

Config de Weechat et +

Les Options modifiées et leur valeur

Ce sont des commandes set lancées dans l'interface Weechat.

Chaque commande modifie une option de configuration.

Vous pouvez lister toutes les options avec cette commande : /fset *

La liste ci-dessous est à titre d'exemple et s'applique à ma situation.

```
/set irc.look.color_nicks_in_nicklist on
/set irc.look.nicks_hide_password "nickserv|nickbot"
/set irc.server.OFTC.addresses "irc.oftc.net/6697"
/set irc.server.OFTC.autoconnect on
/set irc.server.OFTC.autojoin "#debian-facile,#df-offtopic"
/set irc.server.OFTC.autorejoin on
/set irc.server.OFTC.command "/msg nickserv identify ${sec.data.OFTC}"
/set irc.server.OFTC.ssl on
/set irc.server.OFTC.ssl_cert "%h/certs/nick.pem"
/set irc.server.OFTC.ssl_verify on
/set irc.server_default.nicks "tawal,tawal1,tawal2,tawal3,tawal4"
/set irc.server_default.username "tawal"
/set sec.crypt.passphrase_file "%h/certs/passphrase"
/set spell.check.default_dict "fr"
/set spell.check.enabled on
/set spell.check.suggestions 3
/set weechat.bar.buflist.items "buflist"
/set weechat.bar.fset.items "fset"
/set weechat.color.chat_nick_colors
"31,35,38,40,49,63,70,80,92,99,112,126,130,138,142,148,160,167,169,174,176,178,184,186,210,215,215,247"
/set weechat.color.chat_nick_self lightblue
```

Les Commandes passées changeant le comportement

```
/set env TERM screen-256color
/window splith # depuis le 1er canal
/layout store

/secure passphrase Une 'passphrase' de votre choix
/secure set OFTC Mon_mot_de_passe # Donnée utilisée ici : /set
irc.server.OFTC.command "/msg nickserv identify ${sec.data.OFTC}"
```

Chiffrer la passphrase et le certificat ssl - Déchiffrer au lancement de Weechat

C'est selon la version de Weechat.

Actuellement sur Debian Bullseye, la version est la 3.0

Cette version comporte l'option `sec.crypt.passphrase_file` mais pas l'option `sec.crypt.passphrase_command`.

Le problème : en renseignant un fichier pour la passphrase, la 1ère ligne de ce fichier est utilisé pour la passphrase mais elle doit être en clair dans le fichier.

Donc si vous avez cette option `sec.crypt.passphrase_file`, vous pouvez appliquer la solution ci-dessous.

Mais cette solution est aussi pratique pour chiffrer son certificat ssl (ou autre).

Préalable

Nous travaillerons dans le répertoire `~/.weechat/certs` *créez le si besoin*

Et j'utilise `gpg` pour chiffrer/déchiffrer les fichiers.

Donc, une fois le certificat obtenu, nous le cryptons avec : `gpg -ecr Nom_Utilisateur Nom_Certificat`

De même avec la *passphrase* : `gpg -ecr Nom_Utilisateur <(echo "LA PASSPHRASE ICI") > passphrase.gpg`

Nous obtenons ainsi 2 fichiers :

- `Nom_Certificat.gpg`
- `passphrase.gpg`

Il faut les déplacer dans `~/.weechat/certs`

Et nous pouvons supprimer le certificat en clair.

Si vous n'avez pas encore de clé `gpg`, vous pouvez la créer ainsi : `gpg --quick-gen-key`
Et suivez les instructions.

Mise en place du 'hack'

Voici un script qui déchiffre les fichiers avant le lancement de Weechat.

Lance Weechat.

Puis détruit les fichiers en clair dès qu'ils ont été lus (qui ont donc servi).

Il utilise `inotifywait` et donc nécessite le paquet `inotify-tools`.

wee-pass

```
#!/bin/bash

pass_crypt="$HOME/.weechat/certs/passphrase.gpg"
cert_crypt="$HOME/.weechat/certs/nick.pem.gpg"

gpg -dr tawal --output "${pass_crypt%.*}" "$pass_crypt" # Adapter le
nom d'utilisateur.
gpg -dr tawal --output "${cert_crypt%.*}" "$cert_crypt" # Pareil ici.

# Boucle d'effacement des fichiers en clair dès qu'ils ont été lus.
```

```
while read -r file
do
    [ -f "$file" ] && {
        rm "$file"
        ((++i))
    }
    ((i>=2)) && break
done <<(inotifywait -qme access --format "%w" "${pass_crypt%.*}"
"${cert_crypt%.*}") &

/usr/bin/weechat
```

Placez-le dans un répertoire du PATH.

Vous n'avez plus qu'à taper `wee-pass` dans un terminal pour voir Weechat se lancer et les authentifications se faire 😊

From:

<http://debian-facile.org/> - **Documentation - Wiki**

Permanent link:

<http://debian-facile.org/utilisateurs:tawal:configs:irc-weechat>

Last update: **14/05/2023 12:27**

