2025/12/14 19:32 1/3 Installer script srvstatus

# **Installer script srystatus**

• **Notes** : le script ratibor78/srvstatus nous permet de surveiller le status de services *Systemd* via *Grafana*.

#### Installer le script

- Se placer dans /opt : cd /opt
- Cloner le dépôt : git clone https://github.com/ratibor78/srvstatus.git
- Se placer dans le dossier du script : cd /opt/srvstatus
- Installer le paquet : apt install python3-venv
- Créer un venv avec Python 3 : python3 -m 'venv' ./venv
- Activer le *venv* : source venv/bin/activate
- Installer les paquets requis : pip install -r requirements.txt
- Rendre exécutable le script : chmod +x ./service.py

## **Configurer le script**

- Copier le fichier setting.ini depuis le dépôt Github sinp-<region>-srv: wget https://raw.githubusercontent.com/cbn-alpin/sinp-<region>-srv/main/<inst ance>-srv/opt/srvstatus/settings.ini
  - Suivant le SINP et le serveur, remplacer < region > (par paca ou aura) et instance (par web ou db)
- Si le fichier n'est pas dispo dans le dépôt, vous pouvez créer le fichier de config puis le stocker dans le dépôt Git sinp-<region>-srv : cp settings.ini.back settings.ini
- Vérifier ou définir les noms des services *Systemd* à surveiller dans le fichier *setting.ini* : vi settings.ini

## Préparer le Cron du script

- Copier le fichier srvstatus.cron depuis le dépôt Github sinp-<region>-srv: wget https://raw.githubusercontent.com/cbn-alpin/sinp-<region>-srv/main/<inst ance>-srv/opt/srvstatus/srvstatus.cron
  - Suivant le SINP et le serveur, remplacer < region> (par paca ou aura) et instance (par web ou db)
- Si le fichier n'est pas disponible dans le dépôt, vous pouvez créer le fichier puis le stocker dans le dépôt Git sinp-<region>-srv : vi srvstatus.cron
- Y placer le contenu suivant :

```
# /etc/cron.d/srvstatus: crontab entries for the srvstatus script
# Copy this file into /etc/cron.d/ without .cron extension : cp
srvstatus.cron /etc/cron.d/srvstatus
SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/usr/sbin:/usr/sbin
```

```
* * * * * root (sleep 10 ; /opt/srvstatus/venv/bin/python
/opt/srvstatus/service.py > /opt/srvstatus/status.json)
* * * * * root (sleep 20 ; /opt/srvstatus/venv/bin/python
/opt/srvstatus/service.py > /opt/srvstatus/status.json)
          root (sleep 30 ; /opt/srvstatus/venv/bin/python
/opt/srvstatus/service.py > /opt/srvstatus/status.json)
* * * * * root (sleep 40 ; /opt/srvstatus/venv/bin/python
/opt/srvstatus/service.py > /opt/srvstatus/status.json)
* * * * * root (sleep 50 ; /opt/srvstatus/venv/bin/python
/opt/srvstatus/service.py > /opt/srvstatus/status.json)
          root (sleep 60 ; /opt/srvstatus/venv/bin/python
/opt/srvstatus/service.py > /opt/srvstatus/status.json)
```

• Copier le fichier de *Cron* pour l'activer : cp srvstatus.cron /etc/cron.d/srvstatus

#### **Configurer Telegraf**

- Avant lancer Telegraf, assurez vous d'avoir lancer manuellement une première fois le script srvstatus en root afin de créer le fichier status. ison afin qu'il soit correctement pris en compte par le volume de Docker (voir ci-dessous) : /opt/srvstatus/venv/bin/python /opt/srvstatus/service.py > /opt/srvstatus/status.json
- Penser à ajouter le volume correspondant au fichier status.json dans le docker-compose.yml afin que Telegraf y est accès :

```
services:
  telegraf:
    volumes:
      - /opt/srvstatus/status.json:/opt/srvstatus/status.json
```

• Ajouter dans le fichier telegraf.conf utiliser par le container Docker la configuration suivante :

```
[[inputs.exec]]
 commands = [
    "cat /opt/srvstatus/status.json"
 timeout = "5s"
 name override = "services stats"
 data format = "ison"
 tag keys = [
    "service"
 ]
```

https://sinp-wiki.cbn-alpin.fr/ - CBNA SINP

Permanent link:

https://sinp-wiki.cbn-alpin.fr/serveurs/installation/script-srvstatus?rev=1685348285

Last update: 2023/05/29 08:18



2025/12/14 19:32 3/3 Installer script srvstatus