



Criptografie și Securitate

- Prelegerea 5 - Criptografie computațională

Adela Georgescu, Ruxandra F. Olimid

Facultatea de Matematică și Informatică
Universitatea din București

Cuprins

1. Securitate perfectă vs. Criptografie Computațională
2. Criptografie computațională

Securitate perfectă vs. Criptografie computațională

- ▶ Am văzut scheme de criptare care pot fi demonstrate ca fiind sigure în prezența unui adversar cu putere computațională nelimitată;

Securitate perfectă vs. Criptografie computațională

- ▶ Am văzut scheme de criptare care pot fi demonstrate ca fiind sigure în prezența unui adversar cu putere computațională nelimitată;
- ▶ Se mai numesc și **informational-teoretic sigure**;

Securitate perfectă vs. Criptografie computațională

- ▶ Am văzut scheme de criptare care pot fi demonstrate ca fiind sigure în prezența unui adversar cu putere computațională nelimitată;
- ▶ Se mai numesc și **informational-teoretic sigure**;
- ▶ Adversarul nu are suficientă informație pentru a efectua un atac;

Securitate perfectă vs. Criptografie computațională

- ▶ Am văzut scheme de criptare care pot fi demonstrate ca fiind sigure în prezența unui adversar cu putere computațională nelimitată;
- ▶ Se mai numesc și **informational-teoretic sigure**;
- ▶ Adversarul nu are suficientă informație pentru a efectua un atac;
- ▶ Majoritatea construcțiilor criptografice moderne → **securitate computațională**;

Securitate perfectă vs. Criptografie computațională

- ▶ Am văzut scheme de criptare care pot fi demonstrate ca fiind sigure în prezența unui adversar cu putere computațională nelimitată;
- ▶ Se mai numesc și **informational-teoretic sigure**;
- ▶ Adversarul nu are suficientă informație pentru a efectua un atac;
- ▶ Majoritatea construcțiilor criptografice moderne → **securitate computațională**;
- ▶ Schemele moderne *pot fi sparte* dacă un atacator are la dispoziție suficient spațiu și putere de calcul.

Securitate perfectă vs. Criptografie computațională

- Securitatea computațională mai slabă decât securitatea informațional-teoretică;

Securitate perfectă vs. Criptografie computațională

- ▶ Securitatea computațională mai slabă decât securitatea informațional-teoretică;
- ▶ Prima se bazează pe presupunții de securitate; a doua este necondiționată;

Securitate perfectă vs. Criptografie computațională

- ▶ Securitatea computațională mai slabă decât securitatea informațional-teoretică;
- ▶ Prima se bazează pe presupunții de securitate; a doua este necondiționată;
- ▶ Întrebare: de ce renunțăm la securitatea perfectă?

Securitate perfectă vs. Criptografie computațională

- ▶ Securitatea computațională mai slabă decât securitatea informațional-teoretică;
- ▶ Prima se bazează pe presupunții de securitate; a doua este necondiționată;
- ▶ **Întrebare:** de ce renunțăm la securitatea perfectă?
- ▶ **Raspuns:** datorită limitărilor practice!

Securitate perfectă vs. Criptografie computațională

- ▶ Securitatea computațională mai slabă decât securitatea informațional-teoretică;
- ▶ Prima se bazează pe presupunții de securitate; a doua este necondiționată;
- ▶ **Întrebare:** de ce renunțăm la securitatea perfectă?
- ▶ **Raspuns:** datorită limitărilor practice!
- ▶ Preferăm un compromis de securitate pentru a obține construcții practice.

Securitate computațională

Securitate computațională

- ▶ **Ideea de bază:** principiul 1 al lui Kerckhoffs

Securitate computațională

- **Ideea de bază:** principiul 1 al lui Kerckhoffs

*Un cifru trebuie să fie practic, dacă nu matematic,
indescifrabil.*

Securitate computațională

- ▶ **Ideea de bază:** principiul 1 al lui Kerckhoffs

*Un cifru trebuie să fie practic, dacă nu matematic,
indescifrabil.*

- ▶ Sunt de interes mai mare schemele care **practic nu pot fi sparte** deși nu beneficiază de securitate perfectă;

Securitate computațională

- ▶ **Ideea de bază:** principiul 1 al lui Kerckhoffs

*Un cifru trebuie să fie practic, dacă nu matematic,
indescifrabil.*

- ▶ Sunt de interes mai mare schemele care **practic nu pot fi sparte** deși nu beneficiază de securitate perfectă;
 1. Sunt sigure în fața adversarilor **eficienți** care execută atacul într-un interval de timp realizabil/fezabil;

Securitate computațională

- ▶ **Ideea de bază:** principiul 1 al lui Kerckhoffs

*Un cifru trebuie să fie practic, dacă nu matematic,
indescifrabil.*

- ▶ Sunt de interes mai mare schemele care **practic nu pot fi sparte** deși nu beneficiază de securitate perfectă;
 1. Sunt sigure în fața adversarilor **eficienți** care execută atacul într-un interval de timp realizabil/fezabil;
 2. Adversarii pot efectua un atac cu succes cu o **probabilitate foarte mică**;

Securitate computațională

- **Ideea de bază:** principiul 1 al lui Kerckhoffs

Un cifru trebuie să fie practic, dacă nu matematic, indescifrabil.

- Sunt de interes mai mare schemele care **practic nu pot fi sparte** deși nu beneficiază de securitate perfectă;
 1. Sunt sigure în fața adversarilor **eficienți** care execută atacul într-un interval de timp realizabil/fezabil;
 2. Adversarii pot efectua un atac cu succes cu o **probabilitate foarte mică**;
 3. Se impune un nouă modalitate de a defini securitatea:

Definiție

*O schemă este **sigură** dacă orice adversar care dispune de timp polinomial în n (parametrul de securitate) efectuează un atac cu succes numai cu o probabilitate neglijabilă.*

Neglijabil și ne-neglijabil

- **Întrebare:** de ce nu cerem ca probabilitatea de succes a adversarului să fie 0 (ci cerem să fie **neglijabilă**)?

Neglijabil și ne-neglijabil

- ▶ **Întrebare:** de ce nu cerem ca probabilitatea de succes a adversarului să fie 0 (ci cerem să fie **neglijabilă**)?
- ▶ **Raspuns:** pentru că adversarul poate să ghicească (cheia, mesajul clar, etc.)

Neglijabil și ne-neglijabil

- ▶ în practică: ϵ este scalar și

Neglijabil și ne-neglijabil

- ▶ în practică: ϵ este scalar și
 - ▶ ϵ ne-neglijabil dacă $\epsilon \geq 1/2^{30}$

Neglijabil și ne-neglijabil

- ▶ în practică: ϵ este scalar și
 - ▶ ϵ ne-neglijabil dacă $\epsilon \geq 1/2^{30}$
 - ▶ ϵ neglijabil dacă $\epsilon \leq 1/2^{80}$

Neglijabil și ne-neglijabil

- ▶ **în practică:** ϵ este scalar și
 - ▶ ϵ ne-neglijabil dacă $\epsilon \geq 1/2^{30}$
 - ▶ ϵ neglijabil dacă $\epsilon \leq 1/2^{80}$
- ▶ **în teorie:** ϵ este funcție $\epsilon : \mathbb{Z}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ și $p(n)$ este o funcție polinomială în n (ex.: $p(n) = n^d$, d constantă)

Neglijabil și ne-neglijabil

- ▶ **în practică:** ϵ este scalar și
 - ▶ ϵ ne-neglijabil dacă $\epsilon \geq 1/2^{30}$
 - ▶ ϵ neglijabil dacă $\epsilon \leq 1/2^{80}$
- ▶ **în teorie:** ϵ este funcție $\epsilon : \mathbb{Z}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ și $p(n)$ este o funcție polinomială în n (ex.: $p(n) = n^d$, d constantă)
 - ▶ ϵ ne-neglijabilă în n dacă $\exists p(n) : \epsilon(n) \geq 1/p(n)$

Neglijabil și ne-neglijabil

- ▶ **în practică:** ϵ este scalar și
 - ▶ ϵ ne-neglijabil dacă $\epsilon \geq 1/2^{30}$
 - ▶ ϵ neglijabil dacă $\epsilon \leq 1/2^{80}$
- ▶ **în teorie:** ϵ este funcție $\epsilon : \mathbb{Z}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ și $p(n)$ este o funcție polinomială în n (ex.: $p(n) = n^d$, d constantă)
 - ▶ ϵ ne-neglijabilă în n dacă $\exists p(n) : \epsilon(n) \geq 1/p(n)$
 - ▶ ϵ neglijabilă în n dacă $\forall p(n), \exists n_d$ a.î. $\forall n \geq n_d : \epsilon(n) < 1/p(n)$

Neglijabil și ne-neglijabil

- **Întrebare:** de ce această definiție și nu alta?

$$\epsilon(n) \text{ negl. în } n \Leftrightarrow \forall p(n), \exists n_d \text{ a.î. } \forall n \geq n_d : \epsilon(n) < 1/p(n)$$

Neglijabil și ne-neglijabil

- ▶ **Întrebare:** de ce această definiție și nu alta?

$\epsilon(n)$ negl. în $n \Leftrightarrow \forall p(n), \exists n_d \text{ a.î. } \forall n \geq n_d : \epsilon(n) < 1/p(n)$

- ▶ **Răspuns:**

- ▶ Atacul are loc cu probabilitate $\epsilon(n)$...

Neglijabil și ne-neglijabil

- ▶ **Întrebare:** de ce această definiție și nu alta?

$\epsilon(n)$ negl. în $n \Leftrightarrow \forall p(n), \exists n_d \text{ a.î. } \forall n \geq n_d : \epsilon(n) < 1/p(n)$

- ▶ **Răspuns:**

- ▶ Atacul are loc cu probabilitate $\epsilon(n)$...
- ▶ ... deci trebuie repetat de aprox. $1/\epsilon(n)$ ori ca să reușească

Neglijabil și ne-neglijabil

- ▶ **Întrebare:** de ce această definiție și nu alta?

$\epsilon(n)$ negl. în $n \Leftrightarrow \forall p(n), \exists n_d \text{ a.î. } \forall n \geq n_d : \epsilon(n) < 1/p(n)$

- ▶ **Răspuns:**

- ▶ Atacul are loc cu probabilitate $\epsilon(n)$...
- ▶ ... deci trebuie repetat de aprox. $1/\epsilon(n)$ ori ca să reușească
- ▶ Dar din definiție $1/\epsilon(n) > p(n)$...

Neglijabil și ne-neglijabil

- ▶ **Întrebare:** de ce această definiție și nu alta?

$\epsilon(n)$ negl. în $n \Leftrightarrow \forall p(n), \exists n_d \text{ a.î. } \forall n \geq n_d : \epsilon(n) < 1/p(n)$

- ▶ **Răspuns:**

- ▶ Atacul are loc cu probabilitate $\epsilon(n)$...
- ▶ ... deci trebuie repetat de aprox. $1/\epsilon(n)$ ori ca să reușească
- ▶ Dar din definiție $1/\epsilon(n) > p(n)$...
- ▶ ... deci necesită un timp super-polinomial în n

Neglijabil și ne-neglijabil

- ▶ **Întrebare:** de ce această definiție și nu alta?

$\epsilon(n)$ negl. în $n \Leftrightarrow \forall p(n), \exists n_d \text{ a.î. } \forall n \geq n_d : \epsilon(n) < 1/p(n)$

- ▶ **Răspuns:**

- ▶ Atacul are loc cu probabilitate $\epsilon(n)$...
- ▶ ... deci trebuie repetat de aprox. $1/\epsilon(n)$ ori ca să reușească
- ▶ Dar din definiție $1/\epsilon(n) > p(n)$...
- ▶ ... deci necesită un timp super-polinomial în n

Definiția semnifică faptul că sistemul rămâne sigur pentru un adversar **PPT (Probabilistic Polynomial în Timp)**

Important de reținut!

- ▶ Securitate perfectă vs. securitate computațională
- ▶ Neglijabil vs. ne-neglijabil