TP: Sécurité Crypto

Il s'agit d'implémenter sur la même machine (utilisant l'adresse localhost) une communication client serveur, chiffrée par le cryptosystème classique de substitution utilisant la fonction affine y=ax+b mod (26). Le déchiffrement est réalisé par la fonction inverse (x=(y/a - b/a) mod 26)

Une lettre de l'alphabet est remplacée (substitution) par une lettre grâce à cette fonction. Chaque caractère est identifié par sa position dans l'alphabet. Les lettres de l'alphabet sont considérées comme chaine de caractères ou bien des éléments d'un tableau. A correspond à 0, Z à 25.

Exemple: y=(3x+5)mod 26
La fonction inverse x=(9y+7)mod 26
le chiffrement de A donne y=3*0 +5==5 correspond à E
le déchiffrement de S donne x=(9*18 + 7)mod 26=162+7=13 correspond à la lettre N

Votre application doit être paramétrable
La clé (a,b) étant introduite par l'utilisateur, il faut vérifier sa validité.
Les cypto systèmes CESAR et ROT13 sont des cas particuliers du chiffre affine. Tester ces systèmes en introduisant (a,b)=(1,3) et (a,b)=(1,13)

Programmer en Java

Au niveau serveur

- Créer socket
  socketServeur = new ServerSocket(port)
- se mettre en attente d'une demande de connexion du côté client: accept()
- Recevoir le message crypté du client (Inputstream ou autre méthode)
- Decrypter le message grâce à la fonction inverse (x=(y/a - b/a) mod 26), dans notre exemple x=(9y+7)mod26
- Envoyer le message décrypté au client Outputstream

Au niveau client

- Instancier le socket ;
  new socket(ip,port)

- Connecter la socket avec le service côté serveur
- Entrée du message par l'utilisateur
- Crypter message grâce à la fonction y=ax+b mod 26 (a et b introduits par l'utilisateur)

- Afficher le message crypté
- Échanger le message crypté avec le serveur
- Fermer le socket lorsque les communications sont terminées.
-

A. Bilami