

# Chiffrer un morceau

## 1 Descriptif

L'objectif de ce défi est d'implémenter une méthode de la classe `AlgorithmRSA` permettant de chiffrer un morceau du message en utilisant la clé publique. On rappelle que la clé publique de RSA est composée de deux parties  $N$  et  $e$ . On rappelle aussi qu'un morceau  $M$  du message est encodé en faisant simplement  $M^e$  modulo  $N$ .

Attention, votre méthode doit renvoyer un mot binaire dont la taille doit impérativement être fixée à celle de la clé RSA. On utilisera la méthode statique `getTailleCle` de la classe `ParametresRSA` pour récupérer cette valeur.

## 2 Protocole

1. Une fois la connexion établie, le serveur commence par envoyer un premier message annonçant le début du défi :

-- Début du défi : Chiffrer un morceau --

Ce message n'attend pas de réponse.

2. Le serveur envoie ensuite une série de triplets  $(M,N,e)$  où  $M$  est un mot binaire de 126 bits et  $N$  et  $e$  des nombres binaires.
3. Pour chaque triplet  $(M,N,e)$ , le serveur doit recevoir en retour  $M$  chiffré avec la clé  $(N,e)$  (sous forme binaire).
4. Après chaque réponse, le serveur enverra un message commençant par "OK" ou "NOK" suivant si la réponse est correcte ou non.
5. A la fin du défi, le serveur enverra un message indiquant "Défi validé" ou "Défi échoué!". Aucune réponse n'est attendue.
6. Le serveur terminera la communication par le message "FIN", votre client devra alors fermer la socket. Aucune réponse n'est attendue.

## 3 Exemple de communication

Voici un exemple (incomplet) d'une communication pour ce défi. Dans cet exemple les "<" et ">" indiquent le sens de transfert de chaque message et ne doivent pas être présents dans la communication.

