

Améliorer OpenWRT avec OpenClash et Shadowsocks

Mise à jour des listes de paquets

```
root@OpenWrt:~# opkg update
```

```
Téléchargement de https://downloads.openwrt.org/releases/22.03.3/targets/ramips/mt7621/packages/Packages
```

```
Liste des paquets disponibles mise à jour dans /var/opkg-lists/openwrt_core
```

```
Téléchargement de https://downloads.openwrt.org/releases/22.03.3/targets/ramips/mt7621/packages/Packages
```

```
Vérification de la signature réussie.
```

```
Téléchargement de https://downloads.openwrt.org/releases/22.03.3/packages/mipsel_24kc/base/Packages.gz
```

```
Liste des paquets disponibles mise à jour dans /var/opkg-lists/openwrt_base
```

```
Téléchargement de https://downloads.openwrt.org/releases/22.03.3/packages/mipsel_24kc/base/Packages.sig
```

```
Vérification de la signature réussie.
```

```
Téléchargement de https://downloads.openwrt.org/releases/22.03.3/packages/mipsel_24kc/luci/Packages.gz
```

```
Liste des paquets disponibles mise à jour dans /var/opkg-lists/openwrt_luci
```

```
Téléchargement de https://downloads.openwrt.org/releases/22.03.3/packages/mipsel_24kc/luci/Packages.sig
```

```
Vérification de la signature réussie.
```

```
root@OpenWrt:~#
```

Installation du Plugin Shadowsocks

Pour installer le plugin luci-app-shadowsocks-libev :

```
root@OpenWrt:~# opkg install luci-app-shadowsocks-libev
```

```
Installation de luci-app-shadowsocks-libev (git-22.066.30464-cea4277) sur root...
```

```
Téléchargement de https://downloads.openwrt.org/releases/22.03.3/packages/mipsel_24kc/luci/luci-app-sha
```

```
Configuration de luci-app-shadowsocks-libev.
```

Installation d'OpenClash

Consultez le dépôt GitHub d'OpenClash pour plus de détails. Voici les étapes pour installer les composants nécessaires.

1. Installez le serveur SFTP OpenSSH :

```
opkg install openssh-sftp-server
```

2. Utilisez `scp` pour copier le package OpenClash sur le routeur :

```
scp luci-app-openclash_0.46.050-beta_all.ipk root@192.168.1.1:~/
```

Exemple de Configuration OpenClash

Voici un exemple de configuration pour OpenClash. Assurez-vous de personnaliser les paramètres en fonction de vos besoins spécifiques.

```
# Exemple de configuration OpenClash
```

```
port: 7890
socks-port: 7891
redir-port: 7892
allow-lan: true
mode: Rule
log-level: info
external-controller: 127.0.0.1:9090
secret: ""
tunnels:
  - name: "tunnel-1"
    type: "http"
    server: "example.com"
    port: 80
    username: "user"
    password: "pass"
proxies:
  - name: "proxy-1"
    type: "ss"
    server: "server1.example.com"
    port: 8388
    password: "password"
    cipher: "aes-256-cfb"
proxy-groups:
  - name: "auto"
    type: "url-test"
    proxies:
      - "proxy-1"
```

```

url: "http://www.google.com/generate_204"
interval: 300
rules:
- DOMAIN-SUFFIX,google.com,auto
- DOMAIN-SUFFIX,youtube.com,auto
- IP-CIDR,127.0.0.0/8,DIRECT
- GEOIP,CN,DIRECT
- MATCH,auto

```

Explication des Paramètres

- **port**: Le port utilisé pour le proxy HTTP.
- **socks-port**: Le port utilisé pour le proxy SOCKS.
- **redir-port**: Le port utilisé pour la redirection.
- **allow-lan**: Autorise ou non l'accès depuis le réseau local.
- **mode**: Le mode de fonctionnement (par exemple, `Rule` pour un mode basé sur des règles).
- **log-level**: Niveau de journalisation (par exemple, `info` pour des informations générales).
- **external-controller**: Adresse et port pour le contrôle externe.
- **secret**: Mot de passe pour le contrôle externe (laisser vide pour aucun).
- **tunnels**: Configuration des tunnels (par exemple, pour le tunneling HTTP).
- **proxies**: Liste des serveurs proxy disponibles.
- **proxy-groups**: Groupes de proxies pour des tests de performance ou d'autres critères.
- **rules**: Règles de routage pour déterminer comment le trafic est dirigé.

N'oubliez pas de sauvegarder et de redémarrer OpenClash après avoir modifié la configuration.

Voici un exemple de configuration pour OpenClash :

```

port: 7890
socks-port: 7891
mixed-port: 7892
allow-lan: true
mode: Rule
log-level: info
external-controller: 0.0.0.0:9090
experimental:
  ignore-resolve-fail: true

```

```

dns:
  enable: true
  # ipv6: false
  listen: 0.0.0.0:53
  fake-ip-range: 198.18.0.1/16
  default-nameserver:
    - 119.29.29.29
    - 223.5.5.5
    #- 223.6.6.6
  nameserver:
    - https://223.5.5.5/dns-query
    - https://1.12.12.12/dns-query
    #- https://doh.pub/dns-query
    #- https://dns.alidns.com/dns-query
  fake-ip-filter:
    - "*.lan"
    - "*.localdomain"
    - "*.example"
    - "*.invalid"
    - "*.localhost"
    - "*.test"
    - "*.local"

```

proxies: - name: "Mon Proxy SS" type: ss server: 209.97.0.0 port: 57500 cipher: chacha20-ietf-poly1305 password: "jHLE54zNC000000" udp: true plugin: "" plugin-opts: {}

```

proxy-groups:
  - name: "Proxy"
    type: select
    proxies:
      - "My SS Proxy"

```

rules: - GEOIP,CN,DIRECT - MATCH,Proxy ""

Avec ces étapes, vous pouvez améliorer votre appareil OpenWRT pour prendre en charge des configurations réseau avancées et des fonctionnalités de proxy. Consultez toujours la documentation officielle pour les mises à jour et les meilleures pratiques.