

**Projet système et réseau**  
**3A STI INSA CVL**  
Application coopérative médicale  
Christian TOINARD et Hiep TRAN TUAN

Le but du projet est de développer une solution permettant à différents personnels médicaux de réaliser de façon coopérative un dossier médical.

Les personnels médicaux doivent pouvoir consulter un dossier médical, déposer des documents et éditer ces documents selon certains droits définis par le médecin qui crée le dossier du patient. Cependant, en principe tous les documents doivent rester sur un serveur sécurisé et les personnels y accèdent à distance via un programme client dédié.

Le serveur du dossier médical peut être considéré comme de confiance tandis que les postes clients sont considérés comme curieux mais honnêtes.

Pour garantir la confidentialité du dossier médical, on va mettre en place un client qui permet à un personnel médical d'ouvrir une session à distance sur le serveur. Toutes les données restent sur le serveur. Un personnel peut lancer des commandes comme lister les fichiers présents sur le serveur et lancer un éditeur de texte. Le client ne stocke rien en clair mais peut éventuellement utiliser des fichiers chiffrés.

Le client et le serveur sont écrits en C ou en Python mais ils peuvent lancer d'autres programmes. Le serveur peut par exemple lancer une application LibreOffice avec comme paramètre le fichier compteRenduKine.doc. En quelque sorte, le serveur permet aux clients de s'authentifier, lister les fichiers et lancer des commandes sur ces fichiers.

Le rôle minimal du client est de permettre de saisir un login et un mot de passe et d'exécuter des commandes du type lister un répertoire (ls), se déplacer dans un répertoire (exemple : mv PatientDurand) et exécuter une commande (par exemple : vi compteRenduKiné.txt). Il peut par exemple se comporter comme un simple client telnet qui envoie tout ce qui est saisi au clavier et affiche tout ce qu'il reçoit.

Le rôle minimal du serveur est d'authentifier les personnes et d'offrir un certain nombre de commandes (ls, mv, nomExécutable, ...). Il peut se comporter comme un simple serveur telnet qui authentifie un utilisateur, établit une connexion puis ouvre un shell : le serveur 1) redirige le trafic qui entre sur la connexion vers ce shell afin d'exécuter les commandes et 2) redirige la sortie de ce shell sur la connexion.

Cependant, il faudra que vous choisiez au moins une fonctionnalité additionnelle à réaliser parmi celles proposées. La mise en œuvre de plusieurs fonctionnalités vous permettra d'avoir une évaluation qui va au delà de la moyenne.

Pour certaines fonctionnalités vous pouvez vous reposer sur des mécanismes d'administration système et réseau. Dans ce cas vous proposez une procédure d'administration aisée. On privilégie le caractère fonctionnel et l'aspect réaliste de la solution. On peut imaginer qu'elle soit réellement utilisée par du personnel soignant. A titre d'indication on vous donne une liste de fonctionnalités parmi lesquelles choisir. Vous pourrez en proposer d'autres avec l'accord de vos encadrants.

Fonctionnalités additionnelles :

- Le client peut accepter des flux graphiques du serveur par exemple via RDP ou VNC. Pour cela le client ne doit pas posséder de privilèges particuliers.
- Le serveur permet de créer un dossier ou un fichier avec certains privilèges ([rwx] pour un utilisateur ou [rwx] pour un groupe donné) et pour un ensemble d'utilisateurs ou de groupes quelconques par exemple via des ACLs.
- Le client et le serveur utilisent des connexions chiffrées via des clés qui ne sont pas en clair sur le poste client. On peut par exemple se reposer sur un certificat pour le serveur.
- Le client peut stocker localement des fichiers chiffrés et les ouvrir sans qu'ils apparaissent en clair sur le poste client.
- Le client et le serveur fonctionnent sous Windows.
- Le client et le serveur fonctionnent sur des systèmes hétérogènes par exemple le client est sur Windows et le serveur sur un système Unix.

### **Travail attendu et organisation :**

Les projets se dérouleront par groupe de quatre étudiants. La notation est individuelle et fonction de la contribution à la réflexion, la réalisation et la présentation.

On jugera surtout le côté fonctionnel de la solution via une démonstration durant la soutenance. Une première version de solution doit très vite être disponible et présentée durant le suivi du projet. A ce titre les présences aux séances sont obligatoires et chacun présente ses avancées dans la réalisation.

Les séances de suivi sont l'occasion d'échanges durant lesquels vous justifiez votre approche par rapport au sujet. Vous tenez compte des remarques faites. Enfin vous vous engagez à des points de fourniture (d'une première version, d'une seconde jusqu'à la version finale) sur la base d'un planning à la semaine. Les points de fourniture incluent la livraison d'une spécification, des parties de code à intégrer, etc... (attention à faire une intégration continue offrant les différentes versions).

Les solutions doivent fonctionner sur les machines de l'école si vous utilisez des machines virtuelles il sera de votre ressort de les déployer à l'école.

On vous demande de privilégier les solutions que vous êtes capables de justifier et qui fonctionnent dans des délais courts (attention aux usines à gaz et aux engagements non tenus, fantaisistes ou irréalistes et procédez par étapes et versions successives).