M1 2015 - 2016

Introduction à la Sécurité **TD2**

Exercice 1 - Familiarisation avec les LFSR

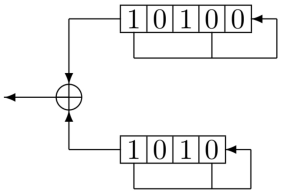
1. Donnez les suites binaires produites par le LFSR de longueur 4 et de polynôme de rétroaction $P(X) = 1 + X + X^2 + X^4$, en fonction des états initiaux du registre. Quelles sont leurs périodes.
2. Mêmes questions avec les polynômes de rétroaction $Q(x) = 1 + X + X^2 + X^3 + X^4$ et $R(X) = 1 + X^3 + X^4$.

Exercice 2 - Périodicité des LFSR

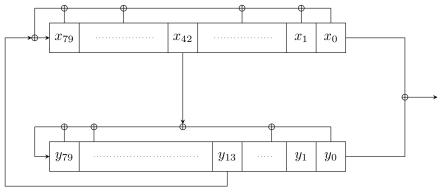
Expliquez pourquoi les suites binaires produites par les LFSR sont périodiques à partir d'un certain rang. Quelle peut être la plus grande période d'une suite produite par un LFSR de longueur L ?

Exercice 3 - Combinaison de LFSR

1. Considérons deux LFSRs dont les sorties sont combinées par un ou-exclusif pour produire une suite binaire.



   La suite produite peut-elle être générée par un seul LFSR ? Si oui, lequel ?
2. Voyez-vous un avantage à combiner deux (ou plus) LFSR ?
3. Reprenez l'exercice avec un **AND** au lieu d'un **XOR**.



Exercice 4 -

Vous travaillez dans une entreprise de sécurité informatique. Votre supérieur hiérarchique, qui a suivi un cours de LFSR, vous amène sa dernière création. Dans ce dispositif génial, on met la clé dans $x_{79}, \ldots, x_0$ et le message à chiffrer dans $y_{79}, \ldots, y_0$. On jette à la poubelle les 1000 premiers bits, puis les 80 bits suivant forme le message chiffré. Le polynôme de rétroaction du premier LFSR est $1 + X + X^2 + X^3 + X^{11} + X^{80}$ et celui du second est $1 + X^2 + X^4 + X^9 + X^{80}$. Mais à ça viennent s'ajouter les bits perturbateurs en provenance de l'autre LFSR...

Que dites-vous à votre chef ?

Exercice 5 - RC4

Le système de chiffrement par flot RC4 a été utilisé dans différents protocoles et notamment le WEP utilisé pour sécuriser les réseaux sans-fils de type Wi-Fi.

Algorithm 1: RC4

Data: Permutation S de $\{0, \ldots, 255\}$ et entier $n > 0$
Result: $(z_1, \ldots, z_n) \in \{0, \ldots, 255\}^n$

$i, j \leftarrow 0$;
for *k from 1 to n* **do**
  $i \leftarrow (i + 1) \bmod 256$;
  $j \leftarrow (j + S[i]) \bmod 256$;
  $S[i] \leftrightarrow S[j]$;
  $z_k \leftarrow S[(S[i] + S[j]) \bmod 256]$;
end
return $z_1, \ldots, z_n$

1. Pour cette question, on considère une variante simplifiée de RC4 où l'opération d'échange $S[i] \leftrightarrow S[j]$ est omise.
   a) Montrer que la suite chiffrante est périodique de période 512.
   b) En déduire un moyen de reconstruire la permutation S à partir de la suite chiffrante.
2. Notons $S_j[i]$ la valeur de la permuttation S en l'octet i après le j-ème tour de boucle de l'algorithme.
   a) Montrer que si $S_0[2] = 0$ et $S_0[1] \neq 2$ alors $z_2 = 1$.
   b) En déduire que la distribution de z_1 n'est pas uniforme lorsque S est initialisée avec une permutation aléatoire.
   c) Supposons que le même texte clair est chiffré pour n destinataires ayant des clés RC4 différentes. Proposer un algorithme qui retrouve le second octet du message clair.