

Securitatea Sistemelor Informatice



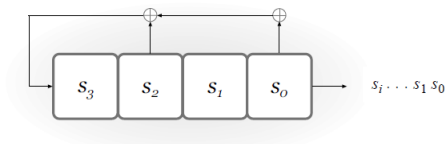
- Curs 4.2 - Exemple sisteme fluide

Adela Georgescu

Facultatea de Matematică și Informatică
Universitatea din București
Anul universitar 2022-2023, semestrul I

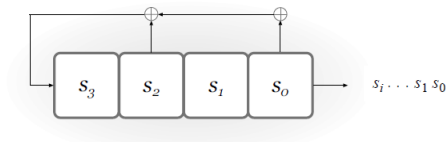
Linear-Feedback Shift Registers (LFSR)

- ▶ sunt foarte eficiente în implementari hardware
- ▶ au proprietăți statistice bune dar totuși sunt predictibile, deci nu sunt PRG-uri sigure din punct de vedere criptografic
- ▶ Mai jos este un exemplu



- ▶ Componente, în general:
 - ▶ n regiștri s_{n-1}, \dots, s_0 - fiecare conține un singur bit
 - ▶ n coeficienți feedback c_{n-1}, \dots, c_0
 - ▶ gradul este n

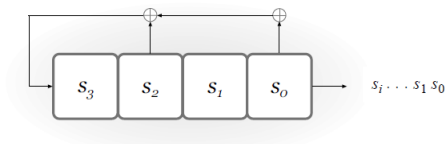
Linear-Feedback Shift Registers (LFSR)



In exemplul de mai sus avem

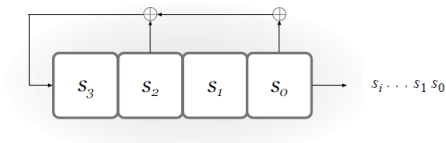
- ▶ $c_0 = c_2 = 1$ si $c_1 = c_3 = 0$
- ▶ fiecare bit de la ieşire este calculat după formula $c_0 s_0 \oplus \dots \oplus c_3 s_3$
- ▶ la fiecare tact de ceas, LFSR scoate la ieşire valoarea din registrul s_0 iar valorile din ceilalţi regiştri sunt deplasate la dreapta cu o poziţie

Linear-Feedback Shift Registers (LFSR)



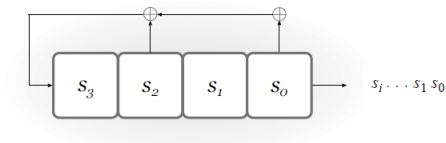
Pentru starea inițială (0,0,1,1), biții de la ieșire vor fi ...

Linear-Feedback Shift Registers (LFSR)



Pentru starea inițială (0,0,1,1), biții de la ieșire vor fi ...
(0,0,1,1)

Linear-Feedback Shift Registers (LFSR)



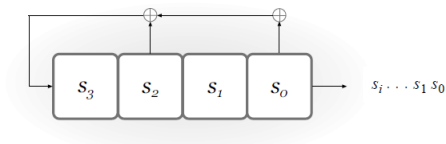
Pentru starea inițială (0,0,1,1), biții de la ieșire vor fi ...

(0,0,1,1)

(1,0,0,1)

...

Linear-Feedback Shift Registers (LFSR)



Pentru starea inițială (0,0,1,1), biții de la ieșire vor fi ...

(0,0,1,1)

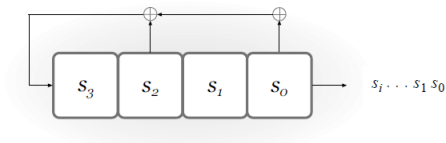
(1,0,0,1)

...

În general

- ▶ starea LFSR constă din n biți (conținutul regiștrilor la un moment dat)

Linear-Feedback Shift Registers (LFSR)



Pentru starea inițială (0,0,1,1), biții de la ieșire vor fi ...

(0,0,1,1)

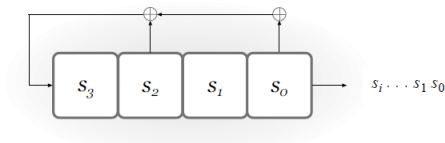
(1,0,0,1)

...

În general

- ▶ starea LFSR constă din n biți (conținutul regiștrilor la un moment dat)
- ▶ există cel mult 2^n stări posibile până când output-ul LFSR-ului se repetă

Linear-Feedback Shift Registers (LFSR)



Pentru starea inițială (0,0,1,1), biții de la ieșire vor fi ...

(0,0,1,1)

(1,0,0,1)

...

În general

- ▶ starea LFSR constă din n biți (conținutul regiștrilor la un moment dat)
- ▶ există cel mult 2^n stări posibile până când output-ul LFSR-ului se repetă
- ▶ cunoscând cel mult $2n$ biți de la ieșire, un atacator poate afla starea inițială și coeficienții de feedback

RC4

Informații generale

RC4 este:

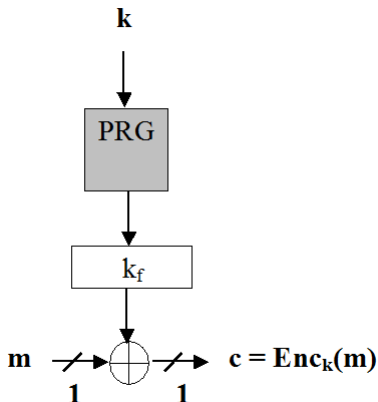
- ▶ introdus de R. Rivest la MIT (1987);
- ▶ înregistrat ca marca a RSA Data Security;
- ▶ păstrat secret până în 1994 când a devenit public;
- ▶ utilizat în WEP, SSL/TLS.

Descriere

- ▶ RC4 este un sistem de criptare fluid pe octeți:

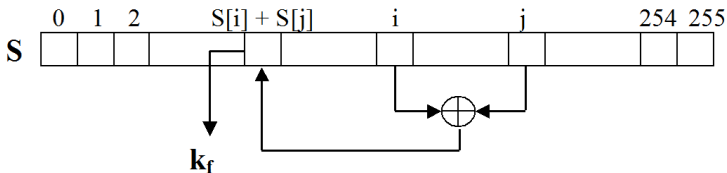
$$m \in \{0, 1\}^8, c \in \{0, 1\}^8$$

- ▶ Ramâne de definit PRG...



Descriere

- ▶ 2 faze:
 - ▶ **inițializare**: determină starea internă, fără să producă chei fluide;
 - ▶ **generare de chei fluide**: modifică starea internă și generează un octet (*cheia fluidă*) care se XOR-ează cu m pentru a obține c ;
- ▶ Starea internă:
 - ▶ un tablou S de 256 octeți: $S[0], \dots, S[255]$;
 - ▶ 2 indici i și j ;
- ▶ Toate operațiile se efectuează pe octeți (i.e. (mod 256)).



Descriere

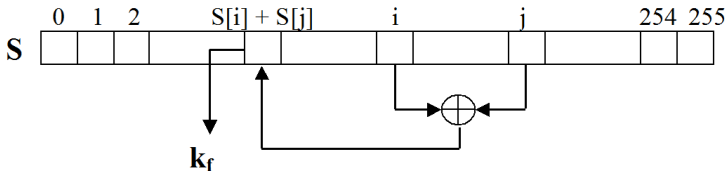
Faza 1. Inițializare

- ▶ $n = \text{numărul octeților din cheie, } 1 \leq n \leq 256$
- ▶
 - $j \leftarrow 0$
 - for** $i = 0$ **to** 255 **do**
 - $S[i] \leftarrow i$
 - end for**
 - for** $i = 0$ **to** 255 **do**
 - $j \leftarrow j + S[i] + k[i \pmod n]$
 - swap ($S[i], S[j]$)
 - end for**
 - $i \leftarrow 0$
 - $j \leftarrow 0$

Descriere

Faza 2. Generarea cheii fluide

- cheia se obține octet cu octet
- - $i \leftarrow i + 1$
 - $j \leftarrow j + S[i]$
 - swap ($S[i], S[j]$)
 - return** $S[S[i] + S[j]]$



Descriere

Detalii de implementare:

- ▶ $5 \leq n \leq 16 \Rightarrow 40 \leq |k| \leq 256$;
- ▶ memorie: 256 octeți (pentru S) și câteva variabile *byte*;
- ▶ operații simple, rapid de executat.

- ▶ primii octeți generați drept cheie fluidă sunt total ne-aleatori și oferă informații despre cheie (Fluhrer, Mantin and Shamir 2001)
- ▶ RC4 pe 104 biți (utilizat pentru WEP pe 128 biți) a fost spart în aprox. 1 min (algoritm al lui Tews, Weinmann, Pychkin 2001, bazat pe ideea lui Klein 2005)
- ▶ un atac recent arată că pot fi determinați primii aprox. 200 octeți din textul clar criptat cu RC4 în TLS cunoscând $[2^{28} - 2^{32}]$ criptări independente (Royal Holloway, 2013)

Vulnerabilitati LFSR

- ▶ LFSR-urile sunt liniare iar liniaritatea induce vulnerabilități (sistemele liniare de ecuații permit aflarea informațiilor sensibile)

Vulnerabilitati LFSR

- ▶ LFSR-urile sunt liniare iar liniaritatea induce vulnerabilități (sistemele liniare de ecuații permit aflarea informațiilor sensibile)
- ▶ Însă combinațiile de mai multe LFSR-uri pot produce sisteme de criptare sigure

Trivium

- ▶ Trivium a fost propus în 2008, este simplu și compact hardware, constă din 3 FSR-uri (feedback shift registers) neliniare de grad 93, 84 respectiv 111
- ▶ Regiștri sunt cuplați: la fiecare tact, cel mai din stânga registru va conține o valoare calculată ca funcție aplicată unui registru din același FSR dar și unor regiștri dintr-un alt FSR
- ▶ cel mai bun atac cunoscut pentru Trivium este cel prin forță brută

