Installation serveur IRC sous debian

L'installation

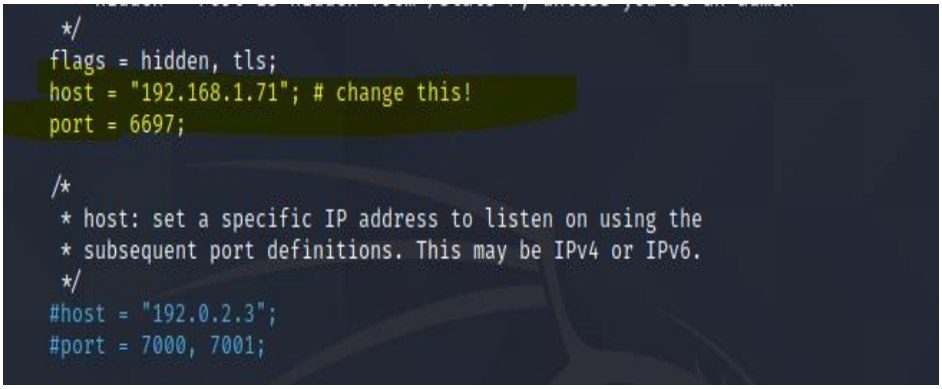
Je vais suivre ce tuto

https://www.moyens.net/tech/comment-transformer-votre-raspberry-pi-en-serveur-irc/

J'installe d'abord le paquet ircd-hybrid

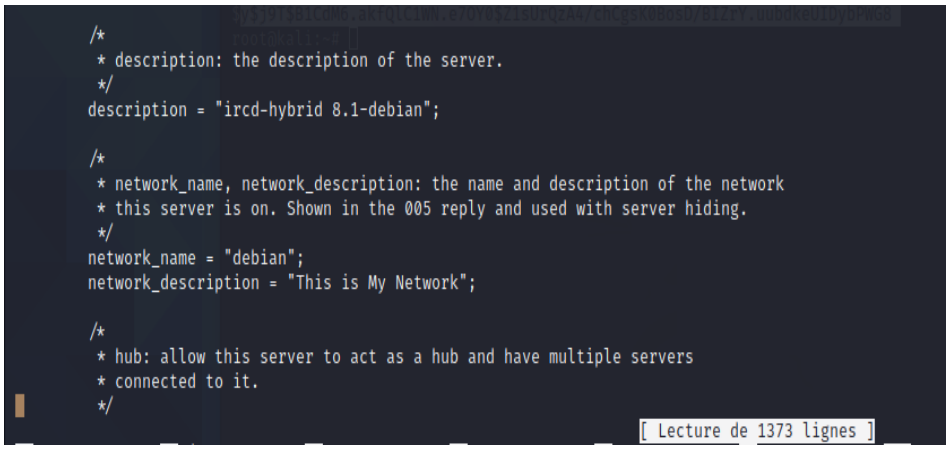
Une fois installé je vais dans le fichier : /etc /irc-hybrid /ircd.conf

Pour l'instant je ne change ni le nom du serveur ni sa description je définis juste une IP et un port



Pour changer le nom et la description je fais ceci

C'est ici

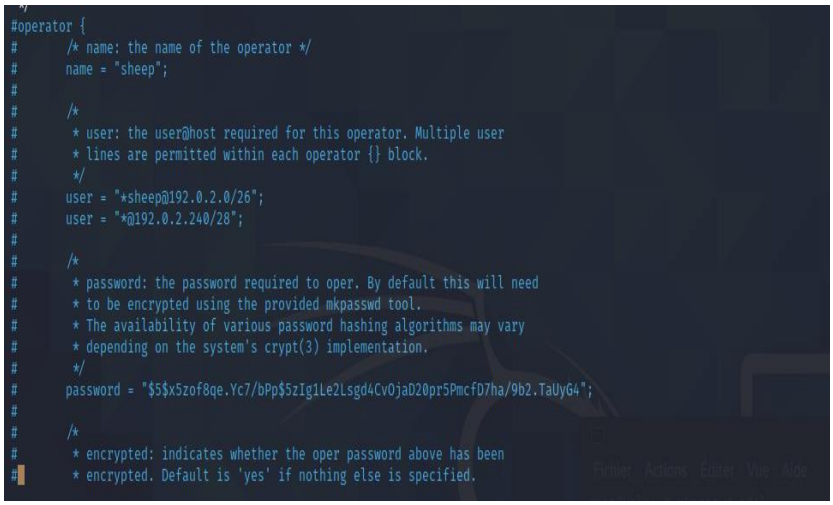


Les operateurs

Pour pouvoir utiliser ce serveur il me faut des utilisateurs et des opérateurs créer

Je vais créer un opérateur qui aura les droits totaux sur le serveur pour modérer etc

Je me rend dans le fichier de conf et je descends à la case operateur



Je décommente les lignes utile je donne un nom à l'user operateur je vais l'appeler operateur

Et dans la case user je vais mettre user = operateur@*

Pour que n'importe quelle personne qui se connecte avec le compte operateur depuis n'importe quel IP et le bon mdp soit accepter

Ensuite je crypte un mot de passe avec la commande mkpasswd <suivi d'un nom>

Sa me donnera une empreinte md5 je la prend et je la colle dans le fichier de conf

Voilà mon operateur est créé je verrai plus tard comment il pourra administrer le serveur

Les utilisateurs

Ensuite je passe aux utilisateurs je vais dans cette case

A la base après test il y'avait mon adresse IP local ça veut dire que seul l'adresse IP local pouvait se connecter en utilisant l'utilisateur test

```
auth {
        /*
         * user: the user@host allowed to connect. Multiple user
         * lines are permitted within each auth {} block.
         */
        user = "test@*";
        #user = "*test@2001:DB8:*";

        /* password: an optional password that is required to use this block. */
        #password = "letmein";

        /*
         * encrypted: indicates whether the auth password above has been
         * encrypted. Default is 'no' if nothing else is specified.
         */
        #encrypted = yes;
```

La quand je l'ai modifié toute les IP pouvaient se connecter avec l'user test

Si je mets une asterix a la place de test tous les utilisateurs pourront se connecter avec n'importe quel nom depuis n'importe quel IP

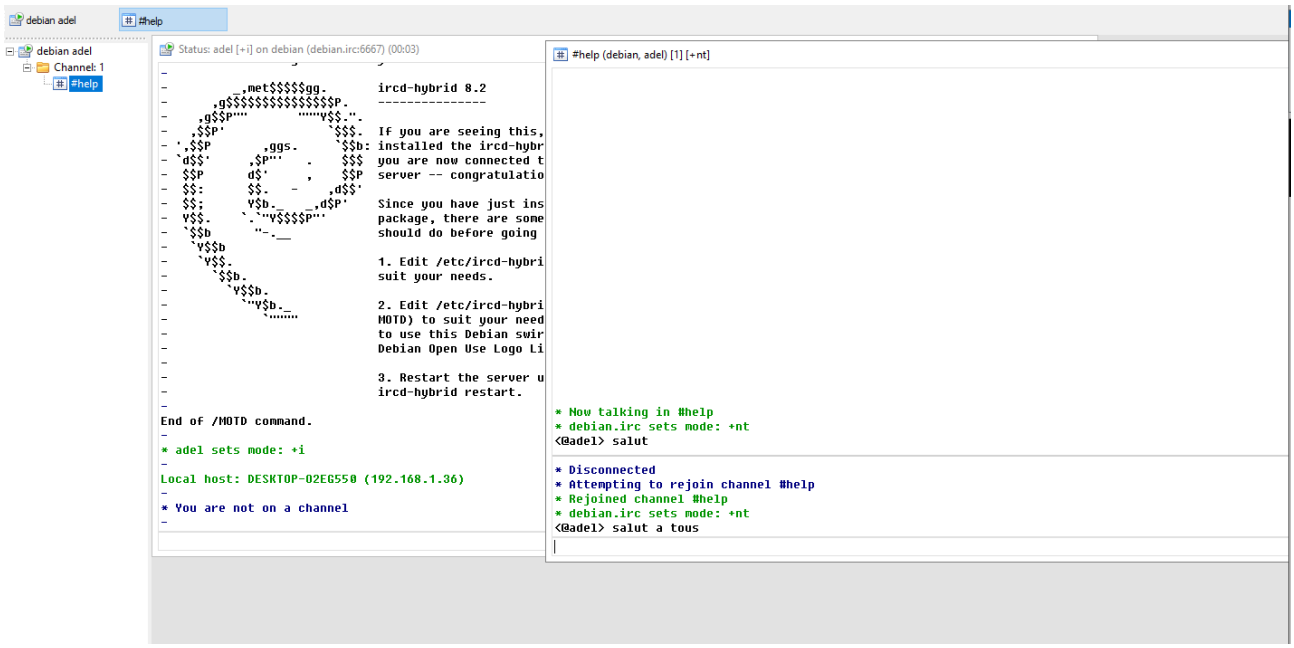
J'ai un souci lorsque je précise qu'un seul utilisateur peut se connecter avec n'importe quel IP ça ne fonctionne pas mais lorsque je mets que n'importe quel IP peut se connecter avec n'importe quel utilisateur sa fonctionne parfaitement

J'ai fait une nouvelle case « auth{« dans mon fichier de conf

```
auth{
user = "adel@*";
class = "users";
#flag = need_ident;
};
```

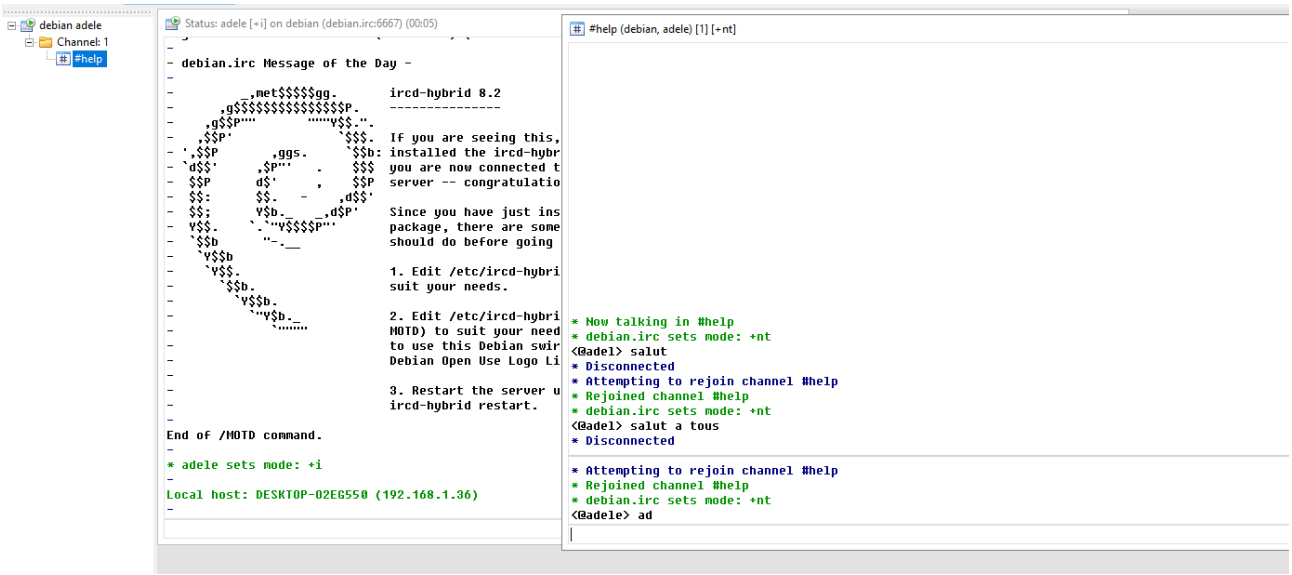
J'ai installé un client irc sur mon windows et lorsque j'essaye de me connecter sa fonctionne parfaitement appart une erreur hostname au debut mais a fini quand meme par me connecter

Voilà



Normalement le seul utilisateur autoriser est adel si je veux vérifier que ça fonctionne correctement je vais me connecter avec un autre user

Cela fonctionne quand même….



J’ai essayé avec un user du nom de « adele »

J’ai décommenter ses lignes

Et lorsque je veux me connecter à l'user test sans mot de passe sa m'affiche une erreur mot de passe incorrect

```
auth {
        /*
         * user: the user@host allowed to connect. Multiple user
         * lines are permitted within each auth {} block.
         */
        user = "test@*";
        #user = "letest@*";
        #user = "*test@2001:DB8:*";

#       /* password: an optional password that is required to use this block. */
        password = "letmein";
```

Si je désactive le cryptage du mdp et que je tape letmein ça fonctionne parfaitement

LES FLAGS

Les flags permettent de mettre des regles dans le block « auth{«

Par exemple obliger le mdp, il faut rajouter flags = et mettre les règles

```
/*
 * need_password - don't allow users who haven't supplied the correct  | ('&' prefix on /stats I if disabled)
 *                 password to connect using another auth {} block
 * need_ident    - require the user to have identd to connect          | ('+' prefix on /stats I)
 * exceed_limit  - allow a user to exceed class limits                 | ('>' prefix on /stats I)
 * kline_exempt  - exempt this user from k-lines                       | ('^' prefix on /stats I)
 * xline_exempt  - exempt this user from x-lines                       | ('!' prefix on /stats I)
 * resv_exempt   - exempt this user from resvs                         | ('$' prefix on /stats I)
 * no_tilde      - remove ~ from a user with no ident                  | ('-' prefix on /stats I)
 * can_flood     - allow this user to exceed flood limits              | ('|' prefix on /stats I)
 * webirc        - enables WEBIRC authentication for web-based         | ('<' prefix on /stats I)
 *                 clients such as Mibbit
 */
flags = need_password, exceed_limit, kline_exempt, xline_exempt, resv_exempt, can_flood,need_ident;
```

Je peux faire un block auth par utilisateur par exemple ici j'ai créé l'user adel avec comme mdp adel j'ai tenté de me connecter en mettant abel comme mdp sa na pas fonctionner

```
auth{
user = "adel@*";
password= "adel";
class = "users";
flags = need_ident,no_tilde,need_password;
};
```

Lorsque je me connecte avec le mauvais mdp sa affiche sa

```
-
Password incorrect
-
Closing Link: 192.168.1.36 (Bad Password)
-
* Disconnected
-
* Connect cancelled
```

Et si je mets le bon mdp sa donne sa

Ça fonctionne

```
-
-                          3. Restart the server using i
-                          ircd-hybrid restart.
-
End of /MOTD command.
-
* adel sets mode: +i
-
Local host: DESKTOP-02EG550 (192.168.1.36)
-
```

Le fichier ircd.motd contient le message de bienvenue

Je vais maintenant générer ma propre clé privée et publique pour crypter les échanges et la communication du mdp

Ici je crée ma clé privée et publique que je stock dans le même répertoire des fichiers de conf

```
.....^C
root@kali:~# openssl genrsa 4096 > irc.key
Generating RSA private key, 4096 bit long modulus (2 primes)
.....++++
.....++++
e is 65537 (0x010001)
root@kali:~# ls
adel.py          blackhorse-honey.com  dico.txt   host.txt       Images          john.txt  nora      preview.colorlib.com  telnet.sh  yes.sh
adel.txt         Bureau                dns.txt    hts-cache      index.html      mail.py   pdf.py    Public                test.py
backblue.gif     cookies.txt           Documents  hts-log.txt    infoKhanAlassal Modèles   ping.txt  sio.txt               user.txt
bettercap.history deusmail.py          fade.gif   hydra.restore  irc.key         Musique   Pipfile   Téléchargements       Vidéos
root@kali:~# mv irc.key /etc/ircd-hybrid/
root@kali:~# cd /etc/ircd-hybrid/
root@kali:/etc/ircd-hybrid# openssl req -x509 -key irc.key -days 30600 -out irc.pem
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:FR
State or Province Name (full name) [Some-State]:PARIS
Locality Name (eg, city) []:PARIS
Organization Name (eg, company) [Internet Widgits Pty Ltd]:
```

Et dans le fichier de conf principale je modifie et j'ajoute ceci

```
     *    chown <ircd-user>.<ircd.group> rsa.key
     *    chmod 0600 rsa.key
 *
 * Note to Debian users: the postinst script created this
 * for you.
     */
    rsa_private_key_file = "/etc/ircd-hybrid/irc.key";

    /*
     * tls_certificate_file: the path to the file containing our
     * TLS certificate for encrypted client connection.
     */
    tls_certificate_file = "/etc/ircd-hybrid/irc.pem";

    /*
     * tls_dh_param_file: the path to the PEM encoded Diffie-Hellman
     * parameter file. DH parameters are required when using
     * ciphers with EDH (ephemeral Diffie-Hellman) key exchange.
```

J'ai installé le client irc hexchat sur mon linux le cryptage TLS a l'air de fonctionné correctement

J'ai aussi installé hexchat sur ma machine windows je fais une capture wireshark sur ma kali en même temps et voila au même moment de la connexion ce que je retrouve la connexion est bien cryptée dès le départ

3	0.000236496	192.168.1.36	192.168.1.71	TCP	54	53787 → 6697 [ACK] Seq=1 Ack=1 Win=2102272 Len=0
4	0.321399458	192.168.1.36	192.168.1.71	TLSv1.2	378	Client Hello
5	0.321421558	192.168.1.71	192.168.1.36	TCP	54	6697 → 53787 [ACK] Seq=1 Ack=325 Win=64128 Len=0
6	0.330759703	192.168.1.71	192.168.1.36	TLSv1.2	2334	Server Hello, Certificate, Server Key Exchange, Certificate Request, Server Hello Done
7	0.330945300	192.168.1.36	192.168.1.71	TCP	54	53787 → 6697 [ACK] Seq=325 Ack=2281 Win=2102272 Len=0
8	0.633077578	192.168.1.36	192.168.1.71	TLSv1.2	192	Certificate, Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
9	0.633101678	192.168.1.71	192.168.1.36	TCP	54	6697 → 53787 [ACK] Seq=2281 Ack=463 Win=64128 Len=0
10	0.633312874	192.168.1.71	192.168.1.36	TLSv1.2	105	Change Cipher Spec, Encrypted Handshake Message
11	0.633327674	192.168.1.71	192.168.1.36	TLSv1.2	135	Application Data
12	0.633381573	192.168.1.71	192.168.1.36	TCP	74	44907 → 113 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=4175327634 TSecr=0 WS=
13	0.633386073	192.168.1.71	192.168.1.36	TLSv1.2	125	Application Data
14	0.633404573	192.168.1.71	192.168.1.49	DNS	85	Standard query 0x397c PTR 36.1.168.192.in-addr.arpa
15	0.633558570	192.168.1.36	192.168.1.71	TCP	54	53787 → 6697 [ACK] Seq=463 Ack=2413 Win=2102016 Len=0

Je vais tenter de rajouter une couche de sécurité est d'envoyer un mdp hasher au lieu de l'envoyer en clair même si on est dans une conversation crypter RSA4096 bits

Donc je génère un mdp hasher avec mkpasswd qui sera « adel » pour ce test

Je fais mkpasswd adel je copie la sortie de cette commande

Je colle la sortie après password

```
auth {
        /*
         * user: the user@host allowed to connect. Multiple user
         * lines are permitted within each auth {} block.
         */
        user = "*@*";
        #user = "letest@*";
        #user = "*test@2001:DB8:*";

#       /* password: an optional password that is required to use this block. */
        password = "$y$j9T$vo5Wn.lGHcAoselhlJzWP.$5BTUakt2DmqBoiXoq0S8/BvyEC4jLqlpfrzzwD8sfC3";


        /*
         * encrypted: indicates whether the auth password above has been
         * encrypted. Default is 'no' if nothing else is specified.
         */
        encrypted = yes;
```

Et je définis encypted à yes

Ça fonctionne parfaitement

Lorsque je mets le mauvais mot de passe sa me met une erreur comme ceci

Et quand je mets le bon sa me connecte parfaitement

```
[20:54:31] * Capabilities supported: account-notify away-notify cap-notify chghost extended-join invite-notify multi-prefix userhost-in-names
[20:54:31] * Capabilities requested: account-notify away-notify cap-notify chghost extended-join multi-prefix userhost-in-names
[20:54:31] * Capabilities acknowledged: account-notify away-notify cap-notify chghost extended-join multi-prefix userhost-in-names
[20:54:31] * Password incorrect
[20:54:31] * Closing Link: 192.168.1.36 (Bad Password)
[20:54:31] * Disconnected (Remote host closed socket)
```

Résolution d'une erreur :

J'ai compris pourquoi mon auth ne marché pas quand je configurer des users dans « flag » j'avais mis la directive need_ident qui oblige la personne qui veut ce connecter à cet utilisateur à avoir déjà une identification IDENTD qui est un protocole d'identification

Wikipédia :
Le protocole d'identification, spécifié dans la RFC 1413, est un protocole Internet qui permet d'identifier l'utilisateur d'une connexion TCP particulière. Un programme de démon populaire pour fournir le service ident est identd.
FIN