REPUBLIQUE DU BENIN

*****

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

****

UNIVERSITE D'ABOMEY CALAVI
(UAC)

****

ECOLE NATIONALE D'ECONOMIE APPLIQUEE ET DE MANAGEMENT
(ENEAM)

DEPARTEMENT : INFORMATIQUE

SPECIALITE : AUDIT DES SYSTEMES INFORMATIQUES

MATIERE : PROGRAMMATION RESEAU

PROJET :
Système de messagerie instantanée sécurisé

Etudiants :

1- ABOUDOU Enagnon Euloge
2- ADIDO Gbênato Francky Keneth
3- D'ALMEIDA Vincent
4- DANSOU Freddy

Chargé de cours :

Dr. Emery Kouassi ASSOGBA

Année Académique : 2023-2024

Projet : Système de messagerie instantanée sécurisé

Contexte : Une entreprise souhaite déployer une solution de messagerie instantanée sécurisée pour améliorer la communication interne tout en garantissant la confidentialité des échanges.

Objectif : Développer une application de messagerie instantanée sécurisée utilisant le protocole TLS pour chiffrer les communications.

Objectifs spécifiques :

Implémenter une architecture client/serveur avec le protocole TCP pour les connexions fiables.

Intégrer le protocole TLS pour sécuriser les messages échangés entre les clients.

Ajouter des fonctionnalités de chat de groupe et de transfert de fichiers.

Résultats attendus :

Une application de messagerie instantanée sécurisée.

Chiffrement des messages et des fichiers transférés.

Amélioration de la communication interne sans compromission de la sécurité.

Livrables attendus :

Code source de l'application.

Documentation technique.

Guide d'utilisation pour les employés.

Table des matières

LIEN GITHUB

https://github.com/Francky-Keneth/MESSAGERIE_INSTANTANEE

1. Documentation Technique

1.1. Introduction

Description Générale :

L'application de messagerie instantanée sécurisée permet aux utilisateurs de l'entreprise de communiquer en temps réel avec des messages chiffrés utilisant le protocole TLS. Elle supporte également les fonctionnalités de chat de groupe et de transfert de fichiers.

Objectifs:

- Faciliter la communication en temps réel au sein d'une organisation.
- Assurer la confidentialité des communications internes.
- Permettre l'envoi de messages, la gestion des groupes et le transfert de fichiers.

1.2. Spécifications

- **Connexion Fiable** : Utilisation du protocole TCP pour assurer une connexion fiable et la livraison des messages.
- **Messagerie Instantanée Sécurisée** : Utilisation de TLS pour chiffrer toutes les communications entre le client et le serveur, garantissant la confidentialité et l'intégrité des messages échangés.
- **Chat de Groupe** : Les utilisateurs peuvent créer, rejoindre des groupes de discussion ou envoyer de message dans un groupe existants en utilisant des commandes spécifiques.
- **Transfert de Fichiers** : Les utilisateurs peuvent envoyer des fichiers à d'autres utilisateurs ou dans des groupes, en utilisant des commandes spécifiques.
- **Interface Utilisateur CLI** : l'utilisation se fait en ligne de commande (par terminal).
- **Gestion des Connexions simultanées** : Le serveur accepte plusieurs connexions client simultanément.
- **Compatibilité Windows et Linux** : Fichier de code source spécifiques pour la configuration et l'exécution sur les deux systèmes d'exploitation Linux et Windows.

1.3. Architecture du Système

Il est implémenté une architecture client/serveur avec le protocole TCP pour les connexions fiables.

Serveur

- Le serveur est responsable de la gestion des connexions des clients, de la diffusion des messages et de la sécurité des échanges. Il écoute sur un port spécifique et est configuré pour accepter des connexions des clients sur toutes les interfaces réseaux disponibles de la machine.
- Pour chaque client connecté ; un nouveau thread est créé pour gérer la communication
- Les messages reçus d'un client sont diffusés à tous les autres clients connectés ou aux clients du groupe spécifié.

Clients

- Les clients se connectent au serveur en fournissant l'adresse IP et le port du serveur comme arguments de la ligne de commande.
- Deux threads sont créés pour gérer l'envoi et la réception des messages.
- Les messages envoyés par le client sont transmis au serveur, qui les diffuse à tous les destinataires.

1.4. Configuration et Déploiement sur un serveur Linux

Utilisation des certificats TLS dans l'application pour sécuriser les messages

Pour configurer TLS, vous aurez besoin de créer des certificats pour votre serveur et, éventuellement, pour vos clients. Voici un guide simple pour générer des certificats avec openssl

- **Installer OpenSSL** (si ce n'est pas déjà fait) :
  sudo apt-get update
  sudo apt-get install openssl
- **Installer libssl-dev** (bibliothèques de développement OpenSSL qui contiennent les fichiers d'en-tête nécessaires.) **:**
  sudo apt-get install libssl-dev

- **Générer une clé privée pour le serveur :**

  openssl genpkey -algorithm RSA -out server.key
- **Créer une demande de certificat (CSR)** :

Pendant ce processus, vous serez invité à entrer des informations pour le certificat, dont le **Common Name**. Vous pouvez simplement appuyer sur Entrée pour utiliser les valeurs par défaut ou entrer des informations spécifiques. Mais le **Common Name (CN)** du certificat doit correspondre à l'adresse IP ou au nom de domaine sous lequel le serveur sera accessible.

openssl req -new -key server.key -out server.csr

- **Signer le certificat avec la clé privée** : (Cela créera un certificat **server.crt** valable pour 365 jours.)

  openssl x509 -req -days 365 -in server.csr -signkey server.key -out server.crt

Compilation et Exécution

Pour que le serveur soit accessible le fichier source **serveur.c** mis à disposition sera compilé et exécuter sur la machine linux :

gcc -o serveur serveur.c -lssl -lcrypto

./serveur

Notes supplémentaires

- ***Pare-feu et Ports*** *: Assurez-vous que le port indiqué est ouvert sur la machine serveur et que le pare-feu ou le routeur permet les connexions entrantes sur ce port.*
- ***SSL/TLS*** *: Assurez-vous que le certificat server.crt sur le client correspond au certificat utilisé par le serveur.*
- ***Accès réseau*** *: Assurez-vous que les deux machines (client et serveur) peuvent se voir sur le réseau. Vous pouvez tester la connectivité avec des outils comme ping.*

2. Guide d'Utilisateur pour le Client de Messagerie Instantanée Sécurisée

2.1. Introduction

Ce guide explique comment installer, configurer et utiliser le client de messagerie instantanée sécurisée utilisant le protocole TLS pour chiffrer les communications.

Prérequis

- Un serveur de messagerie instantanée sécurisé en cours d'exécution.
- Connaissances de base en ligne de commande.

2.2. Étapes pour lancer le client sur Linux

a) **Préparer l'environnement** :

- Assurez-vous que GCC est installé sur votre système. Sinon vous pouvez l'installer en utilisant le gestionnaire de paquets de votre distribution. Pour Debian/Ubuntu :
  sudo apt-get update
  sudo apt-get install build-essential
- Assurez-vous d'avoir OpenSSL installé sur votre machine, sinon installez-le. Pour Debian/Ubuntu faire :
  sudo apt-get install libssl-dev

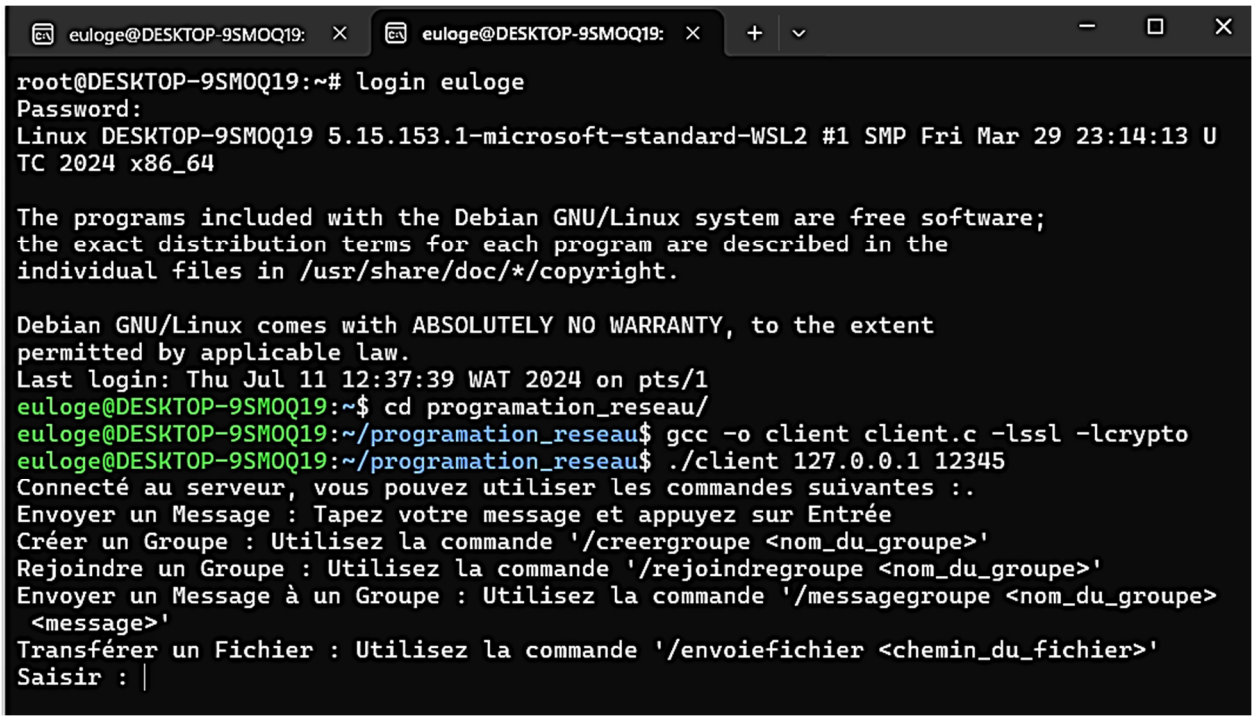
b) **Compiler le client** :

- Ouvrez le terminal et naviguez vers le répertoire contenant votre fichier source client.c.
- Compilez le client ;
  gcc -o client client.c -lssl -lcrypto -lpthread

c) Lancer le client avec les arguments nécessaires :

- Depuis l'invite de commandes, exécutez le client en spécifiant l'adresse IP et le port du serveur distant :

  ./client <server_ip> <server_port>



2.3. Étapes pour lancer le client sur Windows

a) **Préparer l'environnement de développement :**

- Assurez-vous d'avoir installé un compilateur C comme MinGW (Minimalist GNU for Windows) Une alternative recommandée est d'utiliser MSYS2, qui fournit un environnement plus à jour pour les outils GNU sous Windows.
- Assurez-vous d'avoir OpenSSL installé sur votre machine. Vous pouvez télécharger une version précompilée d'OpenSSL pour Windows.

Installation d'un compilateur si nécéssaire (MSYS2) :

- Téléchargez MSYS2 depuis msys2.org
- Suivez les instructions d'installation.
- Ouvrez le terminal MSYS2 et exécuter les commandes suivantes :
  pacman -Syu
  pacman -S mingw-w64-x86_64-gcc
  pacman -S mingw-w64-x86_64-openssl

- Ajoutez le chemin vers le répertoire bin de msys64 (par exemple 'C:\msys64\mingw64\bin') à votre variable d'environnement **PATH** pour pouvoir utiliser gcc depuis n'importe quelle invite de commandes.

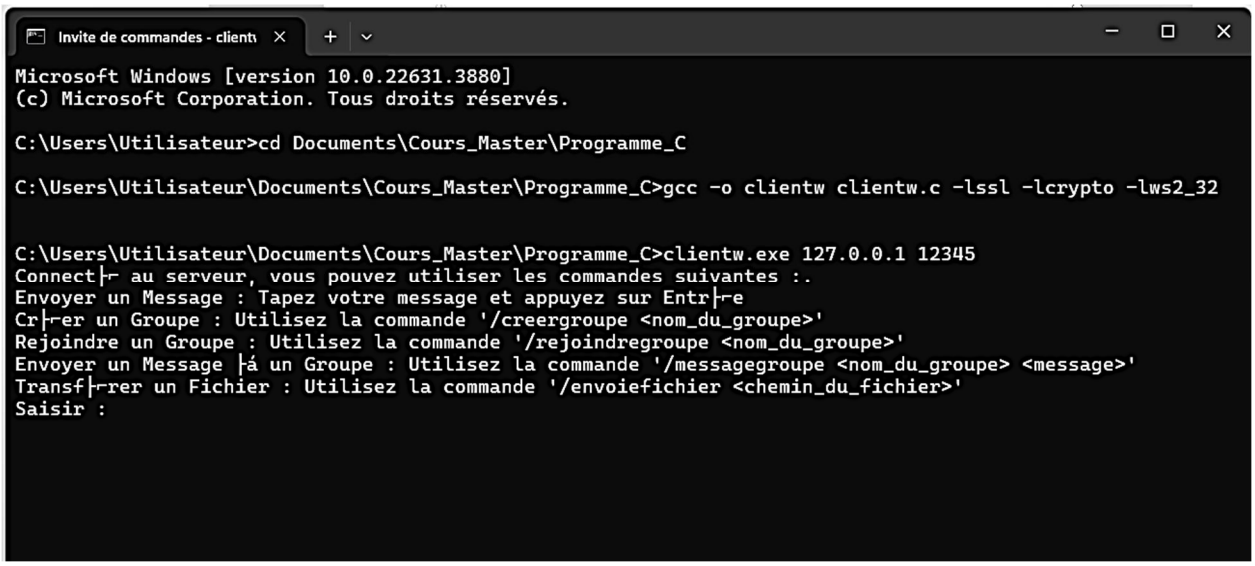
Installation une version précompilée d'OpenSSL si nécéssaire :

- Téléchargez la version la version appropriée pour votre système (Win32 ou Win64)
- Exécutez l'installateur téléchargé.
- Lors de l'installation, choisissez d'installer OpenSSL dans un répertoire accessible, par exemple C:\OpenSSL-Win64.
- Assurez-vous de cocher l'option pour ajouter les fichiers binaires OpenSSL à la variable d'environnement PATH lors de l'installation.

b) **Adapter le code source du client pour windows** :
- Remplacer le fichier d'en-tête arpa/inet.h par leur équivalent windows. Parceque sous Windows, les fonctions de réseau sont généralement incluses dans les en-têtes winsock2.h et ws2tcpip.h
- Ajouter l'initialisation de la bibliothèque Winsock au début du main
- Ajoutez l'appel à WSACleanup avant de quitter le programme

c) **Compiler le client** :
- Ouvrez une invite de commandes et naviguez vers le répertoire contenant votre fichier source clientw.c.
- Compilez le client ;
  gcc -o clientw clientw.c -lssl -lcrypto -lws2_32
- L'option -lws2_32 est nécessaire pour lier la bibliothèque Windows Sockets 2, qui est requise pour les applications réseau sous Windows.

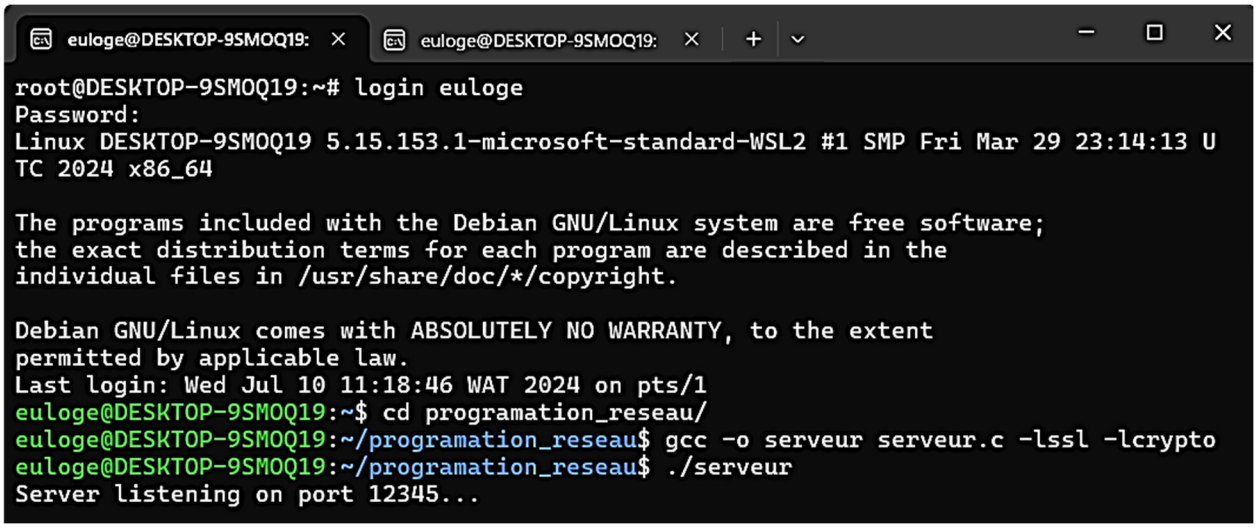
d) **Lancer le client avec les arguments nécessaires** :
- Depuis l'invite de commandes, exécutez le client en spécifiant l'adresse IP et le port du serveur distant : client.exe <server_ip> <server_port>

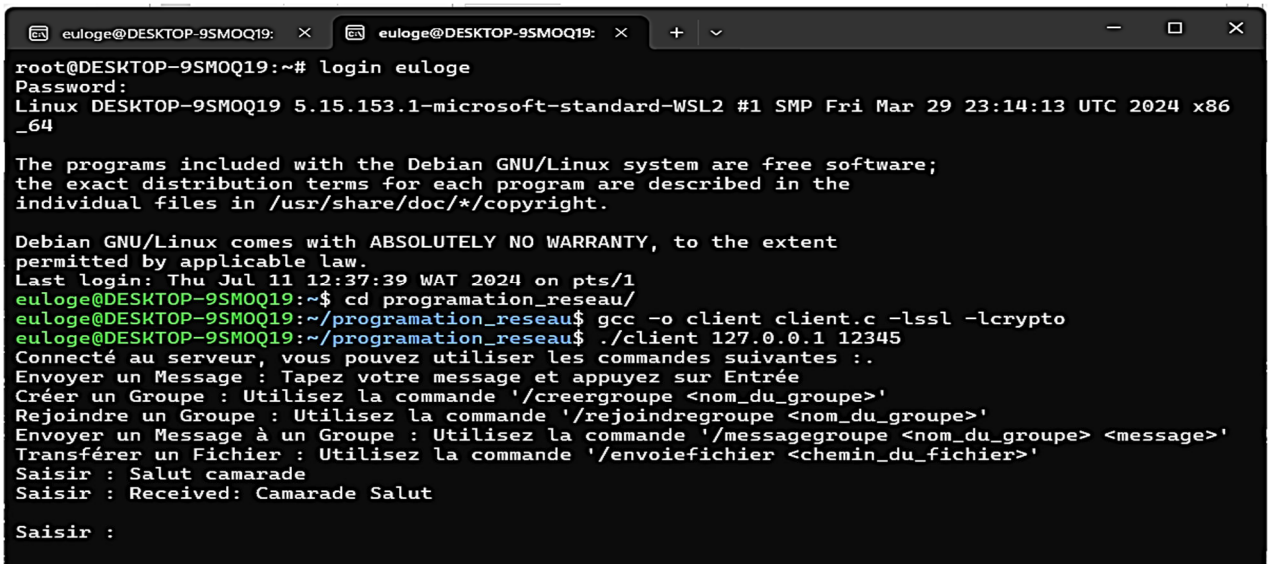


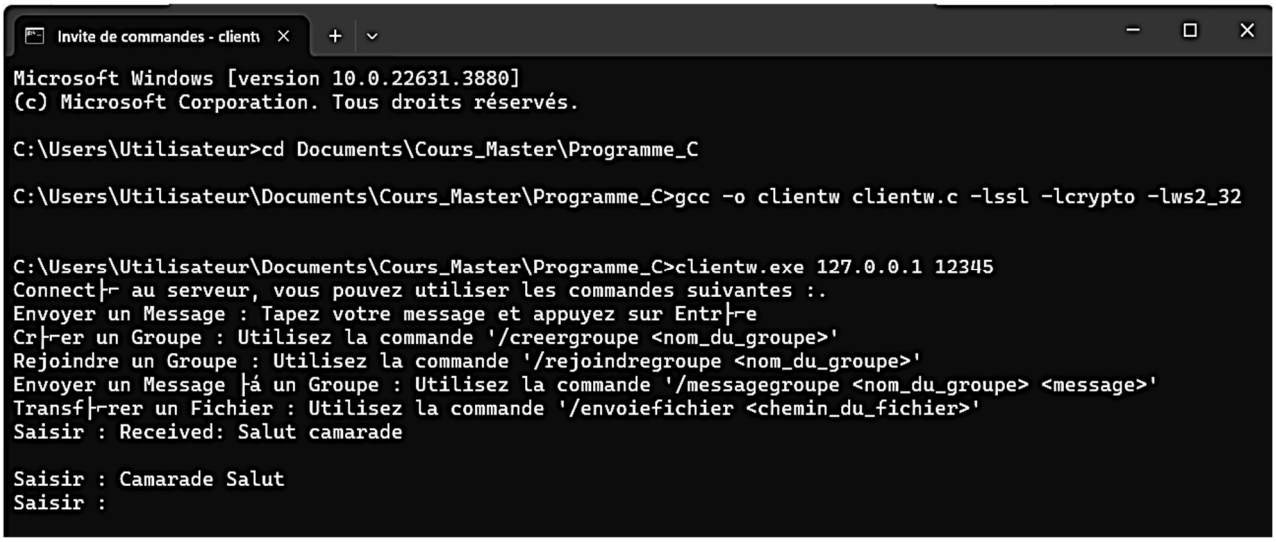
e) Connexion au Serveur et utilisation du Client

Le client tentera de se connecter au serveur de messagerie instantanée sécurisé. Une fois connecté, vous pouvez utiliser les commandes suivantes :

- **Envoyer un Message :** Tapez votre message et appuyez sur Entrée.
- **Créer un Groupe :** Utilisez la commande /creergroupe <nom_du_groupe>.
- **Rejoindre un Groupe :** Utilisez la commande /rejoindregroupe <nom_du_groupe>.
- **Envoyer un Message à un Groupe :** Utilisez la commande /messagegroupe <nom_du_groupe> <message>.
- **Transférer un Fichier :** Utilisez la commande /envoiefichier <chemin_du_fichier>




euloge@DESKTOP-9SMOQ19:
euloge@DESKTOP-9SMOQ19:
root@DESKTOP-9SMOQ19:~# login euloge
Password:
Linux DESKTOP-9SMOQ19 5.15.153.1-microsoft-standard-WSL2 #1 SMP Fri Mar 29 23:14:13 UTC 2024 x86_64
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.
Last login: Thu Jul 11 12:37:39 WAT 2024 on pts/1
euloge@DESKTOP-9SMOQ19:~$ cd programation_reseau/
euloge@DESKTOP-9SMOQ19:~/programation_reseau$ gcc -o client client.c -lssl -lcrypto
euloge@DESKTOP-9SMOQ19:~/programation_reseau$ ./client 127.0.0.1 12345
Connecté au serveur, vous pouvez utiliser les commandes suivantes :.
Envoyer un Message : Tapez votre message et appuyez sur Entrée
Créer un Groupe : Utilisez la commande '/creergroupe <nom_du_groupe>'
Rejoindre un Groupe : Utilisez la commande '/rejoindregroupe <nom_du_groupe>'
Envoyer un Message à un Groupe : Utilisez la commande '/messagegroupe <nom_du_groupe> <message>'
Transférer un Fichier : Utilisez la commande '/envoiefichier <chemin_du_fichier>'
Saisir : Salut camarade
Saisir : Received: Camarade Salut
Saisir :


Invite de commandes - clienti
Microsoft Windows [version 10.0.22631.3880]
(c) Microsoft Corporation. Tous droits réservés.
C:\Users\Utilisateur>cd Documents\Cours_Master\Programme_C
C:\Users\Utilisateur\Documents\Cours_Master\Programme_C>gcc -o clientw clientw.c -lssl -lcrypto -lws2_32
C:\Users\Utilisateur\Documents\Cours_Master\Programme_C>clientw.exe 127.0.0.1 12345
Connect├r au serveur, vous pouvez utiliser les commandes suivantes :.
Envoyer un Message : Tapez votre message et appuyez sur Entr├re
Cr├er un Groupe : Utilisez la commande '/creergroupe <nom_du_groupe>'
Rejoindre un Groupe : Utilisez la commande '/rejoindregroupe <nom_du_groupe>'
Envoyer un Message ├á un Groupe : Utilisez la commande '/messagegroupe <nom_du_groupe> <message>'
Transf├rer un Fichier : Utilisez la commande '/envoiefichier <chemin_du_fichier>'
Saisir : Received: Salut camarade
Saisir : Camarade Salut
Saisir :