

# Securitatea Sistemelor Informatice



## - Curs 3.2 - Aleatorism

Adela Georgescu

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

## Aleatorism

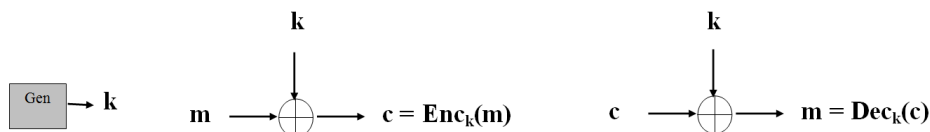
- ▶ Am definit ce înseamnă pentru o schemă de criptare să fie sigură (noțiunea de indistinctibilitate, curs 3), vrem să vedem o construcție
- ▶ În cadrul securității computaționale putem avea
  - ▶ chei mai scurte pentru mesaje mai lungi
  - ▶ re folosirea cheilor pentru mai multe mesaje

Securitatea Sistemelor Informatice

2/16

## Aleatorism

- ▶ Încercăm criptare în stil OTP: cheia va masca mesajul dar
  - ▶ masca nu va fi doar cheia ci  $\text{masca} = f(\text{cheie})$  unde  $f$  este o funcție de extindere a cheii
  - ▶ pentru securitate perfectă masca trebuie să fie perfect aleatoare
  - ▶ pentru securitate computațională, este suficient ca masca să para aleatoare pentru un adversar PPT chiar dacă nu este
- ▶ Vom avea nevoie întâi să definim noțiunea de *generatoare de numere pseudoaleatoare* ca element important de construcție pentru schemele de criptare simetrice



Securitatea Sistemelor Informatice

3/16

## 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  $l$ .

Securitatea Sistemelor Informatice

4/16

## Pseudoaleatorismul

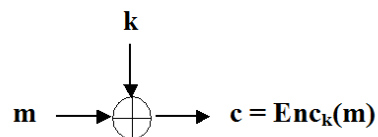
- ▶ În analogie cu ce știm deja:
  - ▶ **pseudoaleatorismul** este o relaxare a **aleatorismului perfect**  
*asa cum*
  - ▶ **securitatea computațională** este o relaxare a **securității perfecte**

## Sistem de criptare

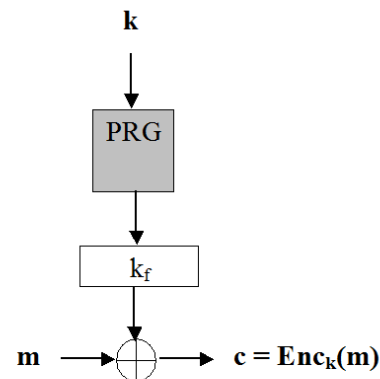
- ▶ Revenind la criptare ...
- ▶ ... aceasta presupune 2 faze:
  - ▶ **Faza 1**: se generează o secvență pseudoaleatoare de biți, folosind un **generator de numere pseudoaleatoare (PRG)**
  - ▶ **Faza 2**: secvența obținută se XOR-ează cu mesajul clar

## Sistem de criptare bazat generator de numere aleatoare

### OTP (One Time Pad)



### Sistem de criptare



## 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")

## Definiție

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

$\forall n \in \{0, 1\}^n$ ,  $G$  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  $\text{negl}$  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  $G$

- ▶  $\mathcal{D} = \text{Distinguisher}$
- ▶ 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

## Exemplu

- ▶ Consideram următorul PRG:  $G(s) = s || \bigoplus_{i=1}^n s_i$
- ▶ factorul de expansiune  $l(n) = n + 1$
- ▶ Consideram algoritmul  $D$  astfel:  $D(w) = 1$  dacă și numai dacă ultimul bit al lui  $w$  este egal cu xor-ul tuturor biților precedenți
- ▶ Se verifica ușor ca  $\Pr[D(G(s)) = 1] = 1$
- ▶ Dacă  $r$  este uniform, atunci bitul final al lui  $r$  este uniform și deci  $\Pr[D(r) = 1] = \frac{1}{2}$
- ▶  $|\frac{1}{2} - 1|$  nu e neglijabilă și deci  $G$  nu este PRG

## Observații

- ▶ Distribuția output-ului unui PRG este departe de a fi uniformă
- ▶ Exemplificăm pentru un  $G$  care dublează lungimea intrării i.e.  $l(n) = 2n$
- ▶ Pentru distribuția uniformă peste  $\{0, 1\}^{2n}$ , fiecare din cele  $2^{2n}$  este ales cu probabilitate ...
- ▶ ...  $\frac{1}{2^{2n}}$
- ▶ Considerăm distribuția output-ului lui  $G$  când primește la intrare un sir uniform de lungime  $n$
- ▶ Numărul de siruri diferite din codomeniul lui  $G$  este cel mult ...
- ▶ ...  $2^n$
- ▶ Probabilitatea ca un sir de lungime  $2n$  să fie output al lui  $G$  este  $2^n / 2^{2n} = 1/2^n$

## Observații

- ▶ Seed-ul unui PRG este analogul cheii unui sistem de criptare
- ▶ seed-ul trebuie ales uniform și menținut secret
- ▶ seed-ul trebuie să fie suficient de lung așa încât un atac prin forță brută să nu fie fezabil

## Sistem de criptare bazat pe PRG

### Definiție

Un sistem de criptare  $(Enc, Dec)$  definit peste  $(\mathcal{K}, \mathcal{M}, \mathcal{C})$  se numește *sistem de criptare bazat pe PRG* 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

### Teoremă

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