



Cycle de Formation des Ingénieurs en Informatique

Rapport de projet python

Application de chat messagerie en utilisant openssl openssl en s'appuyant sur LDAP

Elaboré Par

oumar Traore

Mohamed Lemine Ikhalih

]

Année Universitaire : 2022/2023

Table des matières

| | |
|---|-----|
| TABLE DES MATIERES | II |
| LISTE DES FIGURES | III |
| CHAPITRE I. CONTEXTE GENERALE DU PROJET | 2 |
| I.1 PROBLEMATIQUE..... | 2 |
| I.2 ARCHITECTURE CLIENT-SERVEUR | 2 |
| I.3 AUTHENTIFICATION CLIENT VIA CERTIFICAT D’AUTORITE AUPRES OPENSRL | 3 |
| I.4 SERVEUR LDAP | 3 |
| I.5 LES PACKAGES REQUIS POUR LE DEPLOIEMENT DES SERVICES | 4 |
| CHAPITRE II. SIMULATION ET RESULTAT | 5 |
| II.1 PROTOTYPE DE L’APPLICATION..... | 5 |
| II.1.1 authentifier deux client en saisir le nom d'utilisateur et le mot de passe..... | 5 |
| II.1.2 Authentifier les utilisateurs après le serveur openldap | 6 |
| II.2 LES FONCTIONNALITES DU SERVEUR LDAP AVEC LES CLIENTS EN GESTIONNAIRES DE CERTIFICAT D’AUTORITE EN APPLICATION DE CHAT:..... | 16 |
| CONCLUSION GENERALE | 19 |

Liste des figures

FIGURE 1:ARCHITECTURE CLIENT-SERVEUR2

FIGURE 2:CONNEXION DU PREMIERE UTILISATEUR.....5

FIGURE 3:CONNEXION DU DEUXIEME UTILISATEUR.....6

FIGURE 4:BASEDN.LDIF.....7

INTRODUCTION GENERALE

Une application de chat de messagerie instantanée est une application logicielle qui permet aux utilisateurs de communiquer par écrit en temps réel. Les applications de chat de messagerie instantanée offrent la possibilité d'envoyer des messages texte, d'envoyer des photos et des vidéos, de partager des fichiers et de participer à des conversations de groupe. La plupart des applications de chat de messagerie instantanée offrent également des fonctionnalités supplémentaires, notamment des appels vidéo et audio, des notifications push, des messages programmés, des appels vocaux et des options de sécurité avancées.

L'utilisation d'une application de messagerie pour l'authentification de serveur LDAP en utilisant une autorité de certification OpenSSL peut être très utile. Cela permet aux administrateurs de s'assurer que les données stockées sur le serveur LDAP sont sécurisées et sont protégées contre toute tentative d'accès non autorisé. La mise en œuvre d'une autorité de certification OpenSSL peut être effectuée en utilisant une application de messagerie compatible avec le protocole TLS. L'application de messagerie peut être utilisée pour établir une connexion sécurisée entre le serveur LDAP et le client. Une fois la connexion établie, l'application de messagerie peut être utilisée pour vérifier l'authentification du serveur LDAP et s'assurer que les données stockées sur le serveur sont sécurisé

Chapitre I. Contexte générale du projet

I.1 Problématique

Notre projet consiste à Sécuriser les échanges électroniques entre entreprises, particuliers et administrations.

Les PKI (Public Key Infrastructure) ou infrastructures à clef publique constituent la meilleure réponse technique, organisationnelle et juridique au problème de la sécurité des échanges électroniques : messagerie, paiement électronique, etc. Fondées sur des techniques de cryptographie asymétrique, l'utilisation des PKI permet de garantir l'authentification, l'intégrité, la confidentialité et la non-répudiation des documents échangés entre partenaires en authentifiant les communications via openssl s'appuyant sur openldap .

I.2 Architecture client-serveur

une architecture client-serveur représente l'environnement dans lequel des applications de machines clientes communiquent avec des applications de machines de type serveurs.

L'exemple classique est le navigateur Web d'un client qui demande (on parle de "requête") le contenu d'une page Web à un serveur Web qui lui renvoie le résultat (on parle de « réponse »).

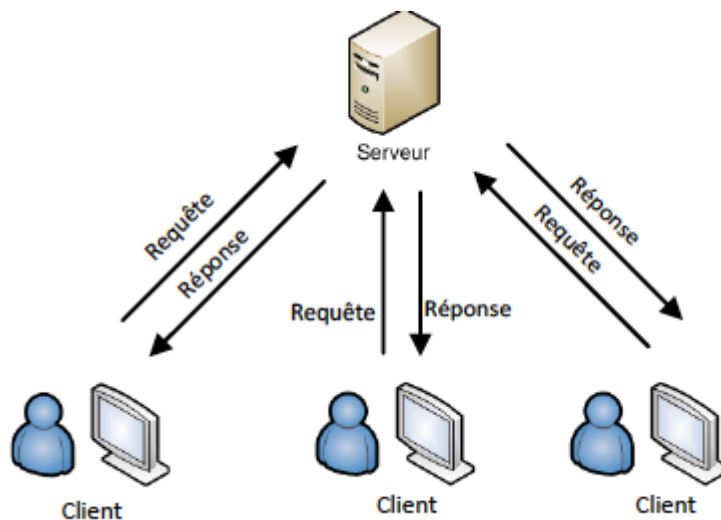


Figure 1:Architecture client-serveur

I.3 Authentification client via certificat d'autorité auprès openssl

Le processus d'authentification client via certificat d'autorité en OpenSSL se déroule comme suit :

- 1) Le client envoie sa demande de certificat à l'autorité de certification (CA).
- 2) L'autorité de certification vérifie les informations fournies par le client et délivre un certificat.
- 3) Le client reçoit le certificat et l'utilise pour générer une clé publique et privée.
- 4) Le client envoie sa clé publique à l'autorité de certification.
- 5) L'autorité de certification vérifie la clé publique et délivre un certificat signé à l'utilisateur.
- 6) Le client envoie le certificat signé à l'autorité de certification.
- 7) L'autorité de certification vérifie le certificat et délivre une clé publique et privée au client.
- 8) Le client utilise sa clé publique et privée pour s'authentifier aup

I.4 Serveur ldap

OpenLDAP est un serveur d'annuaire open source qui permet aux organisations d'accéder, de gérer et de partager des informations à travers un réseau. Il fournit une solution d'annuaire distribué, sécurisée et extensible pour stocker et gérer les informations utilisateur et ressources des réseaux. OpenLDAP peut être utilisé pour stocker des informations telles que les noms d'utilisateur et les mots de passe, pour faciliter la gestion des groupes et des rôles, pour le partage de fichiers et de dossiers, et pour le contrôle des accès à des applications et à des serveurs. De plus, il peut être utilisé pour synchroniser les informations des utilisateurs et des groupes entre plusieurs systèmes et pour gérer l'accès à des systèmes distants. OpenLDAP est une solution puissante et fiable pour les organisations qui souhaitent mettre en œuvre une gestion centralisée de tous les objets des entreprises .

I.5 les packages requis pour le deployment des services

Un package est un logiciel qui peut être installé sur un système d'exploitation. Il s'agit d'un ensemble de fichiers compilés ou de scripts qui peuvent être utilisés pour ajouter des fonctionnalités supplémentaires à un système d'exploitation donné. Les packages peuvent être installés via un gestionnaire de paquets, un programme de gestion de logiciels ou un terminal.

OpenSSL :est un logiciel libre qui fournit des implémentations sécurisées de la couche de transport sécurisé (TLS) et de la couche de session sécurisée (SSH). Il prend en charge les protocoles SSL (Secure Sockets Layer) et TLS (Transport Layer Security) et peut être utilisé pour crypter des communications sur des systèmes réseau. Il est largement utilisé pour la messagerie électronique, le transfert de fichiers

PyCryptodome est une bibliothèque Python qui fournit des services cryptographiques sécurisés. Il s'agit d'un fork de PyCrypto et comporte plusieurs améliorations, notamment la suppression d'une grande partie du code hérité, une meilleure compatibilité avec la dernière version de Python, une meilleure prise en charge de Windows, des algorithmes plus sécurisés et une vitesse améliorée. PyCryptodome prend en charge une large gamme d'algorithmes de chiffrement, y compris les chiffrements symétriques et asymétriques, les hachages et les signatures numériques. Il convient aux utilisateurs novices et avancés et fournit un ensemble complet de fonctions cryptographiques.(**pip install pycryptodome**)

Dans kali linux et les systèmes basés sur Debian, vous pouvez installer OpenLDAP Server à l'aide de la commande apt-get :sudo apt-get install slapd

Cette commande installera le serveur LDAP et tous les packages nécessaires pour le faire fonctionner. Une fois l'installation terminée, vous devrez configurer le serveur LDAP. Pour ce faire, vous pouvez utiliser un outil de configuration graphique appelé «démon de configuration de LDAP» :

```
sudo dpkg-reconfigure slapd
```

Cet outil vous guidera à travers le processus de configuration du serveur LDAP. Une fois le processus de configuration terminé, le serveur LDAP sera prêt à l'emploi.

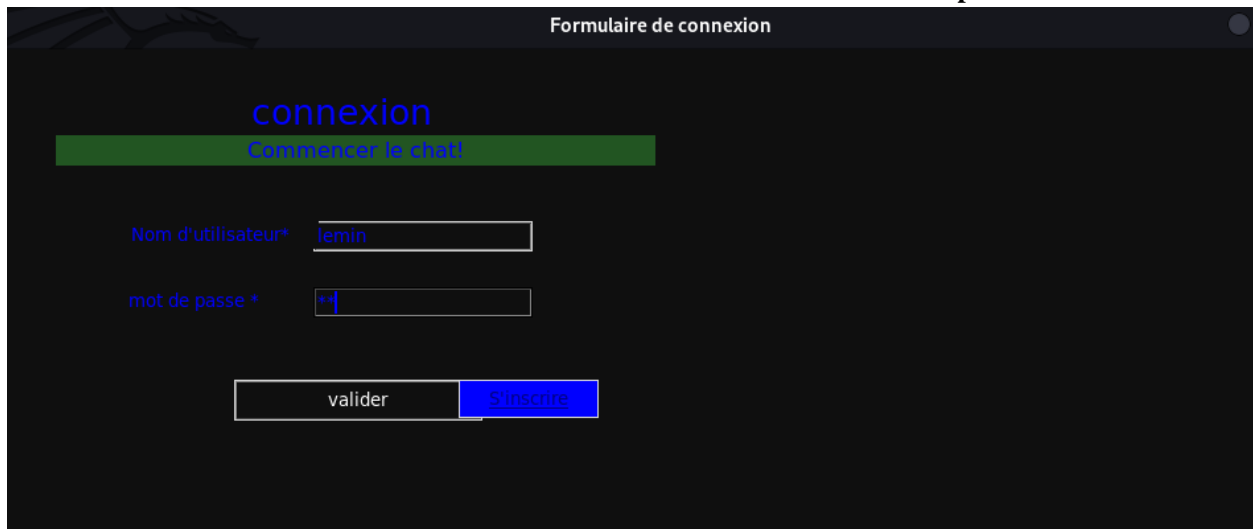
Chapitre II. Simulation et Résultat

II.1 prototype de l'appliation

Dans cette partie du rapport nous allons présenter notre application, et on va voir tous les fonctions possibles, et les operateurs implémentés qui peuvent être effectué par l'utilisateur.

Après l'utilisateur il va authentifier auprès serveur ldap puisque accès a l'interface en faire le demande de certificat(CSR)

II.1.1 authentifier deux client en saisir le nom d'utilisateur et le mot de passe



Formulaire de connexion

connexion

Commencer le chat!

Nom d'utilisateur* lemin

mot de passe * ***

valider S'inscrire

Figure 2:connexion du premiere utilisateur

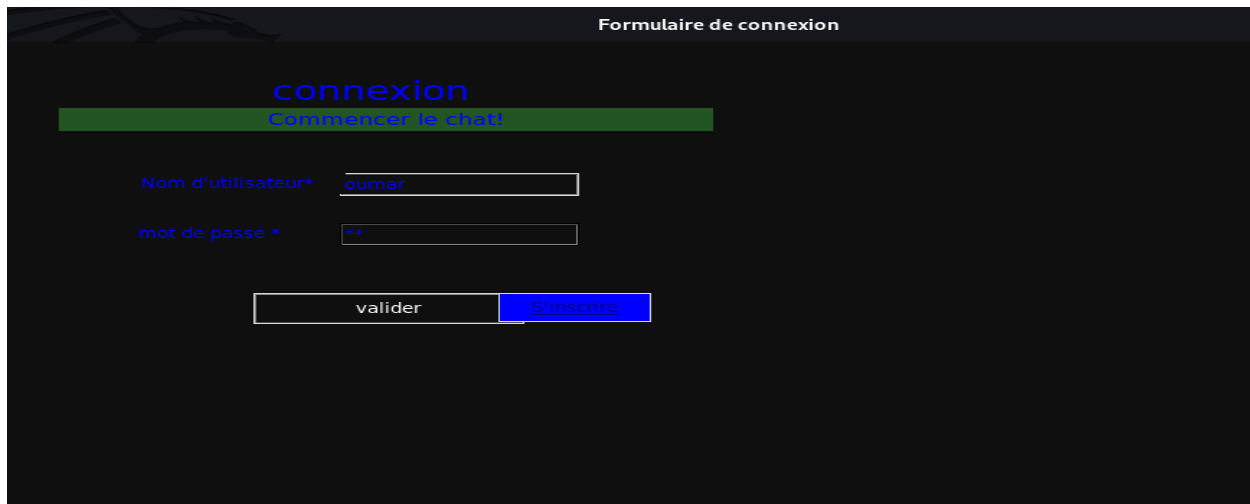


Figure 3:connexion du deuxieme utilisateur

II.1.2 Authentifier les utilisateurs oprés le server openldap

- L'authentification auprès d'un serveur OpenLDAP se fait généralement à l'aide d'un nom d'utilisateur et d'un mot de passe. Le nom d'utilisateur est souvent associé à une entrée dans le serveur de noms LDAP (LN) et le mot de passe est utilisé pour vérifier l'autorisation d'accès. Une fois authentifié, l'utilisateur peut accéder aux informations stockées dans le serveur OpenLDAP.
- On utilise des fichiers de configuration ldap pour la création de schema(basedn.ldif) et on affiche la configuration en utilisant slapcat

il faut des requis pour installer le serveur openldap

```
systemctl enable --now rpcbind
```

```
echo 1 > /proc/sys/net/ipv4/ip_forward
```

```
apt install slapd slap-utils
```

vous allez voir ci-dessous la figure qui démontre le fichier de configuration

```

# cat basedn.ldif
#dn: ou=people,dc=tekup,dc=tn
#objectClass: organization
#ou: people

#dn: ou=groups,dc=tekup,dc=tn
#objectClass: organization
#ou: groups

# Création de l'objet "utilisateur"
  dn: ou=people,dc=tek-up,dc=de
  objectClass: organizationalUnit
  ou: people

  # Création de l'objet "groups"
  dn: ou=groups,dc=tek-up,dc=de
  objectClass: organizationalUnit
  ou: groups

  # Création de l'objet "group"
  dn: ou=group,dc=tek-up,dc=de
  objectClass: organizationalUnit
  ou: group

```

Figure 4:basedn.ldif

On génère un hash pour le password de l'user qu'on va réutiliser jsute après

slappasswd

On crée un fichier pour la création de l'user (avec ses paramètres)

vim ldapuser1.ldif

dn: uid=oumar,ou=people,dc=tek-up,dc=de

objectClass: inetOrgPerson

objectClass: posixAccount

objectClass: shadowAccount

cn: oumar

```
sn:tek-up
```

```
userPassword: {SSHA}XXXXXXXXXXXXXXXX
```

```
loginShell: /bin/bash
```

```
homeDirectory: /home/oumar
```

```
uidNumber: 3000
```

```
gidNumber: 3000
```

```
# On insère le/les users dans le LDAP
```

```
ldapadd -x -D cn=admin,dc=tek-up,dc=de -W -f ldapusers.ldif
```

```
# On crée un fichier pour la création du groupe pour l'user (avec ses paramètres)
```

```
nano ldapgroups.ldif
```

```
dn: uid=oumar,ou=people,dc=tek-up,dc=de
```

```
objectClass: inetOrgPerson
```

```
cn: oumar
```

```
gidNumber: 3000
```

```
memberUid: oumar
```

```
# On insère le/les groupes dans le LDAP
```

```
ldapadd -x -D cn=admin,dc=tek-up,dc=de -W -f ldapgroups.ldif
```

```
--meme demarche pour l'autre utilisateur
```

```
vim ldapuser2.ldif
```

```
dn: uid=lemin,ou=people,dc=tek-up,dc=de
```

```
objectClass: inetOrgPerson
```

```
objectClass: posixAccount
```

```
objectClass: shadowAccount
```

```
cn: lemin
```

```
sn:tek-up
```

```
userPassword: {SSHA}XXXXXXXXXXXXXXXX
```

```
loginShell: /bin/bash
```

```
homeDirectory: /home/lemin
```

```
uidNumber: 3000
```

```
gidNumber: 3000
```

```
# On insère le/les users dans le LDAP
```

```
ldapadd -x -D cn=admin,dc=tek-up,dc=de -W -f ldapusers.ldif
```

```
# On crée un fichier pour la création du groupe pour l'user (avec ses paramètres)
```

```
nano ldapgroups.ldif
```

```
dn: uid=le min,ou=people,dc=tek-up,dc=de
```

```
objectClass: inetOrgPerson
```

```
cn: lemin
```

```
gidNumber: 3000
```

```
memberUid: lemin
```

```
# On insère le/les groupes dans le LDAP
```

```
ldapadd -x -D cn=admin,dc=tek-up,dc=de -W -f ldapgroups.ldif
```

```
L# slapcat
dn: dc=tek-up,dc=de
objectClass: top
objectClass: dcObject
objectClass: organization
o: tek-up
dc: tek-up
structuralObjectClass: organization
entryUUID: d595155e-296f-103d-95a8-7999d9fc40f9
creatorsName: cn=admin,dc=tek-up,dc=de
createTimestamp: 20230115222935Z
entryCSN: 20230115222935.577523Z#000000#000#000000
modifiersName: cn=admin,dc=tek-up,dc=de
modifyTimestamp: 20230115222935Z

Actions      User name
dn: ou=people,dc=tek-up,dc=de
objectClass: organizationalUnit
ou: people
structuralObjectClass: organizationalUnit
entryUUID: 607f51a8-2974-103d-9004-5175c27ce209
creatorsName: cn=admin,dc=tek-up,dc=de
createTimestamp: 20230115230206Z
entryCSN: 20230115230206.625154Z#000000#000#000000
modifiersName: cn=admin,dc=tek-up,dc=de
modifyTimestamp: 20230115230206Z
```

```
dn: uid=oumar,ou=people,dc=tek-up,dc=de
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: shadowAccount
cn: oumar
sn: tek-up
userPassword:: e1NTSEF9TEprZDdGUkRxUl doSXJabjZ3MDh0Y3VaQWpNYkpmcFM=
loginShell: /bin/bash
homeDirectory: /home/oumar
uidNumber: 3000
gidNumber: 3000
structuralObjectClass: inetOrgPerson
uid: oumar
entryUUID: 85de96d2-2976-103d-9006-5175c27ce209
creatorsName: cn=admin,dc=tek-up,dc=de
createTimestamp: 20230115231728Z
entryCSN: 20230115231728.318703Z#000000#000#000000
modifiersName: cn=admin,dc=tek-up,dc=de
modifyTimestamp: 20230115231728Z

dn: ou=group,dc=tek-up,dc=de
objectClass: organizationalUnit
ou: group
structuralObjectClass: organizationalUnit
entryUUID: e01a25be-2979-103d-900a-5175c27ce209
creatorsName: cn=admin,dc=tek-up,dc=de
createTimestamp: 20230115234128Z
entryCSN: 20230115234128.104157Z#000000#000#000000
```

```

LDAP Account Manager - 8.2      admin
dn: uid=lemin,ou=people,dc=tek-up,dc=de
objectClass: inetOrgPerson
objectClass: posixAccount
objectClass: shadowAccount
cn: lemin
sn: tek-up
userPassword:: e1NTSEF9TEprZDdGUkRxUldoSXJabjZ3MDh0Y3VaQWpNYkpmcFM=
loginShell: /bin/bash
homeDirectory: /home/lemin
uidNumber: 1000
gidNumber: 1000
structuralObjectClass: inetOrgPerson
uid: lemin
entryUUID: 4f28c2b2-297a-103d-900b-5175c27ce209
creatorsName: cn=admin,dc=tek-up,dc=de
createTimestamp: 20230115234434Z
entryCSN: 20230115234434.517022Z#000000#000#000000
modifiersName: cn=admin,dc=tek-up,dc=de
modifyTimestamp: 20230115234434Z

```

LDAP Account Manager - 8.2 admin Accounts Tools Help Logout

Users

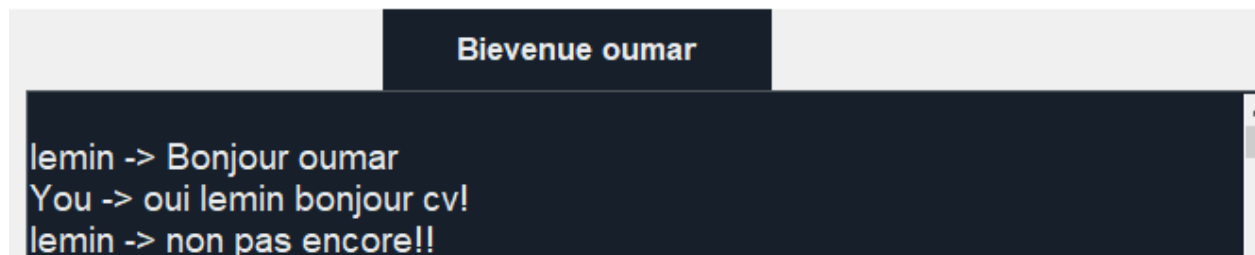
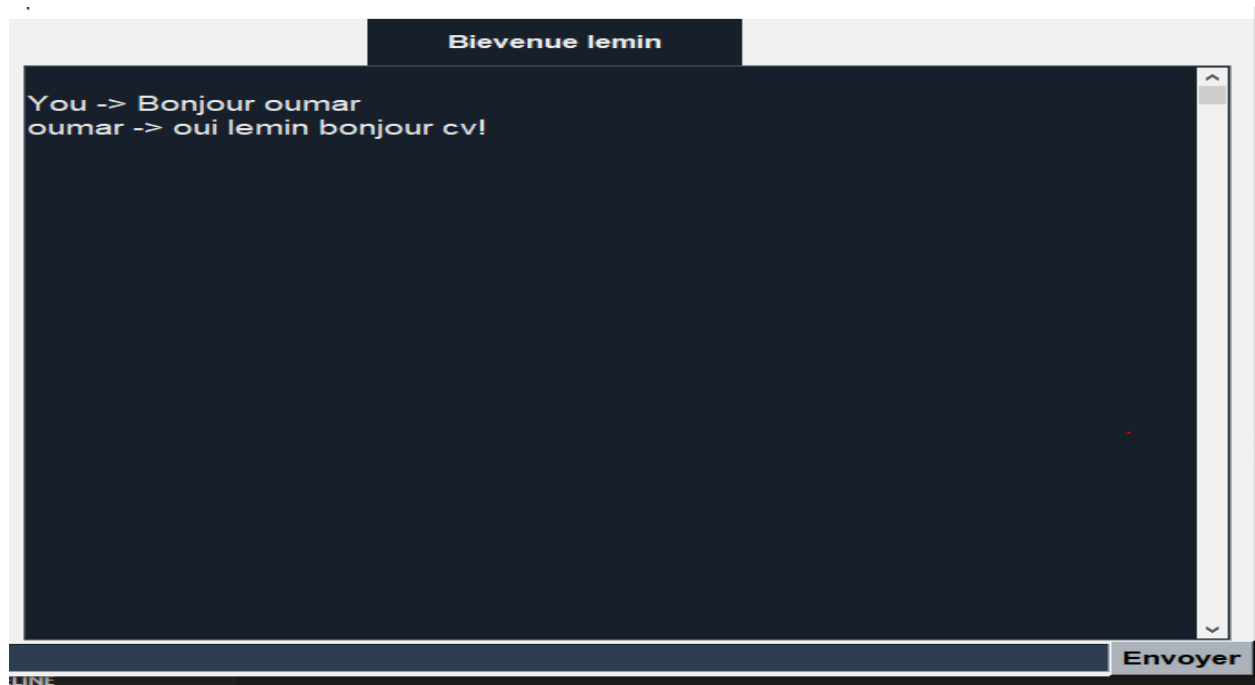
[New user](#) [File upload](#) [Delete selected users](#)

User count: 2

| Actions | User name | First name | Last name | UID number | GID number |
|-----------------------------------|----------------------|----------------------|----------------------|----------------------|----------------------|
| Sort sequence | ▼ ▲ | ▼ ▲ | ▼ ▲ | ▼ ▲ | ▼ ▲ |
| <input type="checkbox"/> Filter ▼ | <input type="text"/> | <input type="text"/> | <input type="text"/> | <input type="text"/> | <input type="text"/> |
| <input type="checkbox"/> | lemin | | tek-up | 1000 | 1000 |
| <input type="checkbox"/> | oumar | | tek-up | 3000 | 3000 |

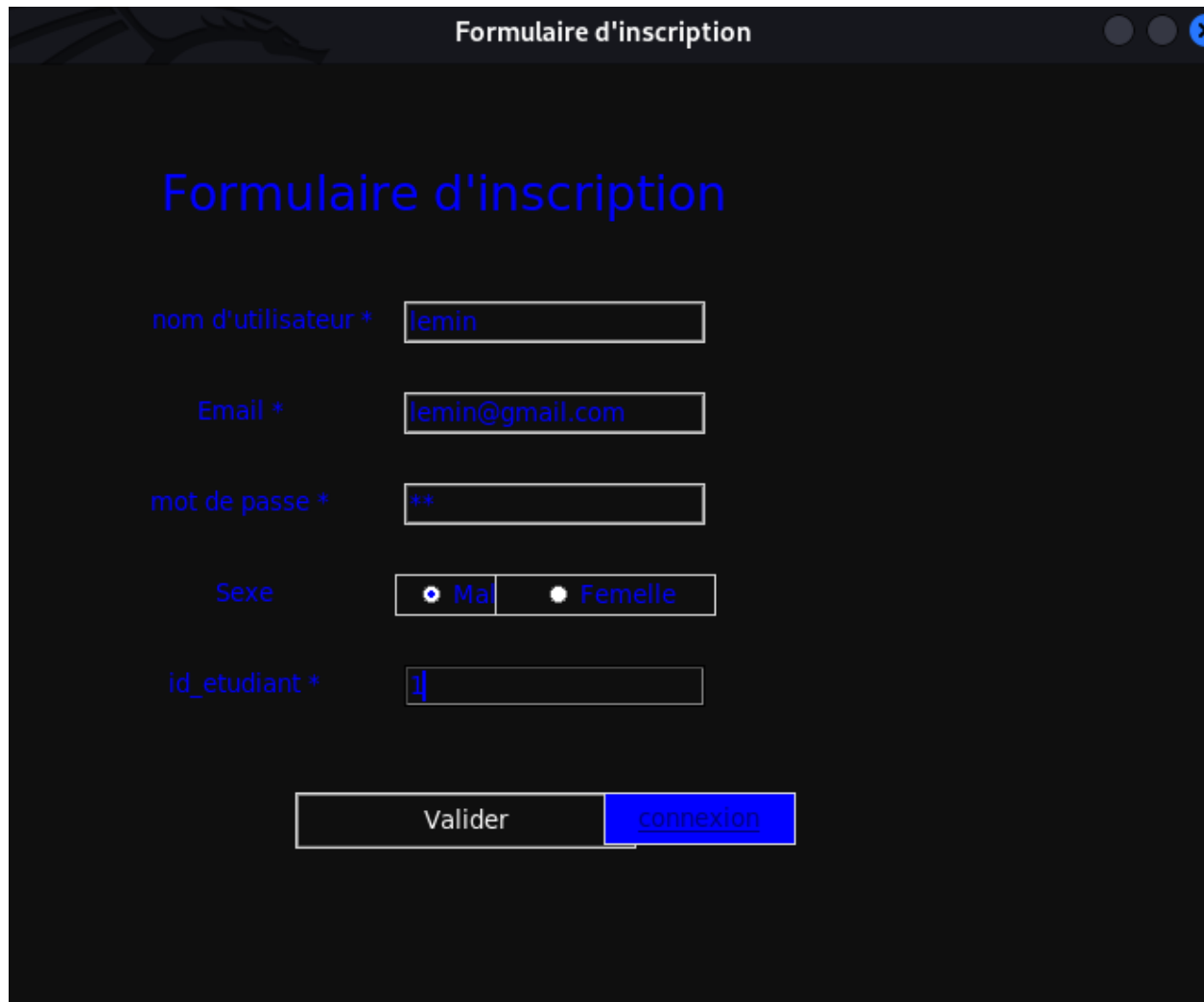
return to your computer, move the mouse pointer outside or press Ctrl+Alt.

maintenant après l'authentification de deux clients depuis serveur ldap voila les communication entre les clients



si on veut faire l'inscription en remplissant les champs et demander l'authentification depuis serveur openldap

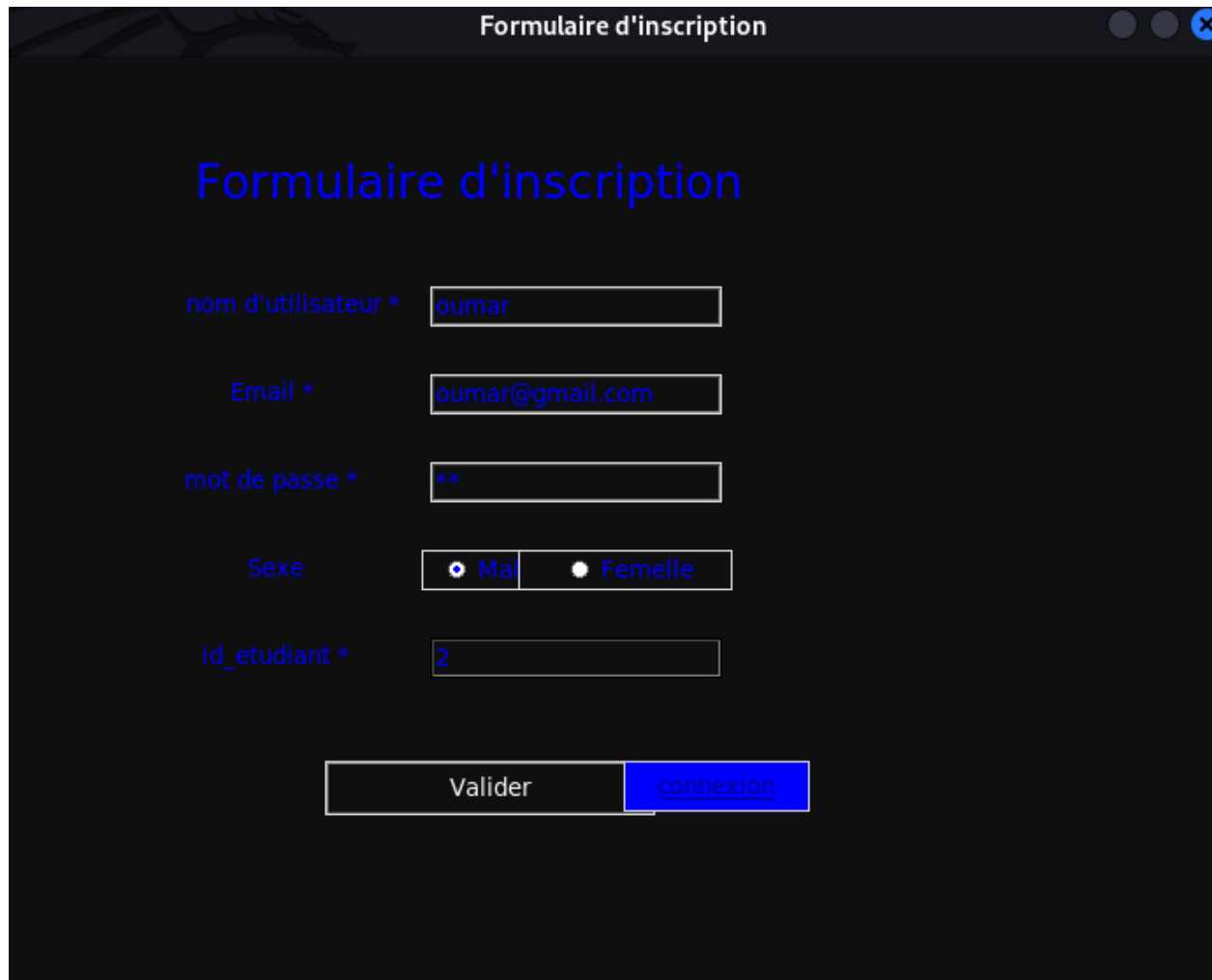
voila les démo sur les figures ci-dessous :



The image shows a web browser window with a dark theme. The title bar at the top says 'Formulaire d'inscription'. The main heading on the page is 'Formulaire d'inscription' in a large, light blue font. Below the heading, there are five form fields, each with a label and an asterisk indicating it is required:

- nom d'utilisateur ***: A text input field containing the text 'lemin'.
- Email ***: A text input field containing the text 'lemin@gmail.com'.
- mot de passe ***: A text input field containing two asterisks '**'.
- Sexe**: A group of two radio buttons. The first is labeled 'Homme' and is selected. The second is labeled 'Femme'.
- id_etudiant ***: A text input field containing the number '1'.

At the bottom of the form, there are two buttons: 'Valider' and 'connexion'.



The image shows a web browser window with a dark theme. The title bar of the browser says "Formulaire d'inscription". The page content has a large heading "Formulaire d'inscription" in blue. Below the heading are five form fields, each with a label and an asterisk indicating it is required:

- nom d'utilisateur *: Input field containing "oumar".
- Email *: Input field containing "oumar@gmail.com".
- mot de passe *: Input field containing "**".
- Sexe: Radio button group with "Male" (selected) and "Female".
- id_etudiant *: Input field containing "2".

At the bottom of the form are two buttons: "Valider" and "connexion". The "connexion" button is highlighted in blue.

II.2 les Fonctionnalités du serveur LDAP avec les clients en gestionnaires de certificat d'autorité en application de chat:

- Authentification centralisée: Le serveur LDAP peut être utilisé pour authentifier les utilisateurs et leurs comptes sur le service de chat. Les informations d'identification et d'autorisation sont stockées dans le référentiel LDAP et peuvent être récupérées par le client de chat pour vérifier les informations d'identification.
- Gestion des certificats d'autorité: Les certificats d'autorité (CA) sont des identifiants numériques qui authentifient les utilisateurs et les serveurs. Les gestionnaires de certificat d'autorité permettent aux utilisateurs de gérer leurs certificats d'autorité, de les valider et de les vérifier. Les certificats d'autorité peuvent être stockés dans le serveur LDAP et peuvent être récupérés par le client de chat pour vérifier les informations d'identification .

Conclusion générale

Nous pouvons conclure que l'utilisation d'OpenLDAP et d'OpenSSL pour créer une application de chat est une solution viable et efficace. OpenLDAP peut être utilisé pour fournir une authentification centralisée et OpenSSL pour fournir une sécurité et un chiffrement des données échangées entre les utilisateurs. Les développeurs peuvent également utiliser OpenLDAP et OpenSSL pour implémenter des fonctionnalités telles que la gestion des noms d'utilisateur et des mots de passe, l'utilisation de certificats pour authentifier les utilisateurs, et la protection des données envoyées et reçues. OpenLDAP et OpenSSL sont donc des outils efficaces et utiles pour développer des applications de chat sécurisées et fiables.