

Gateway Meshtastic eJabberd – Mise en marche

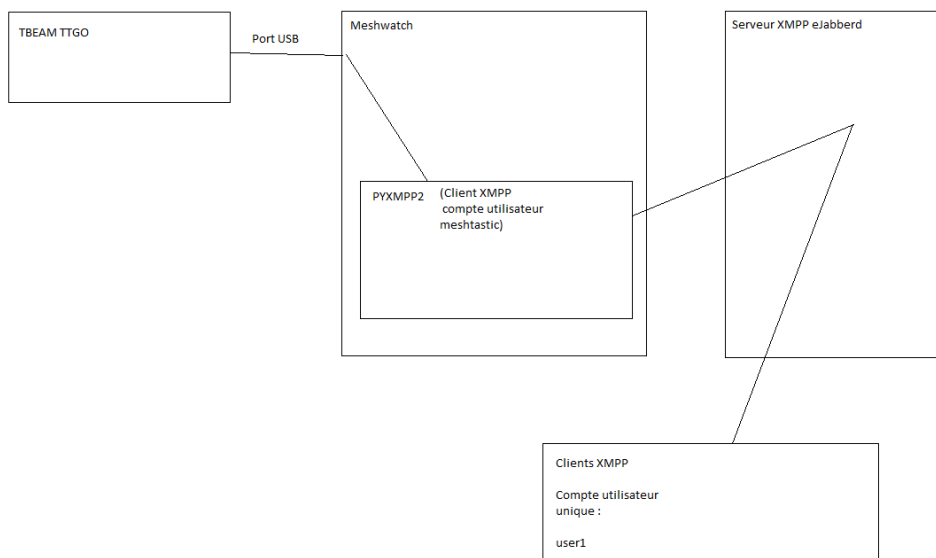
Adapté par ON3MEE

Introduction

Voici le projet d'un gateway Meshtastic eJabberd. Le gateway permet de se connecter à un réseau Meshtastic via un client de messagerie de tchat instantané XMPP sur un LAN. Cela permet de pouvoir se connecter à plusieurs stations (PC, smartphone, ...) sur un seul nœud Meshtastic à une interface de tchat web ou via des applications de tchat clientes XMPP. Ces applications clientes ressemblent à un tchat Messenger et sont compatible Windows, Android, IOS, ...

L'objectif initial du projet est de pouvoir mettre quelques PC (ou smartphones) sur le même nœud meshtastic et afficher un seul tchat pour un poste de commandement de services de secours. Un nœud Meshtastic n'accepte qu'un seul utilisateur !

Coté conception, je me suis basé sur la distribution Freedom Box pour le serveur XMPP et son client web et sur le projet Meshwatch (<https://github.com/datagod/meshwatch>) qui fait l'interface avec le nœud Meshtastic. J'ai adapté le code source de Meshwatch pour y intégrer le client PYXMPP2 (<https://github.com/Jajcus/pyxmpp2>).



AVERTISSEMENT !!!

Je vous préviens que je ne suis pas un codeur et que je déteste PYTHON. Epargnez-moi des remarques du genre : « Vous codez mal, ... ». Je suis très conscient que le projet ici n'est pas du tout bien pensé et que c'est du bricolage. Mais ça fonctionne.

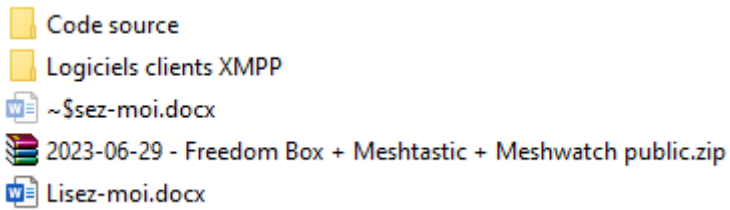
Je vous mets le code à disposition. N'hésitez pas à l'améliorer et à même reprendre le projet. Je suis 100 % ouvert à toute aide comme le veut le monde du logiciel libre.

Remerciements

Je remercie ON3CED (<http://www.on3ced.com/>) pour leur aide apportée pour les différents essais du réseau Meshtastic et du gateway Freedom Box.

On peut aussi remercier [datagod](#) (<https://github.com/datagod/meshwatch/commits?author=datagod>) qui a créé l'outil Meshtalk qui a permis le développement de ce projet.

Contenu du package



- *Code source*

Ce sont les listings des programmes modifiés et utilisé.

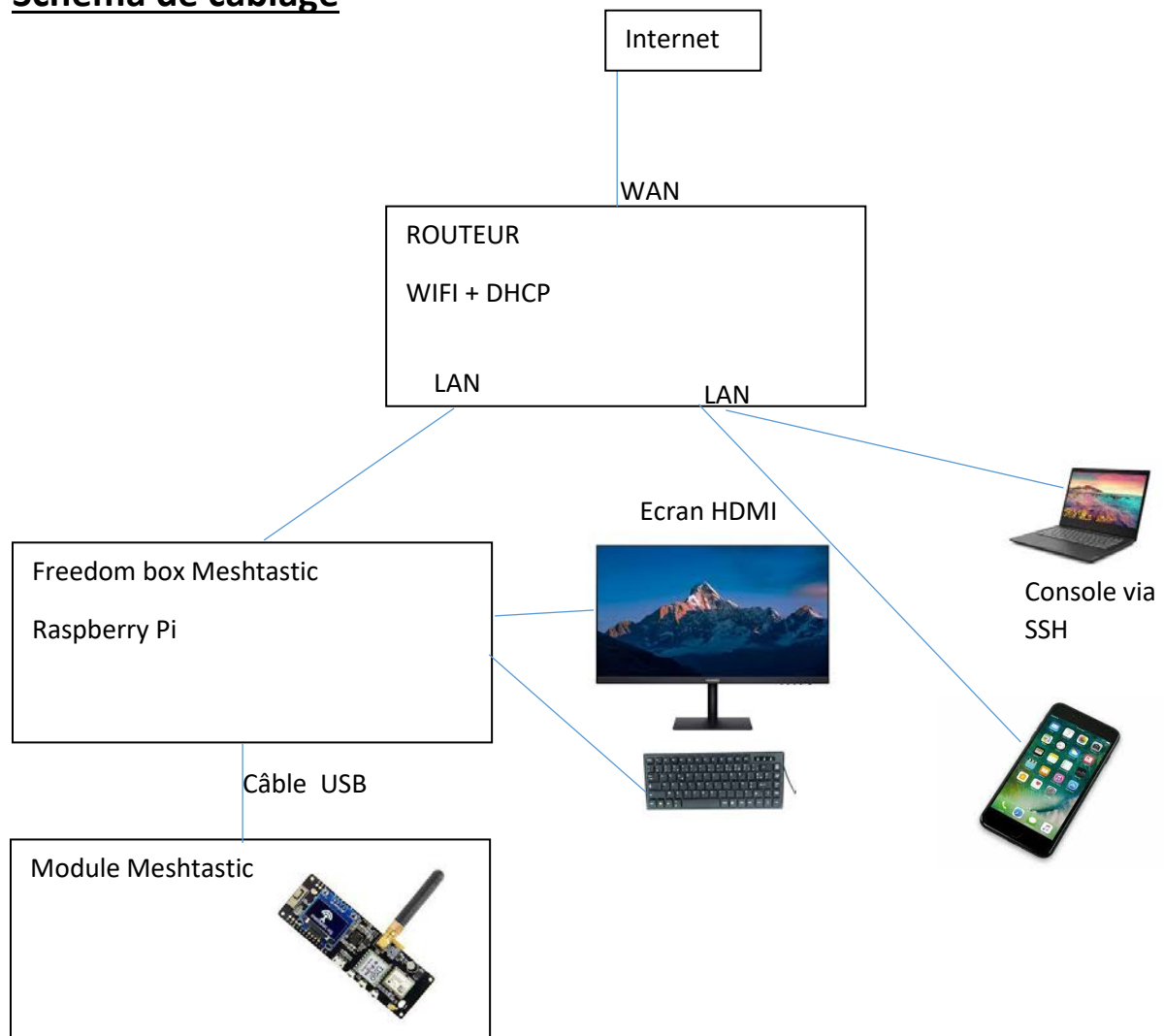
- *Logiciels clients XMPP*

Les installs de différentes applications clientes XMPP pour Windows, Linux et Android.

- *2023-06-09zip*

C'est l'image Freedom Box modifiée à installer sur la carte SD.

Schéma de câblage



Prérequis

Il faut câbler en respectant le schéma de la page précédente. On peut y ajouter des points d'accès WIFI, etc ...

Concernant le routeur, il est indispensable de configurer le nom de domaine correctement : localdomain.

Les comptes XMPP n'accepte que des adresses de ce genre : user@machinex.localdomain . Pas d'adresse IP !!! [user@192.168.1.1](#) ne fonctionnera pas !!! Si la résolution du domaine ne fonctionne pas, le système ne pourra pas fonctionner !

On règle ces paramètres dans le web CLI du routeur.

Dans ce projet, le nom de domaine est localdomain. Le nom de machine est meshtastic. Les comptes utilisateurs sont user1 ; user2 ; ... Cela donne une adresse du style :

[user1@meshtastic.localdomain](#).

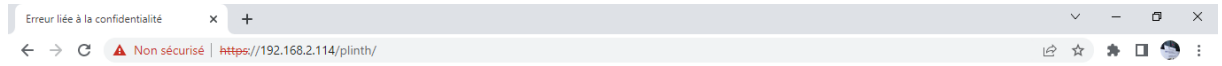
Dans certains cas, il faudra une adresse du style : user1@meshtastic et non user1@meshtastic.localdomain. Cela dépendra de votre routeur. Il faudra donc faire des tests pour déterminer le bon suffixe. Dans ce cas, il faudra modifier les adresses dans le code source de la freedom box (voir annexe de ce document).

Concernant les adresses IP, c'est à vous de décider. Il n'y a pas d'IP à respecter. Le rpi est configuré en IP dynamique. Il est cependant conseillé de fixer l'adresse IP de la freedom box.

Concernant le WIFI, c'est aussi à vous de décider.

Freedom Box et le serveur XMPP eJabberd

1. Mettre l'image «img » sur une carte SD de 16 GB dans un rpi 3 au minimum.
2. Connecter sur l'adresse IP sur rpi via un navigateur web :



Votre connexion n'est pas privée

Des individus malveillants tentent peut-être de subtiliser vos informations personnelles sur le site **192.168.2.114** (mots de passe, messages ou numéros de carte de crédit, par exemple). [En savoir plus](#)

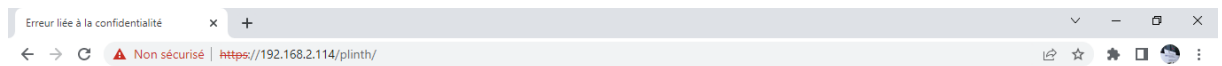
NET-ERR_CERT_AUTHORITY_INVALID



Pour bénéficier du niveau de sécurité le plus élevé de Chrome, [activez la protection renforcée](#)

Paramètres avancés

Revenir en lieu sûr



Votre connexion n'est pas privée

Des individus malveillants tentent peut-être de subtiliser vos informations personnelles sur le site **192.168.2.114** (mots de passe, messages ou numéros de carte de crédit, par exemple). [En savoir plus](#)

NET-ERR_CERT_AUTHORITY_INVALID



Pour bénéficier du niveau de sécurité le plus élevé de Chrome, [activez la protection renforcée](#)

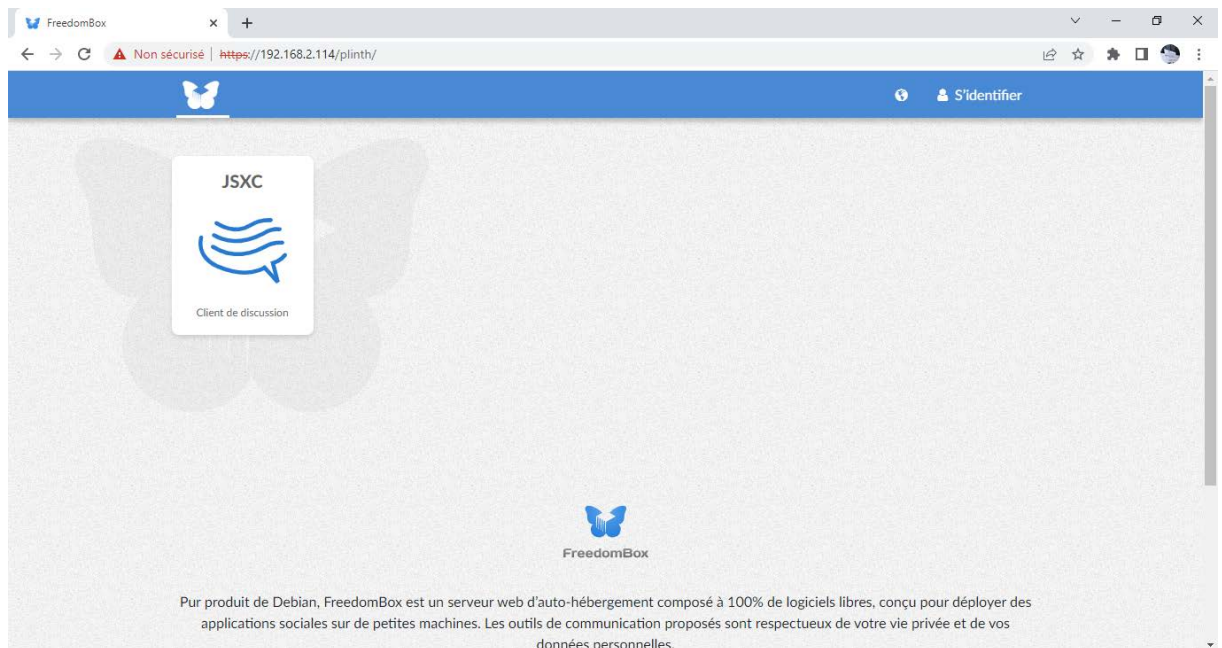
Masquer les paramètres avancés

Revenir en lieu sûr

Impossible de vérifier sur le serveur qu'il s'agit bien du domaine **192.168.2.114**, car son certificat de sécurité n'est pas considéré comme fiable par le système d'exploitation de votre ordinateur. Cela peut être dû à une mauvaise configuration ou bien à l'interception de votre connexion par un pirate informatique.

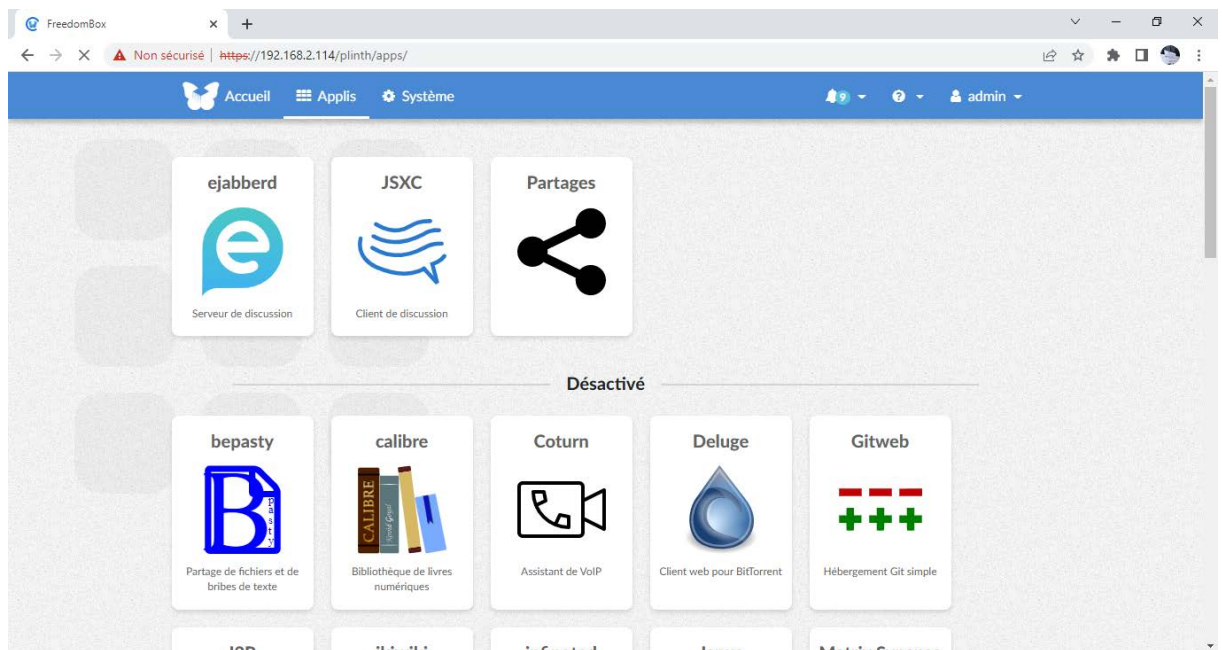
[Continuer vers le site 192.168.2.114 \(dangereux\)](#)

3. Cliquer sur « Continuer vers le site xxx (dangereux). Ce n'est pas dangereux. Le certificats est autosigné !
4. Le web CLI de freedom box apparait :



Login : admin – Mot de passe : adminadmin

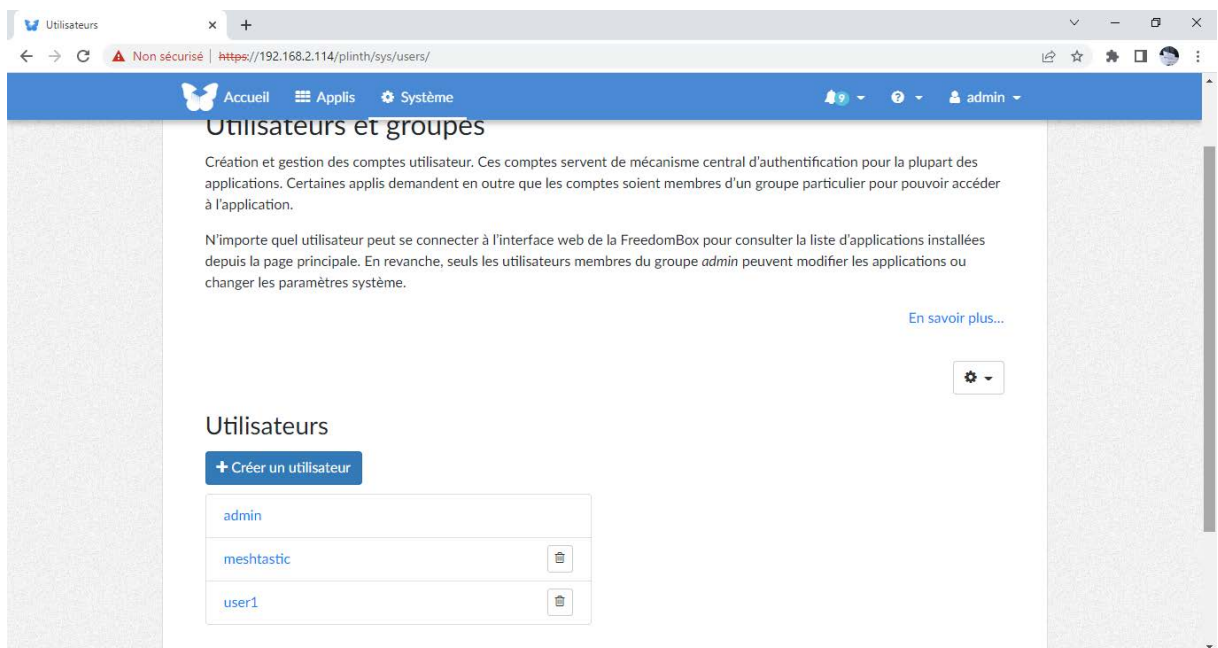
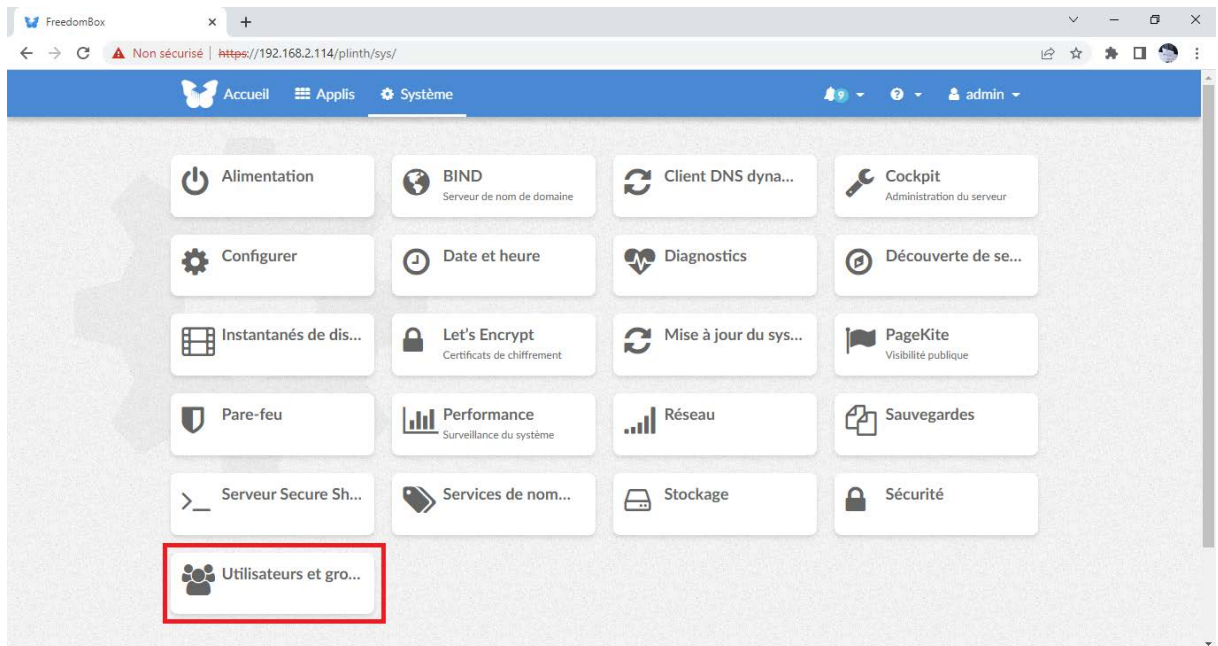
5. Les applications clientes sont ici :



Celle qui nous intéresse est eJabberd pour le serveur de messagerie XMPP et JSXC pour le client web de messagerie.

Laisser les paramètres configurés.

6. Les comptes utilisateurs se règle ici :



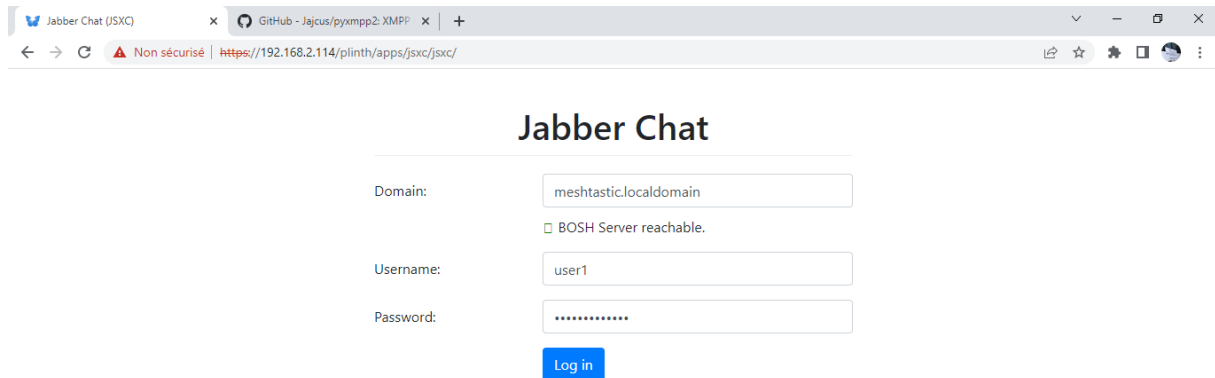
Les comptes déjà créés ne doivent pas être changés. Ils sont liés au programme meshwatch modifié. Libre à vous de changer, mais il faudra adapter le code source en conséquence !

Voici les credentials :

user : meshtastic / password : 2022user12022 -> C'est le compte qui fait la passerelle entre Meshwatch et eJabberd

User : user1 / password : 2022user12022 -> C'est le compte client avec lequel vous allez vous connecter sur le client JSXC.

7. Voici le client JSXC



Jabber Chat

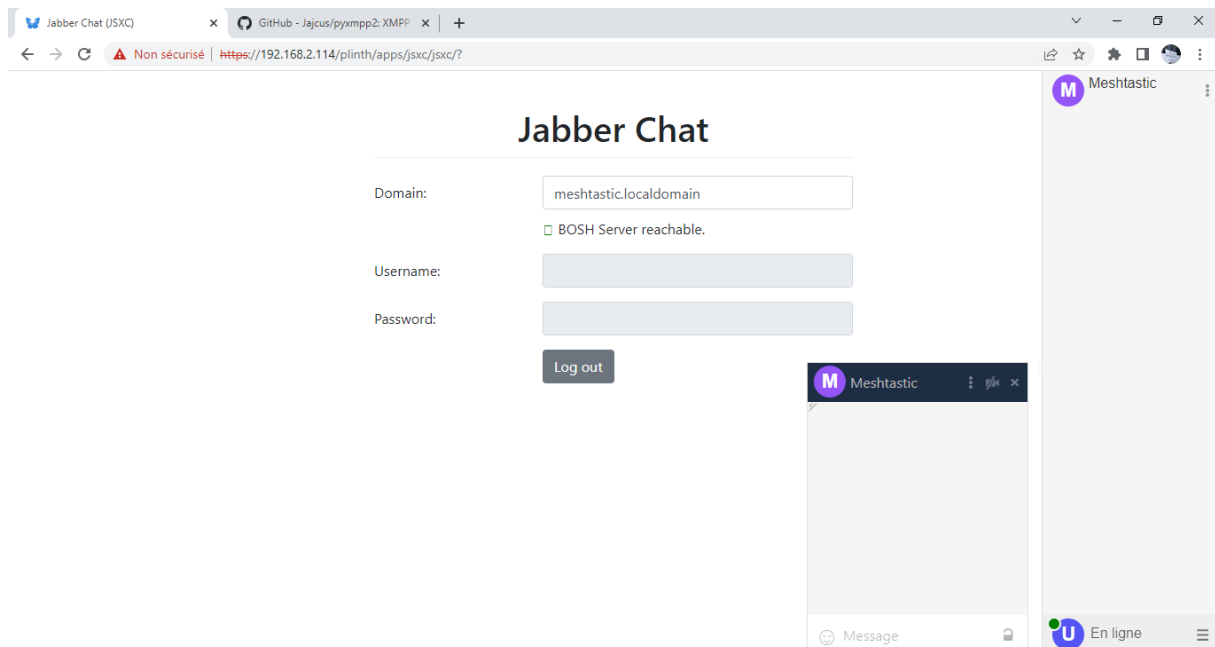
Domain:

☐ BOSH Server reachable.

Username:

Password:

L'adresse XMPP est du genre user1@meshtastic.localdomain . Il est TRES IMPORTANT de respecter la syntaxe user@meshtastic.localdomain . Une IP à la place du domaine ne marche pas. Il faudra bien programmer votre routeur avec le suffixe userx@meshtastic.localdomain ou userx@meshtastic.



Jabber Chat

Domain:

☐ BOSH Server reachable.

Username:

Password:

Meshtastic

Message

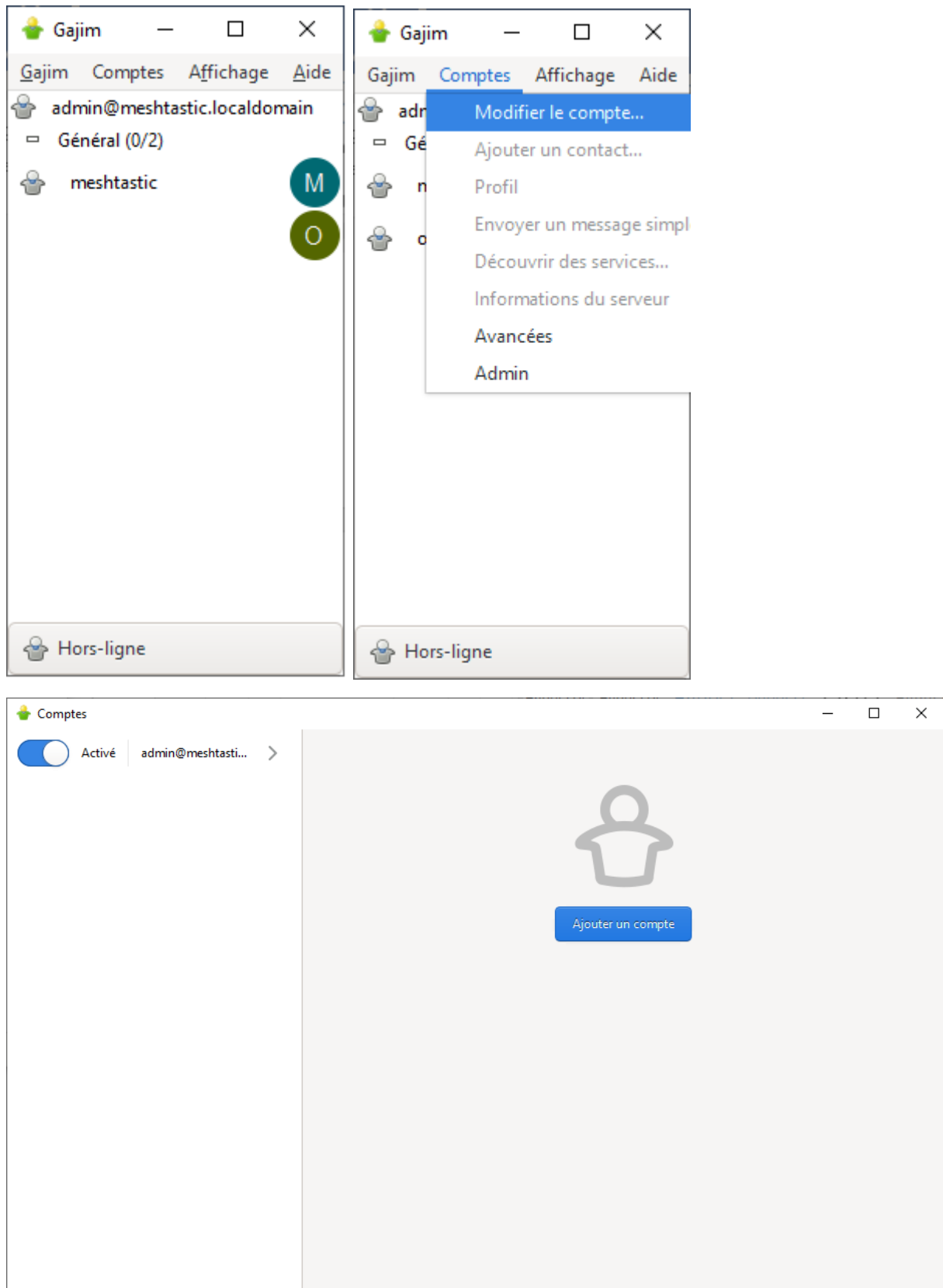
En ligne


Le contact Meshtastic est le gateway.

8. Gajim

Gajim (<https://gajim.org/>) est une application XMPP cliente qui tourne sur Windows et Linux.

Voici comment on la configure :



 Ajouter un compte ×

Bienvenue


Veuillez saisir vos identifiants ou vous inscrire

☒ Paramètres avancés

Connexion

ou bien

S'inscrire

 Paramètres avancés ✕

Paramètres avancées

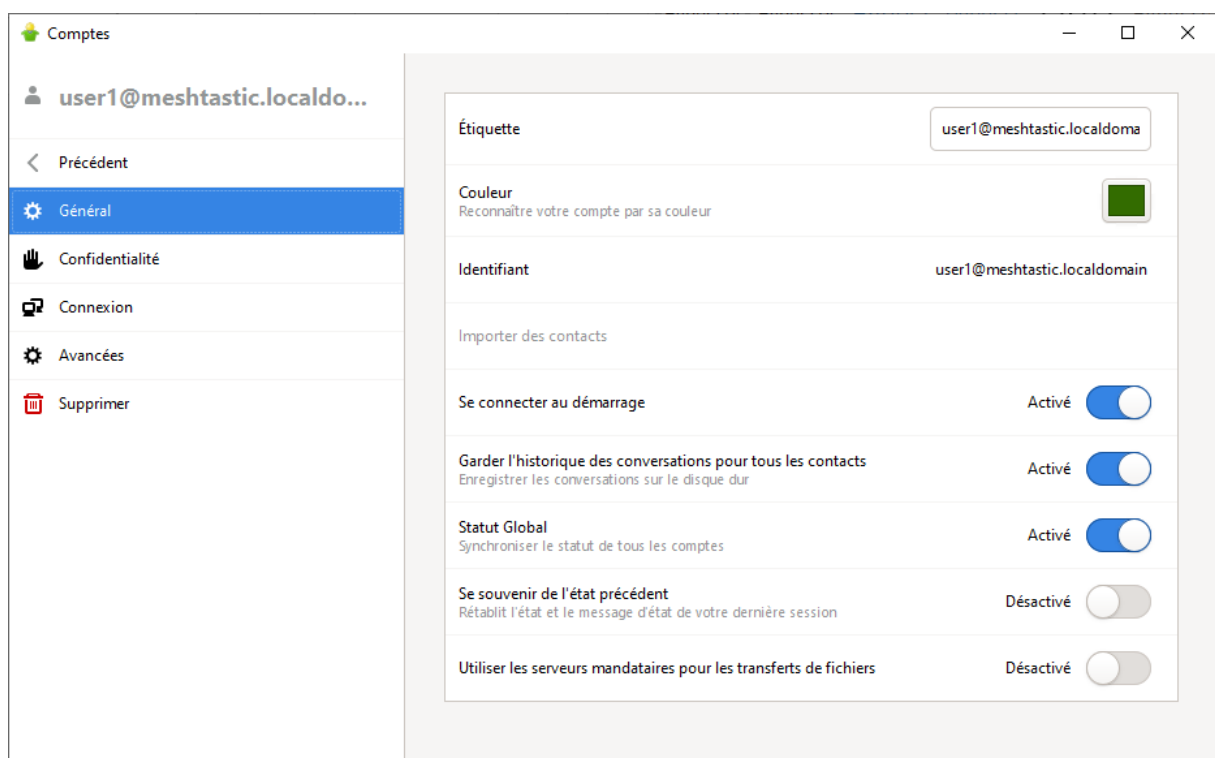
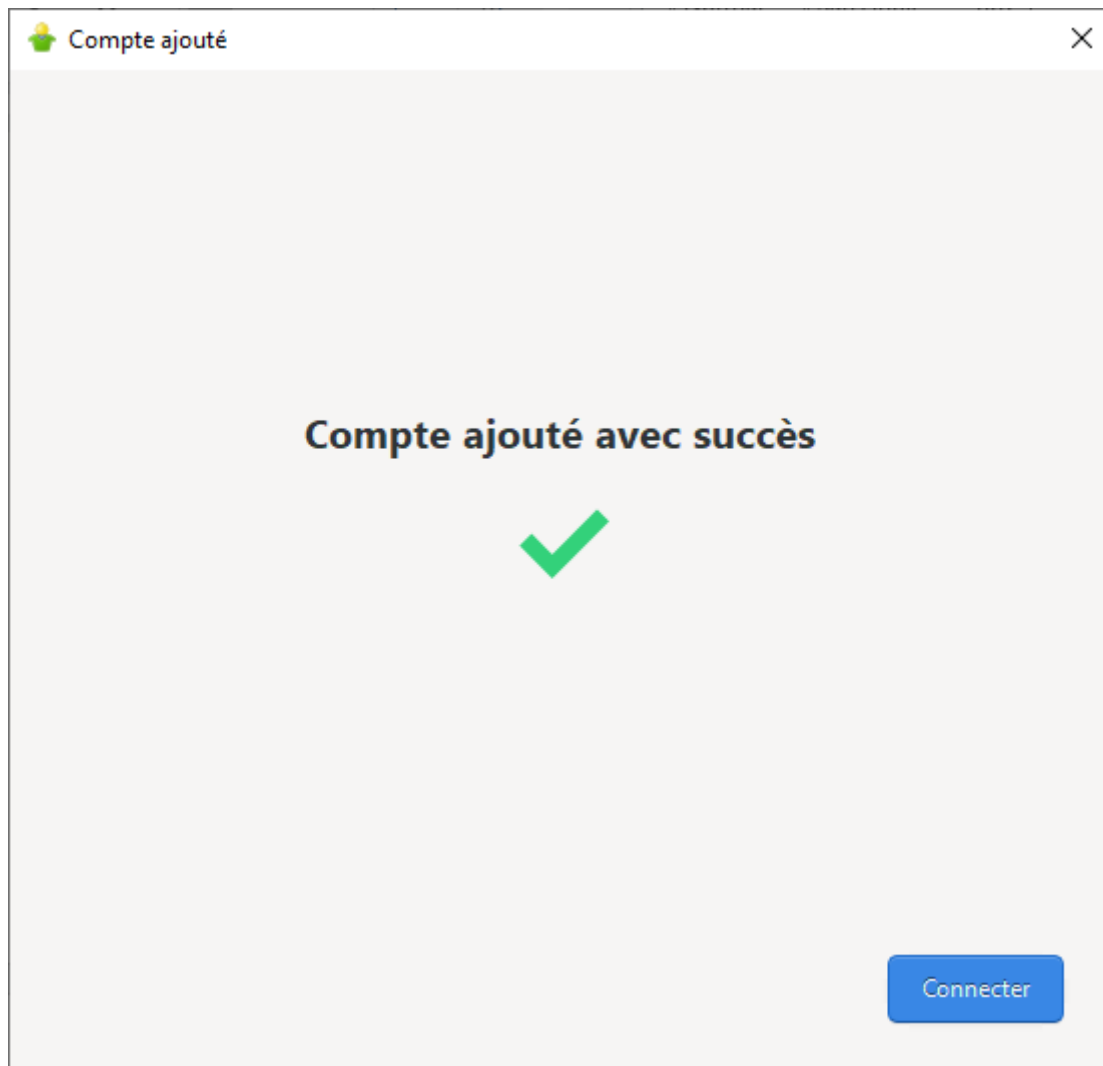
Serveur mandataire Pas de serveur mandataire ▼ ⚙️

Nom d'Hôte meshtastic.localdomain

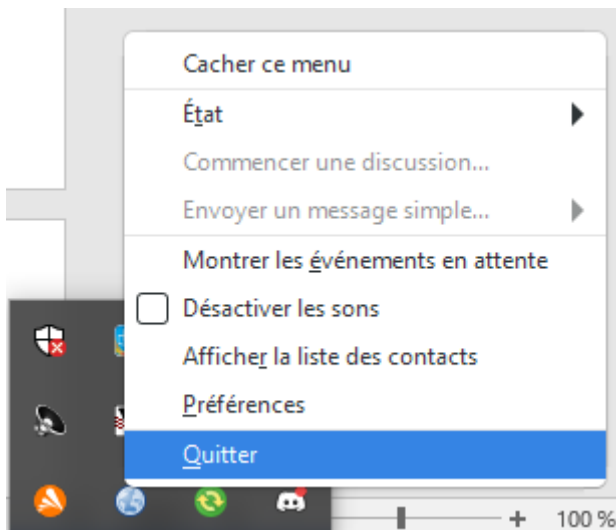
Port 5222

Type PLAIN ▼

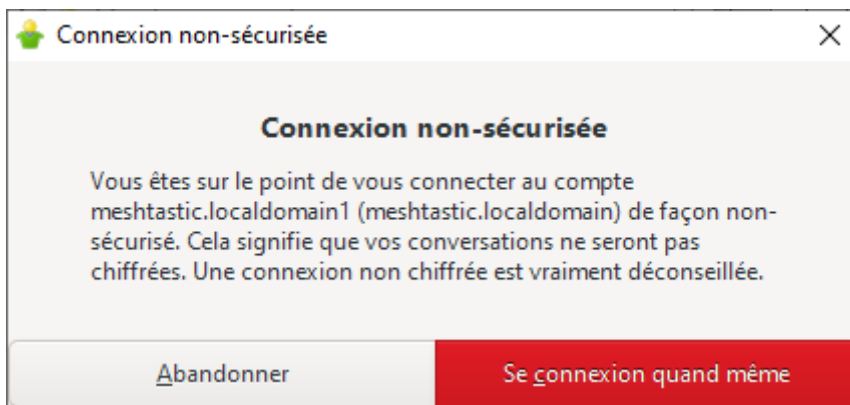
Précédent Se connecter



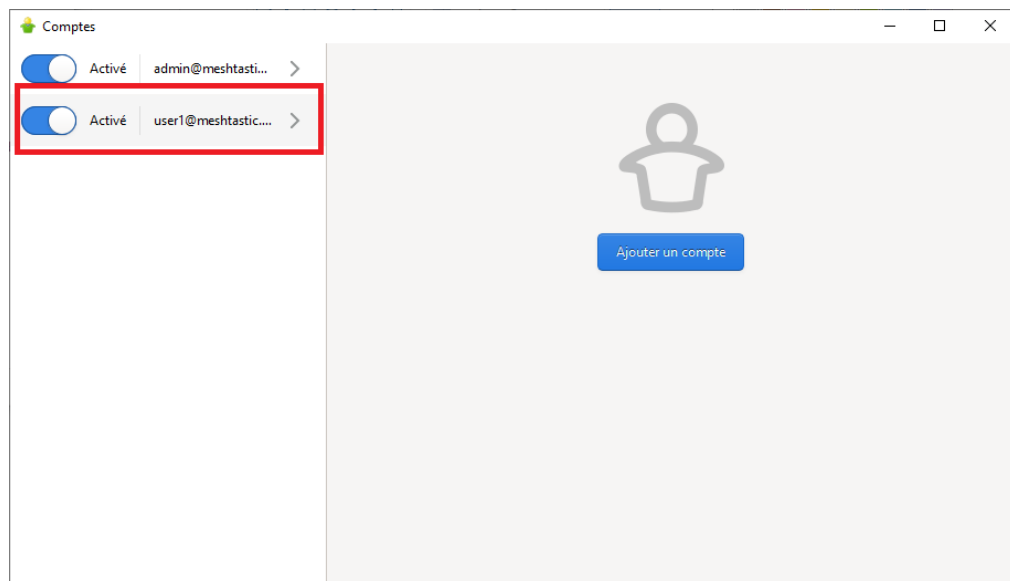
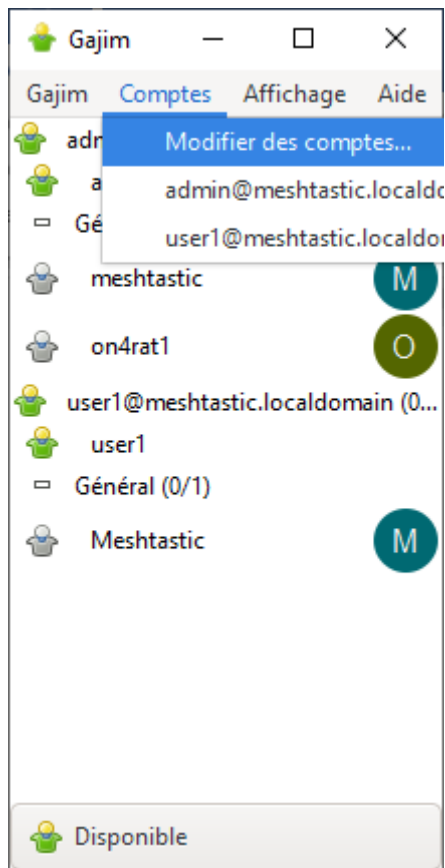
Fermer Gajim :

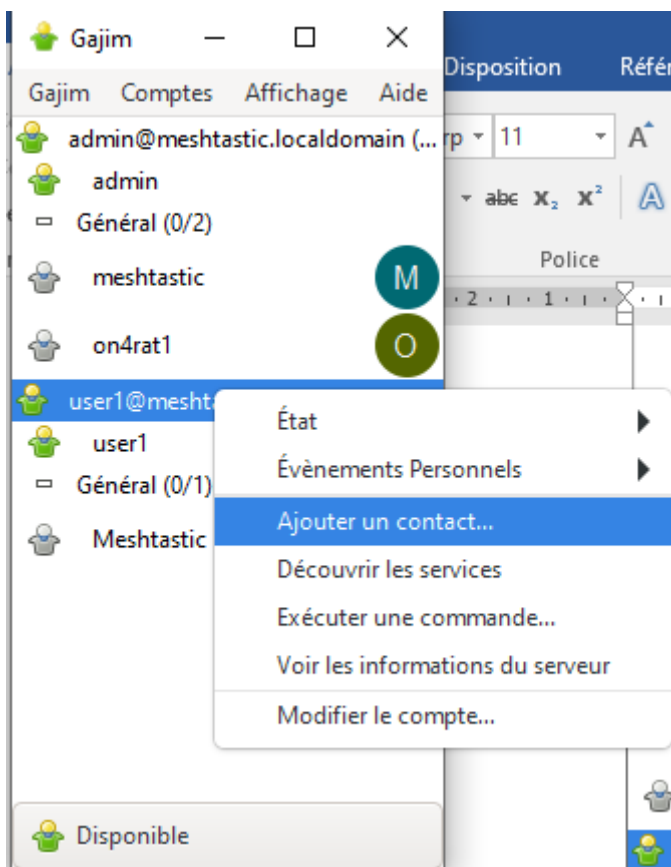
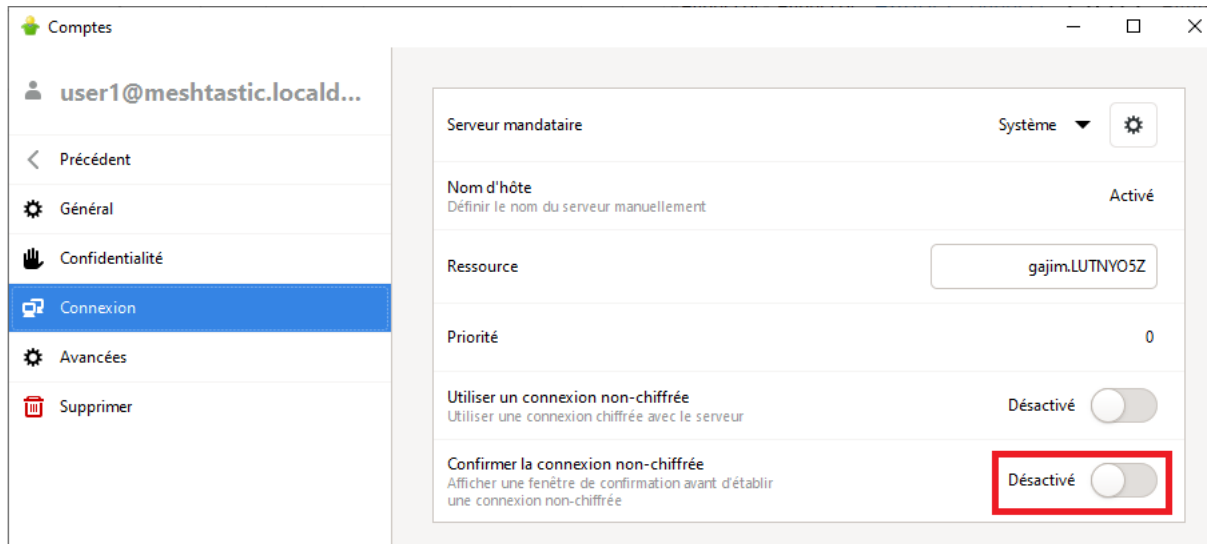



Rouvrir Gajim.



Se connecter quand même.






 Ajouter un contact

Compte

user1@meshtastic.localdomain

Adresse XMPP

meshtastic@meshtastic.localdomain



Surnom

Groupe

Choisissez ou tapez un nouveau nom de ...

☒ Autoriser ce contact à voir mon état

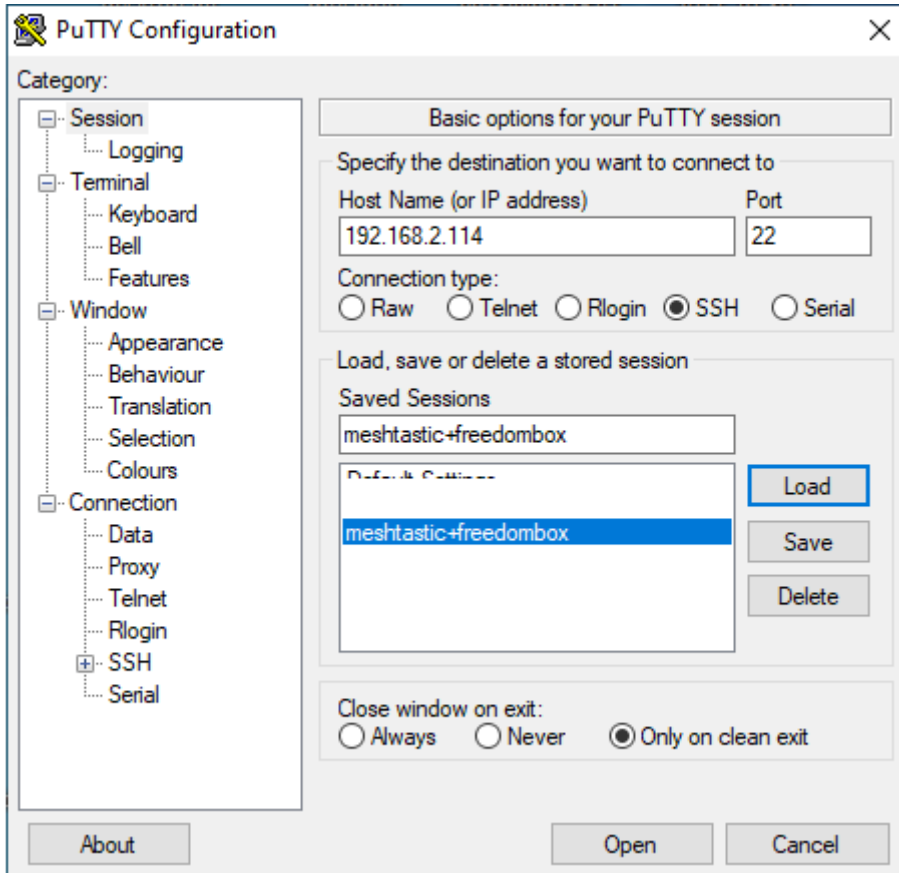
Bonjour, je m'appelle user1. Je souhaiterais vous ajouter à ma liste de contacts.

Annuler

Ajouter

Meshwatch

1. Se connecter par SSH sur le gateway



Login : root / password : admin



2. Mettre en plein écran.
3. Brancher le node Meshtastic sur le port USB (TTGO par exemple).

4. Aller dans le répertoire :

```
/home/admin/meshtalk
```

Commande : `cd /home/admin/meshtalk`

Pour lancer le script gateway commande : `sh meshwatchproc.sh`

```

192.168.2.114 - PuTTY
login as: root
root@192.168.2.114's password:
linux meshtastic 5.10.0-18-armmp #1 SMP Debian 5.10.140-1 (2022-09-02) armv7l

      _ _ _
     / / /
    / / /
   / / /
  / / /
 / / /
/_/_/_

FreedomBox

FreedomBox is a pure blend of Debian GNU/Linux. Web interface is available at
https://localhost/ . FreedomBox manual is available in /usr/share/doc/freedombox
and from the web interface.

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Web console: https://meshtastic.localdomain:9090/ or https://192.168.2.114:9090/

Last login: Sun Oct 30 11:50:34 2022 from 192.168.2.115
root@meshtastic:~# cd /home/admin/meshtalk
root@meshtastic:/home/admin/meshtalk# sh meshwatchproc.sh

```

[illegible]

Tout tourne.

Attention, Meshwatch ne tourne que s'il est affiché sur un écran ! Cela peut être une console SSH ou un écran HDMI.

5. Pour changer les adresses des comptes users dans meshwatch si localdomain ne fonctionne pas :

Dans le répertoire « /home/admin/meshtalk », taper la commande :

nano meshwatch.py

```

GNU nano 5.4 meshwatch.py

#-----
# MeshWatch
#-----
# Author: William McEvoy
# Created: Sept 8 2021
# Purpose: Send and receive messages from a Meshtastic device.
#-----
# Sept 10 2022
# - Adding PYXMP2 client to connect at eJabber server
# - Adding gateway meshtastic to a client eJabber
#-----
# Sept 8, 2021
# - adding formatting and comments
#-----
# Sept 10, 2021
# - added Curses based text interface
# - class and function was created for GPSProbe
#-----
# Sept 10, 2021
# - added recursive function to decode packets of packets
#-----
# Sept 26, 2021
# - renamed project to MeshWatch
#-----
# Oct 01, 2021
# - Added option to send X messages every Y seconds to all nodes for
#   testing the mesh network
# - node information now includes distance in meters from basestation
#-----
# Credit to other projects:
#-----

^G Help      ^O Write Out  ^W Where Is   ^R Cut        ^I Execute    ^G Location   ^U Undo       ^M Set Mark   ^_ To Bracket
^X Exit      ^R Read File  ^N Replace    ^U Paste      ^J Justify    ^O Go To Line ^C Redo       ^M Copy       ^Q Where Was
  
```

Modifier le suffixe à la variable « my_jid » :

```

#This will now be the default behaviour
#parser.add_argument('-r', '--receive', action='store_true', help="recieve and display messages")

# CREDENTIALS OF PYXMP2 CLIENT ACCOUNT TO RECEIVE MESSAGE FROM EJABBER SERVER
#-----
#
# my_jid = login of client account
# your_password = password of account
your_password = 
my_jid = "meshtastic@meshtastic"

#process arguments and assign values to local variables
if (args.send):
  
```

Sauver : CTRL + O.

Quitter : CTRL + X.

Ensuite, taper la commande :

nano sendxmpp.py

Modifier « your_jid » et « target_jid » :

```
#!/usr/bin/python
# -*- coding: utf-8 -*-

#####
#
# SEND MESSAGE TO XMPP SERVER MEMBER (EJABBER) #
#
#####

import sys
import os
from getpass import getpass
from pyxmpp2.simple import send_message

# CREDENTIALS OF PYXMPP2 ACCOUNT

your_jid = "meshtastic@meshtastic" #Client account
your_password = "XXXXXXXXXX" #Password of client account
#target_jid = "admin@meshtastic.localdomain" #Target account who receive messages

#message = sys.argv[1]

#send_message(your_jid, your_password, target_jid, message)

target_jid = "user1@meshtastic"
message = sys.argv[1]

send_message(your_jid, your_password, target_jid, message)
time.sleep(10)
os.system("pyhon3 /home/admin/meshtalk/meshwatch.py")
```

Sauver : CTRL + O.

Quitter : CTRL + X.

Utiliser le client web JSXC

1. Entrer l'adresse de la freedom box



Votre connexion n'est pas privée

Des individus malveillants tentent peut-être de subtiliser vos informations personnelles sur le site **10.2.2.2** (mots de passe, messages ou numéros de carte de crédit, par exemple). [En savoir plus](#)

NET::ERR_CERT_AUTHORITY_INVALID

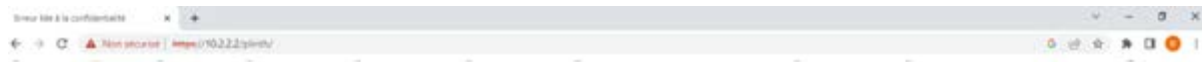


Pour bénéficier du niveau de sécurité le plus élevé de Chrome, [activez la protection renforcée](#).

Paramètres avancés

Revenir en lieu sûr

2. Cliquer sur « Continuer sur le site » :



Votre connexion n'est pas privée

Des individus malveillants tentent peut-être de subtiliser vos informations personnelles sur le site **10.2.2.2** (mots de passe, messages ou numéros de carte de crédit, par exemple). [En savoir plus](#)

NET::ERR_CERT_AUTHORITY_INVALID



Pour bénéficier du niveau de sécurité le plus élevé de Chrome, [activez la protection renforcée](#).

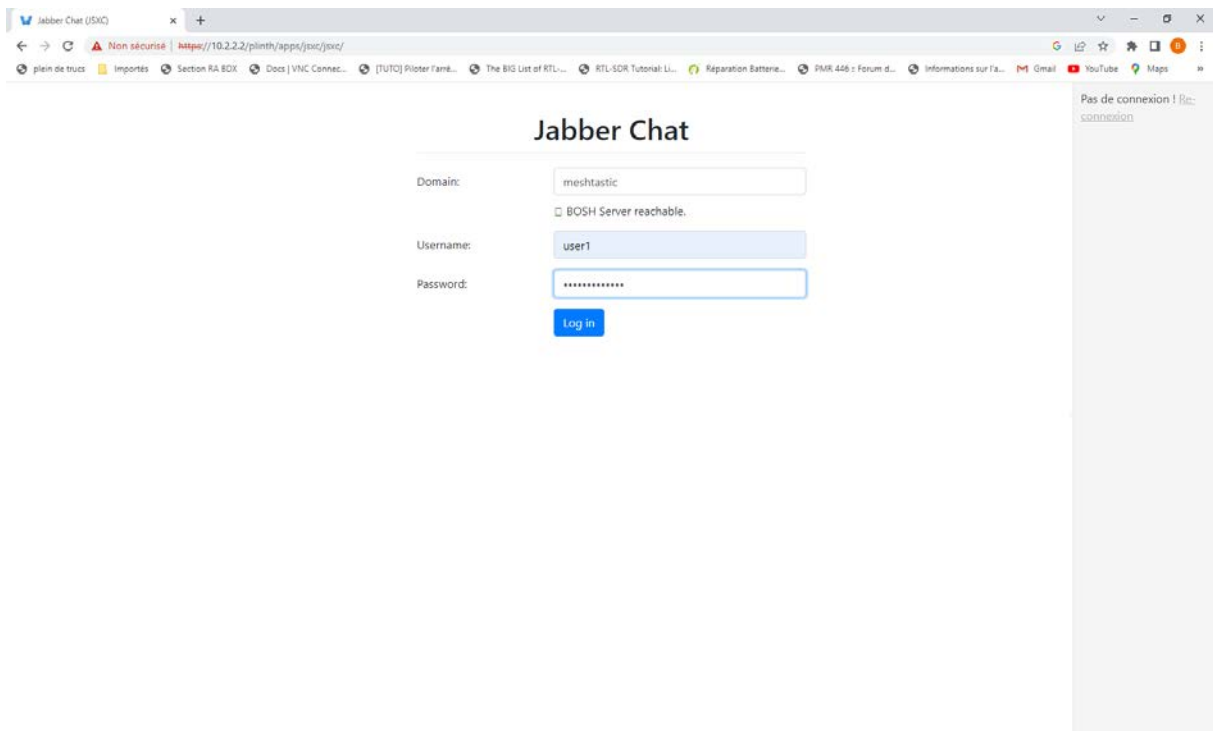
Masquer les paramètres avancés

Revenir en lieu sûr

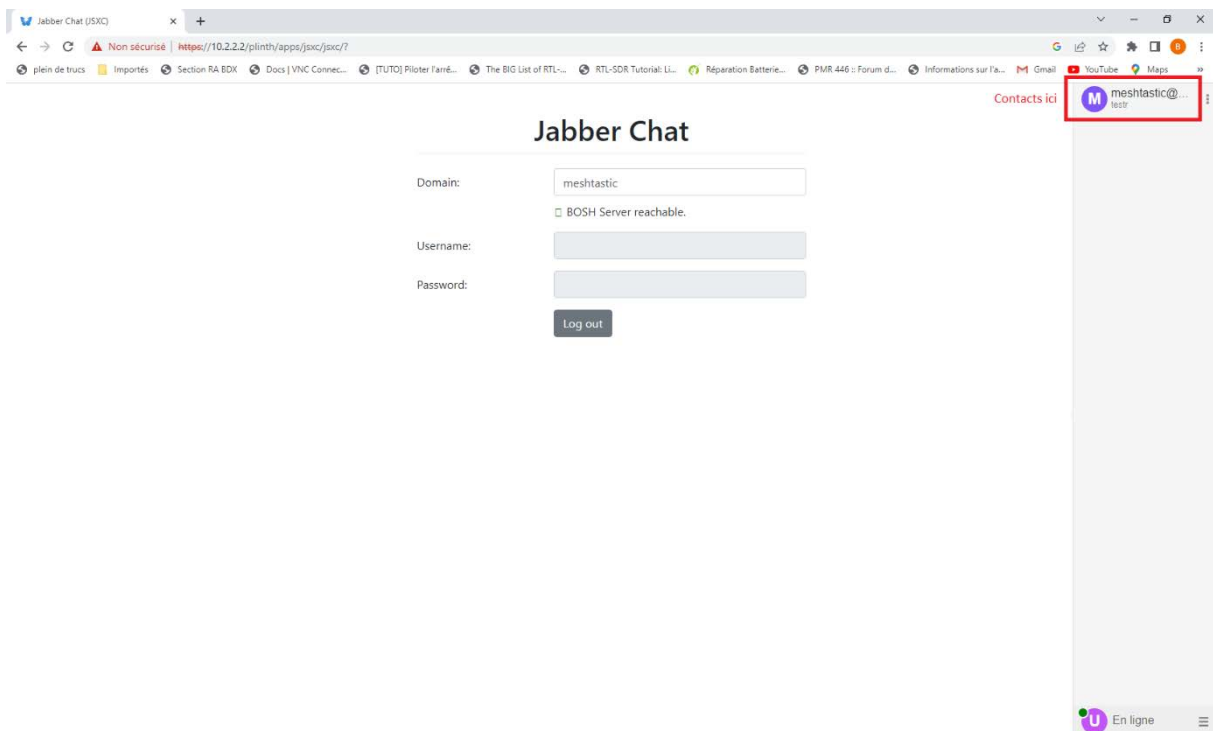
Impossible de vérifier sur le serveur qu'il s'agit bien du domaine **10.2.2.2**, car son certificat de sécurité n'est pas considéré comme fiable par le système d'exploitation de votre ordinateur. Cela peut être dû à une mauvaise configuration ou bien à l'interception de votre connexion par un pirate informatique.

Continuer sur le site 10.2.2.2 (déconseillé)

Cliquer sur JSXC :



Se loguer.

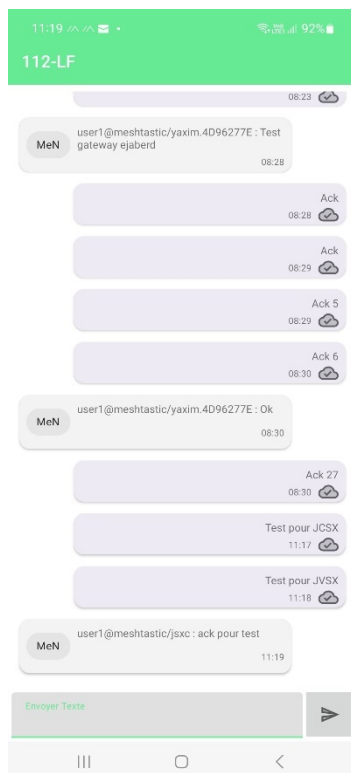


Voici la fenêtre de tchat :

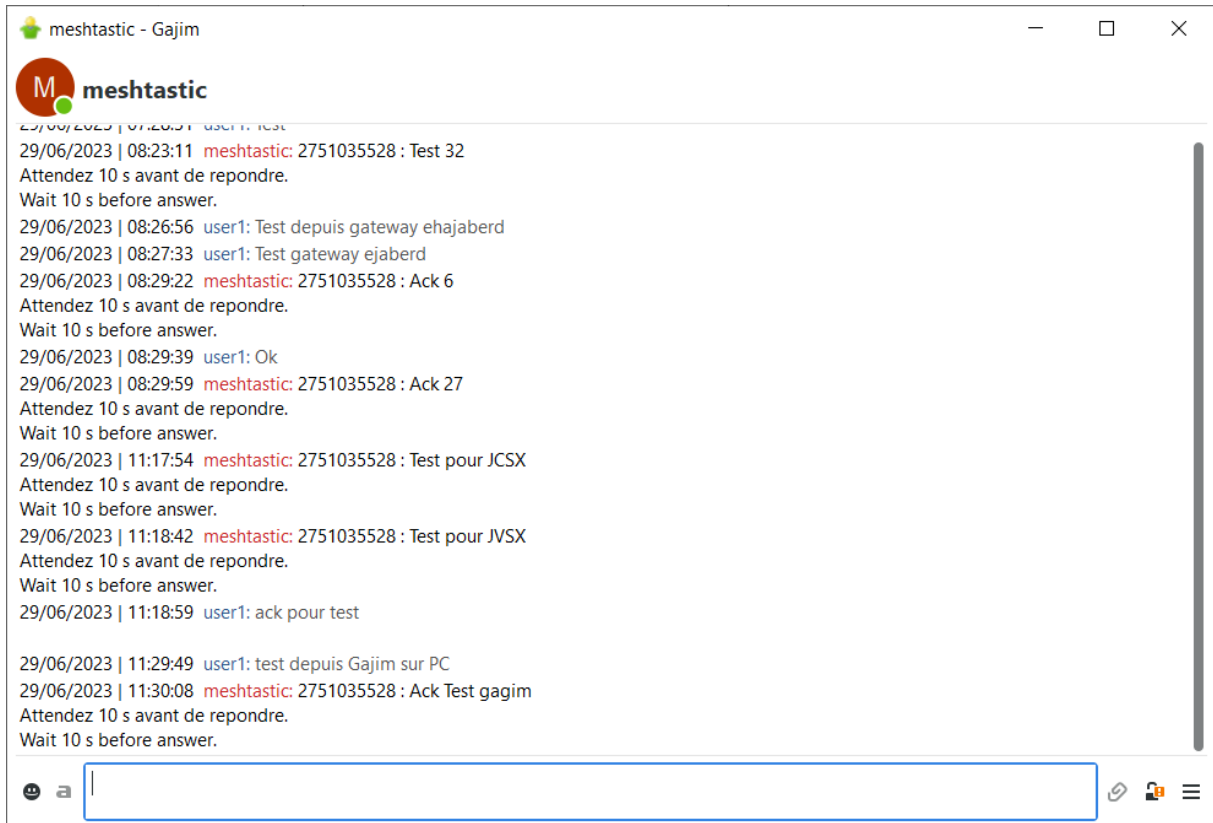
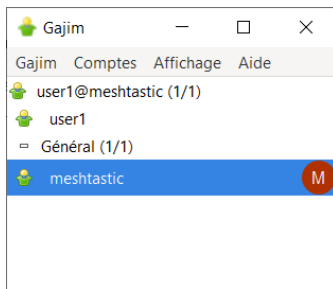


Comme il est bien indiqué, il faut attendre 10 secondes avant de répondre !!!

Message reçu dans Meshtastic :



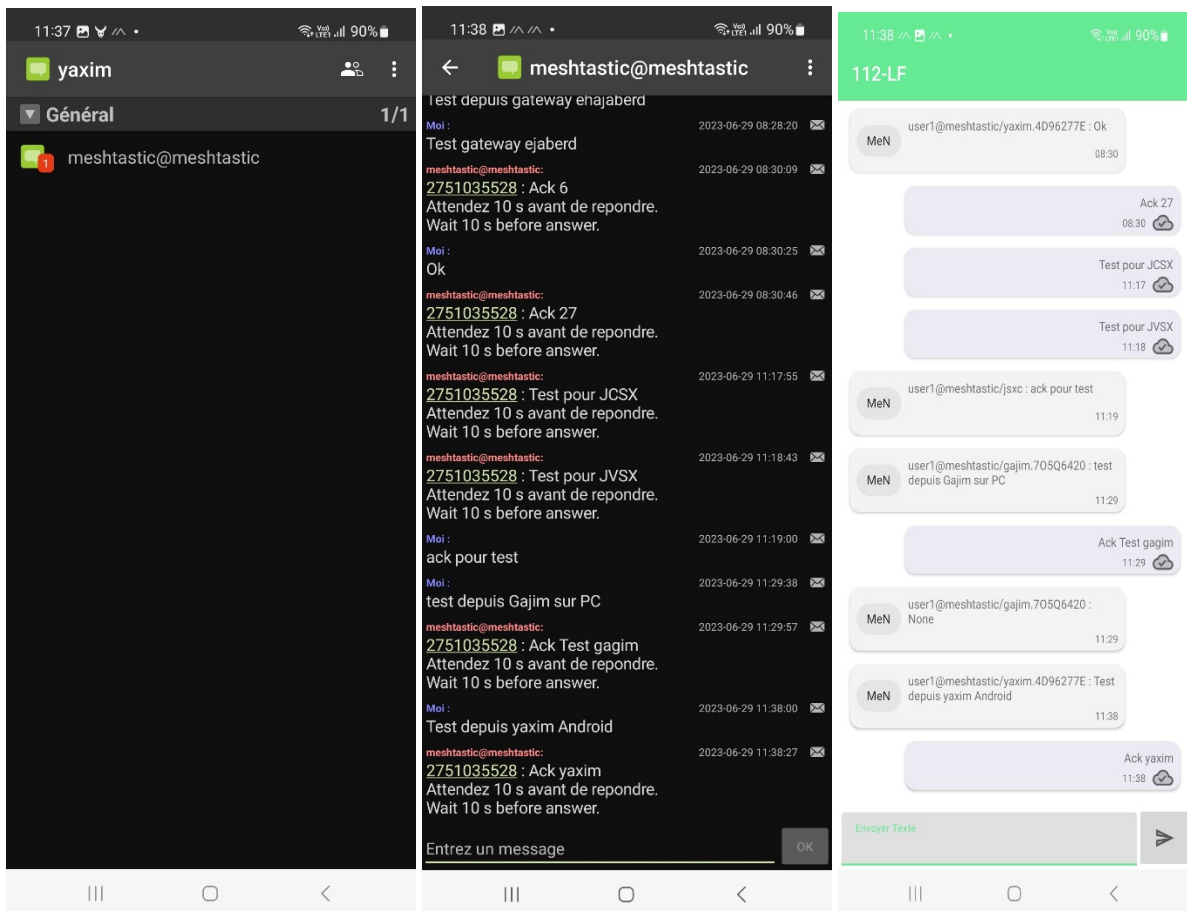
Utiliser le client Gajim sur PC



Utiliser le client Android yaxim

Client android à télécharger sur <https://yaxim.org/>.

Se connecter au WIFI de la Freedom Box.



Paramètres du client :

11:41 90%

← Paramètres du compte

Information de connexion

Identifiant Jabber
user1@meshtastic

Mot de passe du compte
Le mot de passe de votre compte Jabber

Nom affiché
user1

Options de connexion

Message Carbons (XEP-0280)
Demander une copie des messages reçus et envoyés par vos autres clients ☒

Options avancées

Ressource client
yaxim.4D96277E

Priorité du client
Par défaut : 0 (de -128 à 127)

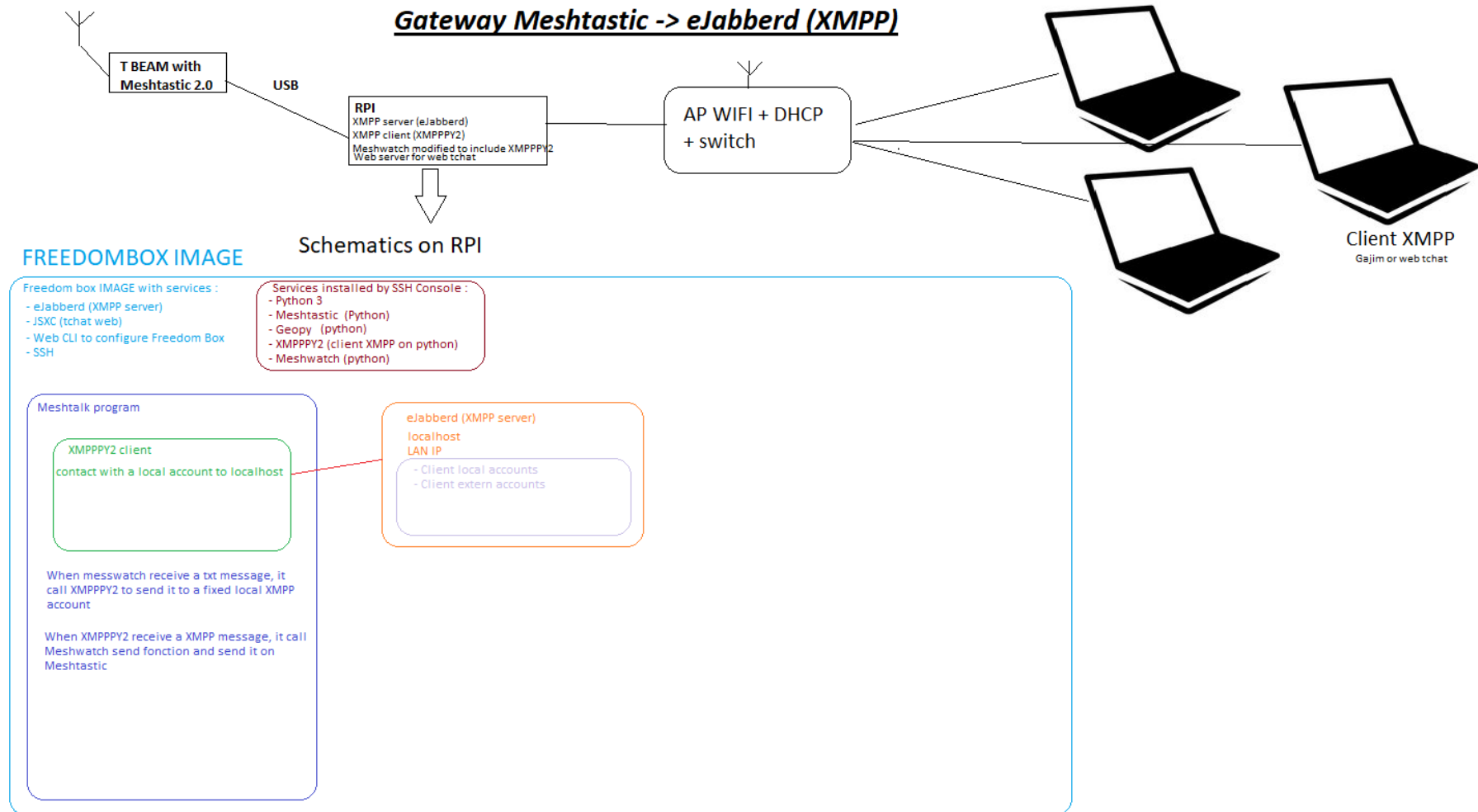
Serveur personnalisé
10.2.2.2

Port du serveur
Par défaut : 5222

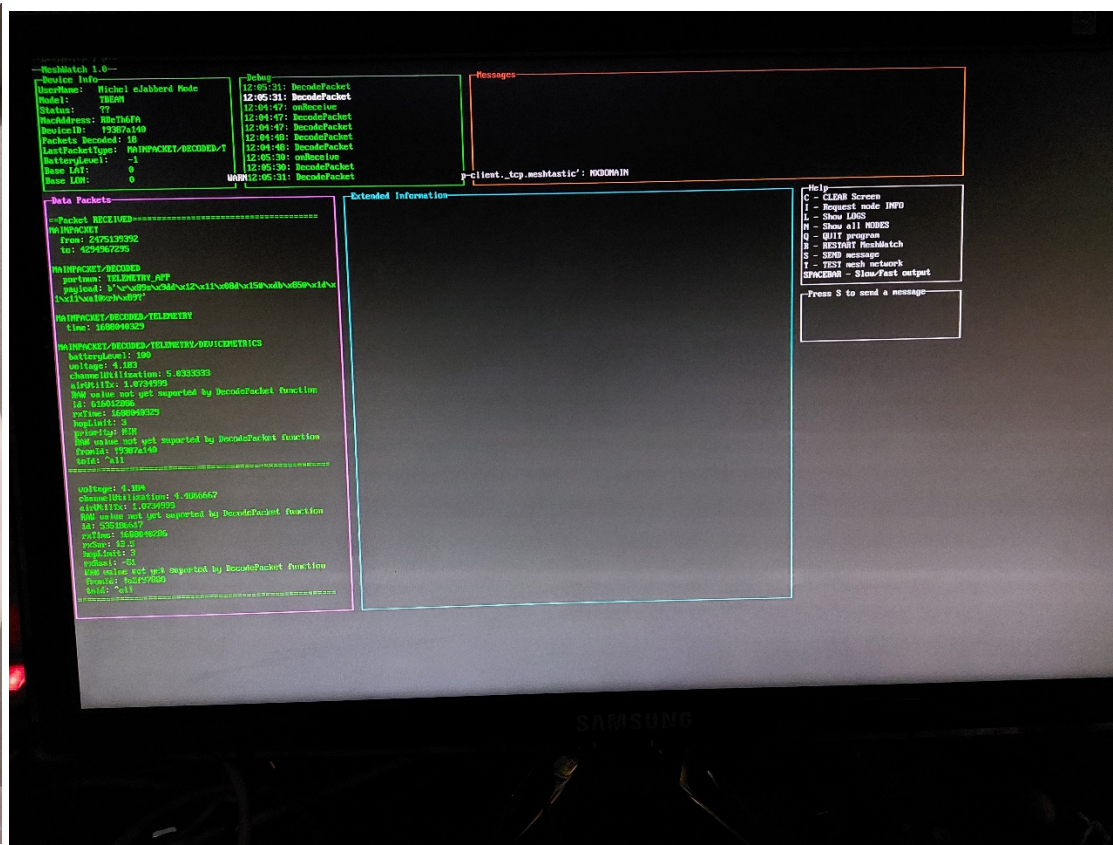
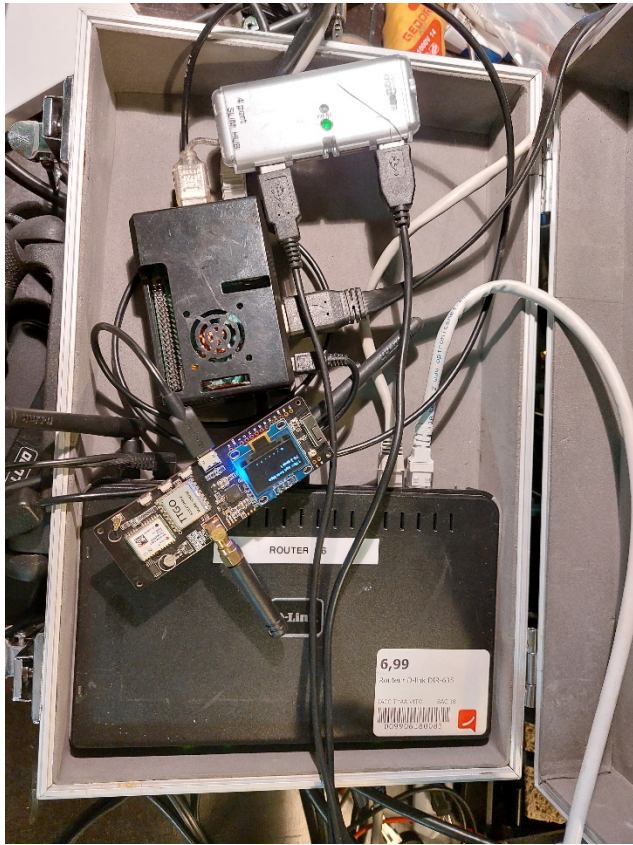
Force TLS/SSL
Toujours chiffrer la connexion au serveur ☐

Bien mettre l'IP de la freedom box à « Serveur personnalisé » !!!

Fonctionnement du programme Meshwatch adapté par ON3MEE



Vue de Meshwatch



A gauche, le projet d'une gobox Meshtastic pour le terrain. On peut y voir un module T BEAM relié en USB à un RPI 3B+ et un routeur WIFI D-LINK. A droite, la vue de Meshwatch sur l'écran.

- **Freedom box**

<https://freedombox.org/fr/>

La distribution Freedom Box pour raspberry pi est l'OS de tout le système. Elle inclut nativement un serveur XMPP et client web de tchat.

- **Meshtastic**

Il faut installer tous les outils Meshtastic ensuite selon la procédure indiquée sur Meshwatch.

- **Meshwatch**

<https://github.com/datagod/meshwatch>

Meshwatch (anciennement Meshtalk) est un outil développé en python pour afficher et envoyer des messages avec un raspberry pi.

Il faut suivre la procédure indiquée sur le github.

Bien mettre à jour les dépendances Meshtastic si vous utilisez la version Meshtastic V2.x !

- **PYXMPP2**

<https://github.com/Jaicus/pyxmpp2>

PYXMPP2 est en client xmpp python. Il a été intégré au code source de Meshwatch afin de faire le lien entre le monde Meshtastic et le serveur XMPP.

Brève description du programme

Tous les programmes se trouvent dans « */home/admin/meshtalk* ».

« *meshwatchproc.sh* » est le programme principal. C'est un watchdog.

« *meshwatch.py* » est le code source de Meshwatch.

« *sendxmpp.py* » est le sous-programme qui sert à envoyer des messages de Meshwatch vers XMPPPY2.

Les codes sources sont disponibles avec les fichiers du projet.

CONCLUSION

Voilà le projet Gateway Meshtastic Freedom Box.

Il peut être utilisé en l'état sur un LAN pour permettre à plusieurs stations de se connecter sur un seul node.

Des évolutions vont être apportées pour enregistrer un log des messages par exemple.

N'hésitez pas à consulter mes autres tutoriels sur www.on3mee.be .

N'hésitez pas aussi à utiliser mon projet parallèle Freedom Box Off Grid :
https://www.on3mee.be/informatique/freedom_box_off_grid.htm