

# Outil Open Source : Auto SS Config

Je suis ravi d'annoncer que j'ai ouvert les sources d'un outil appelé **Auto SS Config**. Cet outil génère et télécharge automatiquement des URL d'abonnement Shadowsocks ou Clash à partir d'URL Shadowsocks, facilitant ainsi la gestion et la mise à jour des configurations de votre serveur proxy.

Cet outil a été un véritable changement de jeu pour moi, surtout lorsque mon serveur Shadowsocks est bloqué. J'utilise Outline Manager pour créer un nouveau serveur, obtenir une nouvelle adresse, et importer directement cette URL via l'application Mac pour contourner les restrictions du GFW. L'exécution de `python upload_configs.py` à partir de ce projet met à jour mes URL d'abonnement, garantissant que tous mes appareils numériques maintiennent des connexions réseau fonctionnelles.

## Fonctionnalités

- **Convertit les URL Shadowsocks en configuration Clash** : Passez facilement d'une configuration de proxy à une autre.
- **Prend en charge plusieurs serveurs Shadowsocks** : Gérez plusieurs serveurs sans difficulté.
- **Télécharge automatiquement les configurations sur Google Cloud Storage** : Gardez vos configurations sécurisées et accessibles.
- **Rend les configurations accessibles publiquement** : Partagez vos configurations avec d'autres utilisateurs.
- **Utilise le contrôle de cache pour des mises à jour immédiates** : Assurez-vous que vos configurations sont toujours à jour.

## Fichiers

- `app_config_tmp.yaml` : Configuration de l'application (nom du bucket, URLs SS).
- `clash_config_tmp.yaml` : Fichier de configuration temporaire pour Clash.
- `upload_configs.py` : Script pour générer la configuration Clash et téléverser les configurations vers Google Cloud Storage.
- `requirements.txt` : Dépendances Python.

## Configuration

1. **Installer les dépendances** : `bash pip install -r requirements.txt`
2. **Configurer les identifiants Google Cloud** :
  - Installez le SDK Google Cloud.
  - Exécutez `gcloud auth application-default login`.
  - Ou définissez la variable d'environnement `GOOGLE_APPLICATION_CREDENTIALS`.
3. **Copiez `app_config_tmp.yaml` vers `app_config.yaml` et configurez** : `yaml bucket_name: votre-nom-de-bucket ss_urls: - ss://method:motdepasse@serveur:port`

## Utilisation

1. **Ajoutez vos URLs Shadowsocks à la liste `ss_urls` dans `app_config.yaml`** : `yaml ss_urls: - ss://method:password@server:port`
2. **Téléverser les configurations** : `bash python upload_configs.py`  
Le script affichera les URL publiques pour les deux configurations.

## Développement

- **Python 3.6+**
- Utilise `ruamel.yaml` pour la manipulation des fichiers YAML.
- Utilise `google-cloud-storage` pour les opérations sur GCS.

## Licence

MIT (Massachusetts Institute of Technology)

---

N'hésitez pas à consulter le dépôt pour plus de détails et pour contribuer !