Open Source Tool: Auto SS Config

I'm excited to announce that I've open-sourced a tool called **Auto SS Config**. This tool automatically generates and uploads Shadowsocks or Clash subscription URLs from Shadowsocks URLs, making it easier to manage and update your proxy server configurations.

This tool has been a game-changer for me, especially when my Shadowsocks server gets blocked. I use Outline Manager to create a new server, obtain a fresh address, and import this URL directly using the Mac app to bypass GFW restrictions. Running `python upload_configs.py` from this project updates my subscription URLs, ensuring all my digital devices maintain functional network connections.

Features

- **Converts Shadowsocks URLs to Clash configuration**: Easily switch between different proxy configurations.
- **Supports multiple Shadowsocks servers**: Manage multiple servers with ease.
- **Automatically uploads configurations to Google Cloud Storage**: Keep your configurations secure and accessible.
- **Makes configurations publicly accessible**: Share your configurations with others.
- **Uses cache control for immediate updates**: Ensure your configurations are always up-to-date.

Files

- `ss_urls.txt`: Contains Shadowsocks server URLs (one per line).
- `ss_urls_tmp.txt`: Temporary file containing Shadowsocks server URLs (one per line).
- `clash_config_tmp.yaml`: Temporary Clash configuration file.
- `app_config.json`: Application configuration (bucket name, file paths).
- `generate_clash_config.py`: Script to generate Clash config from SS URLs.
- `upload_configs.py`: Script to upload configs to Google Cloud Storage.
- `requirements.txt`: Python dependencies.

Setup

1. **Install dependencies**: bash   `pip install -r requirements.txt`

2. **Set up Google Cloud credentials**:

   - Install Google Cloud SDK.
   - Run `gcloud auth application-default login`.
   - Or set `GOOGLE_APPLICATION_CREDENTIALS` environment variable.

3. **Configure `app_config.json`**: json { "bucket_name": "your-bucket-name", "ss_config_file": "ss_urls.txt" }

Usage

1. **Copy `ss_urls_tmp.txt` to `ss_urls.txt` to use**.

2. **Add your Shadowsocks URLs to `ss_urls.txt`, one per line**: `ss://method:password@server:port`

3. **Upload configurations**: bash python upload_configs.py

   The script will output the public URLs for both configurations.

Development

- **Python 3.6+**
- Uses `ruamel.yaml` for YAML handling.
- Uses `google-cloud-storage` for GCS operations.

License

MIT

Feel free to check out the repository for more details and to contribute!