



Criptografie și Securitate

- Prelegerea 7 - Sisteme fluide

Adela Georgescu, Ruxandra F. Olimid

Facultatea de Matematică și Informatică
Universitatea din București

Cuprins

1. Definiție
2. Securitate
3. Moduri de utilizare
4. Exemple

Sisteme fluide

- ▶ Am văzut că securitatea perfectă există, dar nu este practic accesibilă - **OTP**;

Sisteme fluide

- ▶ Am văzut că securitatea perfectă există, dar nu este practic accesibilă - **OTP**;
- ▶ Facem un compromis de securitate, dar obținem o soluție utilizabilă în practică - **sisteme de criptare fluide**;

Sisteme fluide

- ▶ Am văzut că securitatea perfectă există, dar nu este practic accesibilă - **OTP**;
- ▶ Facem un compromis de securitate, dar obținem o soluție utilizabilă în practică - **sisteme de criptare fluide**;
- ▶ Sistemele fluide sunt similare OTP, cu diferența că secvența **perfect aleatoare** de biți cu care se XOR-ează mesajul clar este înlocuită de o secvență **pseudoaleatoare** de biți.

Pseudoaleatorismul

- Un șir **pseudoaleator** "arată" similar unui șir uniform aleator din punct de vedere al oricărui algoritm **polinomial**;

Pseudoaleatorismul

- ▶ Un șir **pseudoaleator** "arată" similar unui șir uniform aleator din punct de vedere al oricărui algoritm **polinomial**;
- ▶ Altfel spus: un algoritm **polinomial** nu poate face diferența între o secvență **perfect aleatoare** și una **pseudoaleatoare** (decât cu probabilitate neglijabilă);

Pseudoaleatorismul

- ▶ Un șir **pseudoaleator** "arată" similar unui șir uniform aleator din punct de vedere al oricărui algoritm **polinomial**;
- ▶ Altfel spus: un algoritm **polinomial** nu poate face diferența între o secvență **perfect aleatoare** și una **pseudoaleatoare** (decât cu probabilitate neglijabilă);
- ▶ Sau: o distribuție a secvențelor de lungime l este **pseudoaleatoare** dacă este **nedistinctibilă** de distribuția uniformă a secvențelor de lungime l ;

Pseudoaleatorismul

- ▶ Un șir **pseudoaleator** "arată" similar unui șir uniform aleator din punct de vedere al oricărui algoritm **polinomial**;
- ▶ Altfel spus: un algoritm **polinomial** nu poate face diferența între o secvență **perfect aleatoare** și una **pseudoaleatoare** (decât cu probabilitate neglijabilă);
- ▶ Sau: o distribuție a secvențelor de lungime l este **pseudoaleatoare** dacă este **nedistinctibilă** de distribuția uniformă a secvențelor de lungime l ;
- ▶ Mai exact: nici un algoritm polinomial nu poate spune dacă o secvență de lungime l este eșantionarea unei distribuții pseudoaleatoare sau este o secvență total aleatoare de lungime l .

Pseudoaleatorismul

- ▶ În analogie cu ce știm deja:
 - ▶ **pseudoaleatorismul** este o relaxare a **aleatorismului perfect**

Pseudoaleatorismul

- ▶ În analogie cu ce știm deja:
 - ▶ **pseudoaleatorismul** este o relaxare a **aleatorismului perfect**
asa cum
 - ▶ **securitatea computațională** este o relaxare a **securității perfecte**

Sisteme fluide

- ▶ Revenind la criptarea fluidă...

Sisteme fluide

- ▶ Revenind la criptarea fluidă...
- ▶ ... aceasta presupune 2 faze:
 - ▶ **Faza 1:** se generează o secvență pseudoaleatoare de biți, folosind un generator de numere pseudoaleatoare (PRG)

Sisteme fluide

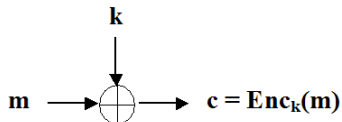
- ▶ Revenind la criptarea fluidă...
- ▶ ... aceasta presupune 2 faze:
 - ▶ **Faza 1:** se generează o secvență pseudoaleatoare de biți, folosind un **generator de numere pseudoaleatoare (PRG)**
 - ▶ **Faza 2:** secvența obținută se XOR-ează cu mesajul clar

Sisteme fluide

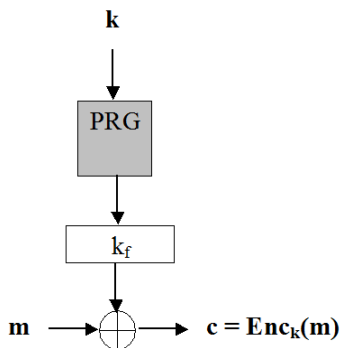
- ▶ Revenind la criptarea fluidă...
- ▶ ... aceasta presupune 2 faze:
 - ▶ **Faza 1:** se generează o secvență pseudoaleatoare de biți, folosind un **generator de numere pseudoaleatoare (PRG)**
 - ▶ **Faza 2:** secvența obținută se XOR-ează cu mesajul clar
- ▶ **Atenție!** De multe ori când ne referim la un sistem de criptare fluid considerăm doar Faza 1

Sisteme fluide

OTP (One Time Pad)



Sisteme fluide



PRG

- ▶ Ramâne să definim noțiunea de **generator de numere aleatoare** sau **PRG** (*PseudoRandom Generator*);

PRG

- ▶ Ramâne să definim noțiunea de **generator de numere aleatoare** sau **PRG** (*PseudoRandom Generator*);
- ▶ Acesta este un algoritm **determinist** care primește o "sămânță" relativ scurtă s (*seed*) și generează o secvență *pseudoaleatoare* de biți;

PRG

- ▶ Ramâne să definim noțiunea de **generator de numere aleatoare** sau **PRG** (*PseudoRandom Generator*);
- ▶ Acesta este un algoritm **determinist** care primește o "sămânță" relativ scurtă s (*seed*) și generează o secvență *pseudoaleatoare* de biți;
- ▶ Notăm $|s| = n$, $|PRG(s)| = l(n)$

PRG

- ▶ Ramâne să definim noțiunea de **generator de numere aleatoare** sau **PRG** (*PseudoRandom Generator*);
- ▶ Acesta este un algoritm **determinist** care primește o "sămânță" relativ scurtă s (*seed*) și generează o secvență *pseudoaleatoare* de biți;
- ▶ Notăm $|s| = n$, $|PRG(s)| = l(n)$
- ▶ PRG prezintă interes dacă:

$$l(n) \geq n$$

(altfel NU "generează aleatorism")

Definitie

Fie $l(\cdot)$ un polinom și G un algoritm polinomial determinist a.î.

$\forall s \in \{0, 1\}^n$, G generează o secvență de lungime $l(n)$.

G se numește **generator de numere pseudoaleatoare (PRG)** dacă se satisfac 2 proprietăți:

1. **Expansiune**: $\forall n, l(n) \geq n$
2. **Pseudoaleatorism**: \forall algoritm PPT \mathcal{D} , \exists o funcție neglijabilă negl a.î.:

$$|\Pr[D(r) = 1] - \Pr[D(G(s)) = 1]| \leq \text{negl}(n)$$

unde $r \leftarrow^R \{0, 1\}^{l(n)}$, $s \leftarrow^R \{0, 1\}^n$

$l(n)$ se numește **factorul de expansiune** al lui G

Notății

- ▶ $\mathcal{D} = \textit{Distinguisher}$
- ▶ PPT = Probabilistic Polynomial Time
- ▶ $x \xleftarrow{R} X = x$ este ales uniform aleator din X
- ▶ $\textit{negl}(n)$ = o funcție neglijabilă în (parametrul de securitate) n

Notății

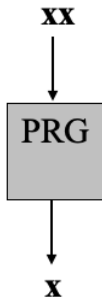
- ▶ $\mathcal{D} = \textit{Distinguisher}$
- ▶ PPT = Probabilistic Polynomial Time
- ▶ $x \leftarrow^R X = x$ este ales uniform aleator din X
- ▶ $\text{negl}(n)$ = o funcție neglijabilă în (parametrul de securitate) n

În plus:

- ▶ Vom nota \mathcal{A} un adversar (Oscar / Eve), care (în general) are putere polinomială de calcul

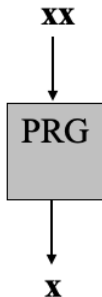
DA sau NU

Este acesta un PRG?



DA sau NU

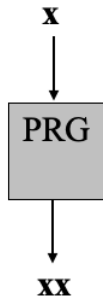
Este acesta un PRG?



NU - secvența de ieșire este mai scurtă decât cea de intrare

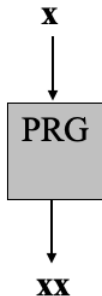
DA sau NU

Este acesta un PRG?



DA sau NU

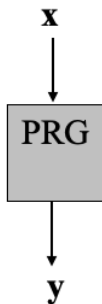
Este acesta un PRG?



NU - secvența de ieșire este clar non-pseudoaleatoare

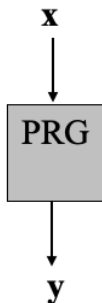
DA sau NU

Este acesta un PRG? $|y| \geq |x|$, y pseudoaleator, fixat pentru $\forall x$



DA sau NU

Este acesta un PRG? $|y| \geq |x|$, y pseudoaleator, fixat pentru $\forall x$



NU - secvența de ieșire este aceeași indiferent de secvența de intrare

Sisteme fluide

Definitie

Un sistem de criptare (Enc, Dec) definit peste $(\mathcal{K}, \mathcal{M}, \mathcal{C})$ se numește *sistem de criptare fluid* dacă:

1. $Enc : \mathcal{K} \times \mathcal{M} \rightarrow \mathcal{C}$

$$c = Enc_k(m) = G(k) \oplus m$$

2. $Dec : \mathcal{K} \times \mathcal{C} \rightarrow \mathcal{M}$

$$m = Dec_k(c) = G(k) \oplus c$$

unde G este un generator de numere pseudoaleatoare cu factorul de expansiune l , $k \in \{0, 1\}^n$, $m \in \{0, 1\}^{l(n)}$

Securitate - interceptare unică

Teorema

Dacă G este PRG, atunci sistemul fluid definit anterior este un sistem de criptare simetric de lungime fixă computațional sigur pentru un atacator pasiv care poate intercepta un mesaj.

Demonstrație intuitivă

- ▶ OTP este perfect sigur;

Demonstrație intuitivă

- ▶ OTP este perfect sigur;
- ▶ Criptarea fluidă se obține din OTP prin înlocuirea *pad* cu $G(k)$;

Demonstrație intuitivă

- ▶ OTP este perfect sigur;
- ▶ Criptarea fluidă se obține din OTP prin înlocuirea *pad* cu $G(k)$;
- ▶ Dacă G este PRG, atunci *pad* și $G(k)$ sunt indistingtibile pentru orice \mathcal{A} adversar PPT;

Demonstrație intuitivă

- ▶ OTP este perfect sigur;
- ▶ Criptarea fluidă se obține din OTP prin înlocuirea pad cu $G(k)$;
- ▶ Dacă G este PRG, atunci pad și $G(k)$ sunt indistingtibile pentru orice \mathcal{A} adversar PPT;
- ▶ În concluzie, OTP și sistemul de criptare fluid sunt indistingtibile pentru \mathcal{A} .

Securitate - interceptare multiplă

- ▶ Un sistem de criptare fluid în varianta prezentată este **determinist**: *unui text clar îi corespunde întotdeauna același mesaj criptat*;
- ▶ În consecință, utilizarea unui sistem fluid în forma prezentată pentru criptarea mai multor mesaje (cu aceeași cheie) este **nesigură**;
- ▶ Un sistem de criptare fluid se folosește în practică în 2 moduri: **sincronizat** și **nesincronizat**.

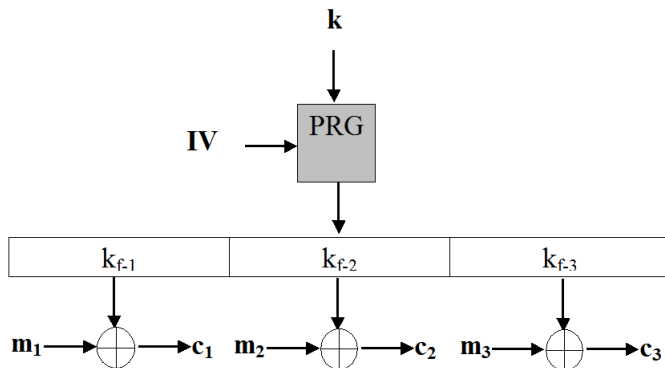
Moduri de utilizare

- ▶ **modul sincronizat**: partenerii de comunicație folosesc pentru criptarea mesajelor *părți succesive* ale secvenței pseudoaleatoare generate;
- ▶ **modul nesincronizat**: partenerii de comunicație folosesc pentru criptarea mesajelor secvențe pseudoaleatoare *diferite*.

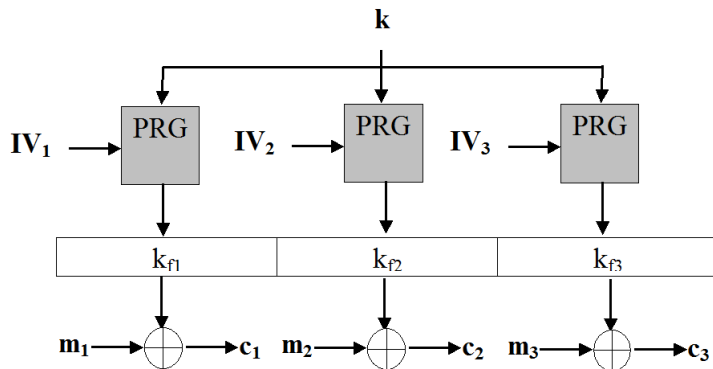
Atenție!

PRG va necesita 2 intrări: cheia k și un vector de inițializare IV .

Modul sincronizat



Modul nesincronizat



Moduri de utilizare

Modul sincronizat

- ▶ mesajele sunt criptate în mod **succesiv** (participanții trebuie să știe care părți au fost deja folosite)
- ▶ necesită **păstrarea** stării
- ▶ mesajele succesive pot fi percepute ca un *singur mesaj clar lung*, obținut prin concatenarea mesajelor succesive
- ▶ se pretează unei singure sesiuni de comunicații

Modul nesincronizat

- ▶ mesajele sunt criptate în mod **independent**
- ▶ NU necesită **păstrarea** stării
- ▶ valorile IV_1, IV_2, \dots sunt alese uniform aleator pentru fiecare mesaj transmis
- ▶ valorile IV_1, IV_2, \dots (dar și IV în modul sincronizat) fac parte din mesajul criptat (sunt necesare pentru decriptare)

Proprietăți necesare ale PRG în modul nesincronizat

Fie $G(s, IV)$ un PRG cu 2 intrări:

- ▶ $s = \text{seed}$
- ▶ $IV = \text{Initialization Vector}$

PRG trebuie să se satisfacă (cel puțin):

1. $G(s, IV)$ este o secvență pseudoaleatoare chiar dacă IV este public (i.e. securitatea lui G constă în securitatea lui s);
2. dacă IV_1 și IV_2 sunt valori uniform aleatoare, atunci $G(s, IV_1)$ și $G(s, IV_2)$ sunt indistingtibile.

Exemple

- ▶ **RC4** (Ron's Cipher 4):
 - ▶ definit de R.Rivest, în 1987
 - ▶ utilizat în WEP
 - ▶ inițial secret !

Exemple

- ▶ **RC4** (Ron's Cipher 4):
 - ▶ definit de R.Rivest, în 1987
 - ▶ utilizat în WEP
 - ▶ inițial secret !
- ▶ **WEP** (Wired Equivalent Privacy):
 - ▶ standard IEEE 802.11, 1999 (rețele fără fir)
 - ▶ înlocuit în 2003 de WPA (Wi-Fi Protected Access), 2004 WPA2 - IEEE 802.11i

Exemple

- ▶ A5/1:
 - ▶ definit în 1987 pentru Europa și SUA
 - ▶ A5/2 definit în 1989 ca o variantă mai slabă pentru alte zone geografice
 - ▶ utilizat în rețelele de telefonie mobilă GSM
 - ▶ inițial secret !

Exemple

- ▶ **A5/1:**
 - ▶ definit în 1987 pentru Europa și SUA
 - ▶ A5/2 definit în 1989 ca o variantă mai slabă pentru alte zone geografice
 - ▶ utilizat în rețelele de telefonie mobilă GSM
 - ▶ inițial secret !
- ▶ **SEAL** (Software-Optimized Encryption Algorithm)
 - ▶ definit de D.Coppersmith și P.Rogaway, în 1993
 - ▶ prezintă o implementare foarte eficientă pe procesoarele pe 32 de biți
 - ▶ versiunea curentă (SEAL 3.0) este patentată IBM

Important de reținut!

- ▶ Noțiunile de pseudoaleatorism, PRG
- ▶ OTP vs. Sisteme fluide
- ▶ Transpunerea sistemelor fluide în practică