: processes,host=raspberrypi dead=01,paging=01,blocked=01,total_threads=4401,idle=801,zombies=01,stopped=01,running=01,sleeping=1311,total=2261,unknown=01 1745588910000000000
```

Configuration de Telegraf

1. Modifier la configuration pour test :

```
sudo nano /etc/telegraf/telegraf.conf
```

Ajouter à la fin :

```
[[outputs.file]]
  files = ["stdout", "/tmp/telegraf_output.txt"]
  data_format = "influx"
```

2. Relancer et vérifier :

```
sudo systemctl restart telegraf
cat /tmp/telegraf_output.txt
```

Installation d'InfluxDB

1. Télécharger et ajouter la clé :

```
wget https://repos.influxdata.com/influxdata-archive_compat.key -O
influxdata.key
gpg --dearmor influxdata.key
sudo mv influxdata.key.gpg /usr/share/keyrings/influxdata-
archive_compat.gpg
```

2. Ajouter le dépôt officiel InfluxData :

```
echo "deb [signed-by=/usr/share/keyrings/influxdata-archive_compat
.gpg] https://repos.influxdata.com/debian $(lsb_release -cs)
stable" | sudo tee /etc/apt/sources.list.d/influxdata.list
```

3. Mise à jour :

```
sudo apt update
```

4. Installer InfluxDB :

```
sudo apt install influxdb
```

5. Activer le démarrage automatique :

```
sudo systemctl enable influxdb
sudo systemctl start influxdb
```

6. Vérifier si InfluxDB fonctionne :

```
sudo systemctl status influxdb
```

On obtient :

```
pi@raspberrypi:~$ sudo systemctl status influxdb
● influxdb.service - InfluxDB is an open-source, distributed, time series database
   Loaded: loaded (/lib/systemd/system/influxdb.service; enabled; preset: enabled)
   Active: active (running) since Thu 2025-04-24 17:38:34 CEST; 22h ago
     Docs: https://docs.influxdata.com/influxdb/
   Process: 889 ExecStart=/usr/share/influxdb/scripts/influxd-systemd-start.sh (code=exited, status=0/SUCCESS)
   Main PID: 1071 (influxd)
     Tasks: 13 (limit: 1579)
        CPU: 34.081s
   CGroup: /system.slice/influxdb.service
           └─1071 /usr/bin/influxd -config /etc/influxdb/influxdb.conf

Apr 25 15:48:37 raspberrypi influxd-systemd-start.sh[1071]: [httpd] :1.1 - [25/Apr/2025:15:48:37 +0200] "POST /write?db=telegraf HTTP/1.1" 204 0 "-" "Telegraf/1.34.2 Go/1.24.1" 20f6d4aa-21dc-11f0-823e-dca632f72bab 39218
Apr 25 15:48:47 raspberrypi influxd-systemd-start.sh[1071]: [httpd] :1.1 - [25/Apr/2025:15:48:47 +0200] "POST /write?db=telegraf HTTP/1.1" 204 0 "-" "Telegraf/1.34.2 Go/1.24.1" 29e6aa88-21dc-11f0-823e-dca632f72bab 38867
Apr 25 15:49:58 raspberrypi influxd-systemd-start.sh[1071]: [httpd] :1.1 - [25/Apr/2025:15:49:57 +0200] "POST /write?db=telegraf HTTP/1.1" 204 0 "-" "Telegraf/1.34.2 Go/1.24.1" 2ce1b222-21dc-11f0-823e-dca632f72bab 950880
Apr 25 15:50:07 raspberrypi influxd-systemd-start.sh[1071]: [httpd] :1.1 - [25/Apr/2025:15:50:07 +0200] "POST /write?db=telegraf HTTP/1.1" 204 0 "-" "Telegraf/1.34.2 Go/1.24.1" 32d7af68-21dc-11f0-823e-dca632f72bab 12253
Apr 25 15:50:17 raspberrypi influxd-systemd-start.sh[1071]: [httpd] :1.1 - [25/Apr/2025:15:50:17 +0200] "POST /write?db=telegraf HTTP/1.1" 204 0 "-" "Telegraf/1.34.2 Go/1.24.1" 38cd0809-21dc-11f0-823e-dca632f72bab 172450
Apr 25 15:50:27 raspberrypi influxd-systemd-start.sh[1071]: [httpd] :1.1 - [25/Apr/2025:15:50:27 +0200] "POST /write?db=telegraf HTTP/1.1" 204 0 "-" "Telegraf/1.34.2 Go/1.24.1" 39c3a0c8-21dc-11f0-823e-dca632f72bab 23297
Apr 25 15:50:37 raspberrypi influxd-systemd-start.sh[1071]: [httpd] :1.1 - [25/Apr/2025:15:50:37 +0200] "POST /write?db=telegraf HTTP/1.1" 204 0 "-" "Telegraf/1.34.2 Go/1.24.1" 44b9a180-21dc-11f0-823e-dca632f72bab 18522
Apr 25 15:50:47 raspberrypi influxd-systemd-start.sh[1071]: [httpd] :1.1 - [25/Apr/2025:15:50:47 +0200] "POST /write?db=telegraf HTTP/1.1" 204 0 "-" "Telegraf/1.34.2 Go/1.24.1" 4aafad5d-21dc-11f0-823f-dca632f72bab 36117
Apr 25 15:50:57 raspberrypi influxd-systemd-start.sh[1071]: [httpd] :1.1 - [25/Apr/2025:15:50:57 +0200] "POST /write?db=telegraf HTTP/1.1" 204 0 "-" "Telegraf/1.34.2 Go/1.24.1" 50a590f2-21dc-11f0-8240-dca632f72bab 16830
Apr 25 15:51:07 raspberrypi influxd-systemd-start.sh[1071]: [httpd] :1.1 - [25/Apr/2025:15:51:07 +0200] "POST /write?db=telegraf HTTP/1.1" 204 0 "-" "Telegraf/1.34.2 Go/1.24.1" 569badd1-21dc-11f0-8241-dca632f72bab 30379
pi@raspberrypi:~$
```

Configuration d'InfluxDB pour Telegraf

1. Se connecter à InfluxDB :

```
influx
```

2. Créer une base de données :

```
CREATE DATABASE telegraf
```

3. Quitter :

```
exit
```

4. Modifier le fichier `telegraf.conf` :

```
[[outputs.influxdb]]
  urls = ["http://localhost:8086"]
  database = "telegraf"
```

5. Redémarrer Telegraf :

```
sudo systemctl restart telegraf
```

Installation de Grafana

1. Ajouter la clé et le dépôt :

```
wget -q -O - https://packages.grafana.com/gpg.key | sudo apt-key
add -
echo "deb https://packages.grafana.com/oss/deb stable main" | sudo
tee /etc/apt/sources.list.d/grafana.list
```

2. Mise à jour :

```
sudo apt-get update
```

3. Installer Grafana :

```
sudo apt-get install grafana
```

4. Démarrer Grafana :

```
sudo systemctl start grafana-server
```

5. Activer le démarrage automatique :

```
sudo systemctl enable grafana-server
```

6. Vérifier si Grafana tourne bien :

```
sudo systemctl status grafana-server
```

On obtient, le résultat suivant :

```
grafana-server.service - Grafana instance
Loaded: loaded (/lib/systemd/system/grafana-server.service; enabled; preset: enabled)
Active: active (running) since Thu 2025-04-24 17:38:34 CEST; 22h ago
Docs: http://docs.grafana.org
Main PID: 1296 (grafana)
Tasks: 22 (limit: 1579)
CPU: 46.739s
CGroup: /system.slice/grafana-server.service
└─1296 /usr/share/grafana/bin/grafana server --config=/etc/grafana/grafana.ini --pidfile=/run/grafana/grafana-server.pid --packaging=deb cfg:default.paths.logs=/var/log/grafana cfg:default.paths.data=/var/lib/grafana cfg:default.paths.plugins=/var/lib/grafana/plugins

Apr 25 15:15:56 raspberrypi grafana[1296]: logger:cleanup t=2025-04-25T15:15:56.618138816+02:00 level=info msg="Completed cleanup jobs" duration=159.441142ms
Apr 25 15:15:57 raspberrypi grafana[1296]: logger:plugins.update.checker t=2025-04-25T15:15:57.251069055+02:00 level=info msg="Update check succeeded" duration=68.505344ms
Apr 25 15:17:50 raspberrypi grafana[1296]: logger:infra.usagestat t=2025-04-25T15:17:50.557091739+02:00 level=info msg="Usage stats are ready to report"
Apr 25 15:25:56 raspberrypi grafana[1296]: logger:cleanup t=2025-04-25T15:25:56.602207861+02:00 level=info msg="Completed cleanup jobs" duration=142.25722ms
Apr 25 15:25:57 raspberrypi grafana[1296]: logger:plugins.update.checker t=2025-04-25T15:25:57.242868976+02:00 level=info msg="Update check succeeded" duration=60.706283ms
Apr 25 15:35:56 raspberrypi grafana[1296]: logger:cleanup t=2025-04-25T15:35:56.699514354+02:00 level=info msg="Completed cleanup jobs" duration=239.968934ms
Apr 25 15:35:57 raspberrypi grafana[1296]: logger:plugins.update.checker t=2025-04-25T15:35:57.25897555+02:00 level=info msg="Update check succeeded" duration=68.631913ms
Apr 25 15:45:56 raspberrypi grafana[1296]: logger:cleanup t=2025-04-25T15:45:56.658819235+02:00 level=info msg="Completed cleanup jobs" duration=199.719927ms
Apr 25 15:45:57 raspberrypi grafana[1296]: logger:plugins.update.checker t=2025-04-25T15:45:57.266885784+02:00 level=info msg="Update check succeeded" duration=84.987626ms
Apr 25 15:47:50 raspberrypi grafana[1296]: logger:infra.usagestat t=2025-04-25T15:47:50.530832231+02:00 level=info msg="Usage stats are ready to report"
Pi@raspberrypi:~$
```

7. Connaitre l'adresse IP du Raspberry :

```
hostname -I
```

8. Accès via navigateur :

```
http://adresse_IP_Raspberry:3000
```

Configuration de Grafana

Pour connecter Grafana à InfluxDB :

1. Aller dans **Connections > Data sources**
2. Rechercher et sélectionner InfluxDB
3. Dans la section **HTTP**, ajouter :

```
http://localhost:8086
```

4. Dans **InfluxDB Details**, renseigner le nom de la base de données : **telegraf**
5. Sauvegarder la configuration

Première vizualisation graphique

Pour avoir le premier graphique, aller sur Dashboards, puis "New Dashboard", "Add Visualization", choisir influxdb. Ensuite sur l'interface que vous avez choisissiez votre requête en fonction de l'information à matérialiser et le type de graphe approprié.

Voici un exemple :

