Fiche n°2

Chiffrement symétrique en *Java* avec AES

Cet article explique comment utiliser AES pour chiffrer et déchiffrer un message à l'aide d'une clé de 16 caractères.

1 Le programme

Dans un premier temps, il faut générer le mot de passe. Nous partons d'une chaîne de caractères de 16 octets (attention à la représentation) et nous utilisons la classe `SecretKeySpec`. La construction d'un objet utilise le tableau d'octets correspondant au mot de passe et le nom de l'algorithme utilisé (ici AES).

```
String motDePasse = "0123456789012345";
SecretKeySpec specification = new SecretKeySpec(motDePasse.getBytes(), "
   AES");
```

Une fois la clé prête, nous créons une instance de `Cypher` qui correspond au chiffreur à l'aide de la méthode `getInstance`. Nous le configurons à l'aide de la méthode `init` qui prend en paramètre le mode (ici `ENCRYPT_MODE`) et la clé.

```
String message = "Message␣à␣coder";
Cipher chiffreur = Cipher.getInstance("AES");
chiffreur.init(Cipher.ENCRYPT_MODE, specification);
byte[] bytes = chiffreur.doFinal(message.getBytes());
```

Pour le déchiffrement, nous pouvons créer une nouvelle instance de `Cypher` mais configurée cette fois-ci avec la constante `DECRYPT_MODE`.

```
Cipher dechiffreur = Cipher.getInstance("AES");
dechiffreur.init(Cipher.DECRYPT_MODE, specification);
bytes = dechiffreur.doFinal(bytes);
```

2 Exécution

Pour tester le programme, exécutez simplement la commande suivante :

```
java ChiffrementAES 0123456789012345 "Bonjour␣tout␣le␣monde"
```

Le texte `Bonjour tout le monde` sera chiffré puis déchiffré à l'aide de la clé spécifiée.