Criptografie și Securitate

- Prelegerea 20 -
Permutări cu trapă secretă

Adela Georgescu, Ruxandra F. Olimid

Facultatea de Matematică și Informatică
Universitatea din București

Cuprins

1. Definiție

2. Problema rucsacului

3. Construcția sistemelor de criptare asimetrice

Permutări cu trapă secretă

- Reamintim noțiunea de funcție **one-way**;

Permutări cu trapă secretă

- Reamintim noțiunea de funcție **one-way**;
- Aceasta este o funcție pentru care este **ușor** de calculat valoarea funcției...

Permutări cu trapă secretă

- Reamintim noțiunea de funcție **one-way**;
- Aceasta este o funcție pentru care este **ușor** de calculat valoarea funcției...
- ... dar este **dificil** de calculat valoarea funcției inverse;

Permutări cu trapă secretă

- Reamintim noțiunea de funcție **one-way**;
- Aceasta este o funcție pentru care este **ușor** de calculat valoarea funcției...
- ... dar este **dificil** de calculat valoarea funcției inverse;
- Am întâlnit noțiunea când am studiat funcțiile hash;

Permutări cu trapă secretă

- Reamintim noțiunea de funcție **one-way**;
- Aceasta este o funcție pentru care este **ușor** de calculat valoarea funcției...
- ... dar este **dificil** de calculat valoarea funcției inverse;
- Am întâlnit noțiunea când am studiat funcțiile hash;
- Dacă $H : \{0,1\}^* \rightarrow \{0,1\}^n$ este o funcție hash (rezistentă la prima preimagine), atunci:

Permutări cu trapă secretă

- Reamintim noțiunea de funcție **one-way**;
- Aceasta este o funcție pentru care este **ușor** de calculat valoarea funcției...
- ... dar este **dificil** de calculat valoarea funcției inverse;
- Am întâlnit noțiunea când am studiat funcțiile hash;
- Dacă $H : \{0,1\}^* \rightarrow \{0,1\}^n$ este o funcție hash (rezistentă la prima preimagine), atunci:
  - Fiind dat x, este *eficient* de calculat $H(x)$;

Permutări cu trapă secretă

- Reamintim noțiunea de funcție **one-way**;
- Aceasta este o funcție pentru care este **ușor** de calculat valoarea funcției...
- ... dar este **dificil** de calculat valoarea funcției inverse;
- Am întâlnit noțiunea când am studiat funcțiile hash;
- Dacă $H : \{0,1\}^* \rightarrow \{0,1\}^n$ este o funcție hash (rezistentă la prima preimagine), atunci:
    - Fiind dat x, este *eficient* de calculat $H(x)$;
    - Cunoscând $H(x)$ este *(computațional) dificil* de calculat x.

Permutări cu trapă secretă

- Definim noțiunea de permutare cu trapă secretă sau TDP (TrapDoor Permutation);

Permutări cu trapă secretă

- Definim noțiunea de permutare cu trapă secretă sau TDP (TrapDoor Permutation);
- Acesta este o permutare one-way...

Permutări cu trapă secretă

- Definim noțiunea de permutare cu trapă secretă sau TDP (TrapDoor Permutation);
- Acesta este o permutare one-way...
- ... care permite calculul eficient al inversului dacă se cunoaște o informație adițională, numită cheie secretă;

Permutări cu trapă secretă

- Definim noțiunea de permutare cu trapă secretă sau TDP (TrapDoor Permutation);
- Acesta este o permutare one-way...
- ... care permite calculul eficient al inversului dacă se cunoaște o informație adițională, numită cheie secretă;
- Utilizarea cheii secrete permite deținătorului să folosească o trapă secretă, de unde provine și denumirea construcției.

Permutări cu trapă secretă

Definiție

O permutare cu trapă secretă sau TDP (TrapDoor Permutation) este un triplet (Gen, F, F^{-1}) *unde:*

1. *Gen este un algoritm nedeterminist PPT care generează o pereche de chei* (pk, sk)*;*
2. $F(pk, \cdot) : \mathcal{X} \rightarrow \mathcal{X}$ *este o funcție one-way;*
3. $F^{-1}(sk, \cdot) : \mathcal{X} \rightarrow \mathcal{X}$ *este o funcție eficient calculabilă;*

$$\forall x \in \mathcal{X}, F^{-1}(sk, F(pk, x)) = x$$

Permutări cu trapă secretă

Definiție

O *permutare cu trapă secretă* *sau* *TDP (TrapDoor Permutation)* *este un triplet* (Gen, F, F^{-1}) *unde:*

1. *Gen este un algoritm nedeterminist PPT care generează o pereche de chei* (pk, sk)*;*
2. $F(pk, \cdot) : \mathcal{X} \rightarrow \mathcal{X}$ *este o funcție one-way;*
3. $F^{-1}(sk, \cdot) : \mathcal{X} \rightarrow \mathcal{X}$ *este o funcție eficient calculabilă;*

$$\forall x \in \mathcal{X}, F^{-1}(sk, F(pk, x)) = x$$

- F este *sigură* dacă poate fi eficient evaluată, dar nu poate fi inversată fără cunoașterea cheii secrete sk (decât cu probabilitate neglijabilă);

Permutări cu trapă secretă

Definiție

O *permutare cu trapă secretă* *sau* *TDP (TrapDoor Permutation)* *este un triplet* (Gen, F, F^{-1}) *unde:*

1. *Gen este un algoritm nedeterminist PPT care generează o pereche de chei* (pk, sk)*;*
2. $F(pk, \cdot) : \mathcal{X} \to \mathcal{X}$ *este o funcție one-way;*
3. $F^{-1}(sk, \cdot) : \mathcal{X} \to \mathcal{X}$ *este o funcție eficient calculabilă;*

$$\forall x \in \mathcal{X}, F^{-1}(sk, F(pk, x)) = x$$

- F este *sigură* dacă poate fi eficient evaluată, dar nu poate fi inversată fără cunoașterea cheii secrete sk (decât cu probabilitate neglijabilă);
- Notații: $F(pk, \cdot) = F_{pk}(\cdot)$, $F^{-1}(sk, \cdot) = F_{sk}^{-1}(\cdot)$.

Problema rucsacului

- Un exemplu de funcție *one-way* este problema rucsacului;

Problema rucsacului

- Un exemplu de funcție *one-way* este problema rucsacului;
- Se dă un vector $A = (a_1, a_2, \ldots, a_n)$ de n elemente distincte $a_i \in \mathbb{Z}_+$ și o valoare $k \in \mathbb{Z}_+$;

Problema rucsacului

- Un exemplu de funcție *one-way* este problema rucsacului;
- Se dă un vector $A = (a_1, a_2, \ldots, a_n)$ de n elemente distincte $a_i \in \mathbb{Z}_+$ și o valoare $k \in \mathbb{Z}_+$;
- Se cere să se determine elementele vectorului a căror sumă este k;

Problema rucsacului

- Un exemplu de funcție *one-way* este problema rucsacului;
- Se dă un vector $A = (a_1, a_2, \ldots, a_n)$ de n elemente distincte $a_i \in \mathbb{Z}_+$ și o valoare $k \in \mathbb{Z}_+$;
- Se cere să se determine elementele vectorului a căror sumă este k;
- Pentru un vector de n elemente, problema se poate rezolva verificând pe rând toate submulțimile lui A;

Problema rucsacului

- Un exemplu de funcție *one-way* este problema rucsacului;
- Se dă un vector $A = (a_1, a_2, \ldots, a_n)$ de n elemente distincte $a_i \in \mathbb{Z}_+$ și o valoare $k \in \mathbb{Z}_+$;
- Se cere să se determine elementele vectorului a căror sumă este k;
- Pentru un vector de n elemente, problema se poate rezolva verificând pe rând toate submulțimile lui A;
- Cum numărul submulțimilor este de în $2^n - 1$, această modalitate de rezolvare este imposibilă pentru n mare;

Problema rucsacului

- Un exemplu de funcție *one-way* este problema rucsacului;
- Se dă un vector $A = (a_1, a_2, \ldots, a_n)$ de n elemente distincte $a_i \in \mathbb{Z}_+$ și o valoare $k \in \mathbb{Z}_+$;
- Se cere să se determine elementele vectorului a căror sumă este k;
- Pentru un vector de n elemente, problema se poate rezolva verificând pe rând toate submulțimile lui A;
- Cum numărul submulțimilor este de în $2^n - 1$, această modalitate de rezolvare este imposibilă pentru n mare;
- Problema este (în general) dificilă.

Problema rucsacului

- Există însă clase ușoare ale problemei rucsacului;

Problema rucsacului

- Există însă clase ușoare ale problemei rucsacului;
- Una dintre acestea o reprezintă vectorii super-crescători;

Problema rucsacului

- Există însă clase ușoare ale problemei rucsacului;
- Una dintre acestea o reprezintă vectorii super-crescători;
- Un vector $A = (a_1, a_2, \ldots, a_n)$ este *super-crescător* dacă satisface:

$$\forall j \geq 2, a_j > \sum_{i=1}^{j-1} a_i$$

Problema rucsacului

- Există însă clase ușoare ale problemei rucsacului;
- Una dintre acestea o reprezintă vectorii super-crescători;
- Un vector $A = (a_1, a_2, \ldots, a_n)$ este *super-crescător* dacă satisface:

$$\forall j \geq 2, a_j > \sum_{i=1}^{j-1} a_i$$

- Un exemplu este vectorul:

$$A = \{1, 3, 5, 11, 21, 44, 87\}$$

$3 > 1$
$5 > 1 + 3$
$11 > 1 + 3 + 5$
$21 > 1 + 3 + 5 + 11$
$44 > 1 + 3 + 5 + 11 + 21$
$87 > 1 + 3 + 5 + 11 + 21 + 44$

Problema rucsacului

- Dăm un algoritm de rezolvare a problemei rucsacului pentru vectori super-crescători;

Problema rucsacului

- Dăm un algoritm de rezolvare a problemei rucsacului pentru vectori super-crescători;
- Cunoscând k, se parcurge vectorul de la dreapta spre stânga;

Problema rucsacului

- Dăm un algoritm de rezolvare a problemei rucsacului pentru vectori super-crescători;
- Cunoscând k, se parcurge vectorul de la dreapta spre stânga;
- Dacă $k \geq a_i$, atunci a_i face parte din sumă (suma tuturor celorlalte elemente este mai mică decât a_i);

Problema rucsacului

- Dăm un algoritm de rezolvare a problemei rucsacului pentru vectori super-crescători;
- Cunoscând k, se parcurge vectorul de la dreapta spre stânga;
- Dacă $k \geq a_i$, atunci a_i face parte din sumă (suma tuturor celorlalte elemente este mai mică decât a_i);
- Dacă a_i face parte din sumă, atunci valoarea k se actualizează cu $k - a_i$;

Problema rucsacului

- Dăm un algoritm de rezolvare a problemei rucsacului pentru vectori super-crescători;
- Cunoscând k, se parcurge vectorul de la dreapta spre stânga;
- Dacă $k \geq a_i$, atunci a_i face parte din sumă (suma tuturor celorlalte elemente este mai mică decât a_i);
- Dacă a_i face parte din sumă, atunci valoarea k se actualizează cu $k - a_i$;
- Se repetă procedeul până se parcurge întreg vectorul sau k devine 0.

Problema rucsacului

- Pentru exemplul anterior $A = \{1, 3, 5, 11, 21, 44, 87\}$, fie $k = 58$;

Problema rucsacului

- Pentru exemplul anterior $A = \{1, 3, 5, 11, 21, 44, 87\}$, fie $k = 58$;
- Se obține:
  $k = 58 < 87 \Rightarrow 87$ nu apare în sumă

Problema rucsacului

- Pentru exemplul anterior $A = \{1, 3, 5, 11, 21, 44, 87\}$, fie $k = 58$;
- Se obține:
  $k = 58 < 87 \Rightarrow 87$ nu apare în sumă
  $k = 58 > 44 \Rightarrow 44$ apare în sumă și $k = 58 - 44 = 14$

Problema rucsacului

- Pentru exemplul anterior $A = \{1, 3, 5, 11, 21, 44, 87\}$, fie $k = 58$;
- Se obține:
  $k = 58 < 87 \Rightarrow 87$ nu apare în sumă
  $k = 58 > 44 \Rightarrow 44$ apare în sumă și $k = 58 - 44 = 14$
  $k = 14 < 21 \Rightarrow 21$ nu apare în sumă

Problema rucsacului

- Pentru exemplul anterior $A = \{1, 3, 5, 11, 21, 44, 87\}$, fie $k = 58$;
- Se obține:
  $k = 58 < 87 \Rightarrow 87$ nu apare în sumă
  $k = 58 > 44 \Rightarrow 44$ apare în sumă și $k = 58 - 44 = 14$
  $k = 14 < 21 \Rightarrow 21$ nu apare în sumă
  $k = 14 > 11 \Rightarrow 11$ apare în sumă și $k = 14 - 11 = 3$

Problema rucsacului

- Pentru exemplul anterior $A = \{1, 3, 5, 11, 21, 44, 87\}$, fie $k = 58$;
- Se obține:
  $k = 58 < 87 \Rightarrow 87$ nu apare în sumă
  $k = 58 > 44 \Rightarrow 44$ apare în sumă și $k = 58 - 44 = 14$
  $k = 14 < 21 \Rightarrow 21$ nu apare în sumă
  $k = 14 > 11 \Rightarrow 11$ apare în sumă și $k = 14 - 11 = 3$
  $k = 3 < 5 \Rightarrow 5$ nu apare în sumă

Problema rucsacului

- Pentru exemplul anterior $A = \{1, 3, 5, 11, 21, 44, 87\}$, fie $k = 58$;
- Se obține:
  $k = 58 < 87 \Rightarrow 87$ nu apare în sumă
  $k = 58 > 44 \Rightarrow 44$ apare în sumă și $k = 58 - 44 = 14$
  $k = 14 < 21 \Rightarrow 21$ nu apare în sumă
  $k = 14 > 11 \Rightarrow 11$ apare în sumă și $k = 14 - 11 = 3$
  $k = 3 < 5 \Rightarrow 5$ nu apare în sumă
  $k = 3 = 3 \Rightarrow 3$ apare în sumă și $k = 3 - 3 = 0$

Problema rucsacului

- Pentru exemplul anterior $A = \{1, 3, 5, 11, 21, 44, 87\}$, fie $k = 58$;
- Se obține:
  $k = 58 < 87 \Rightarrow 87$ nu apare în sumă
  $k = 58 > 44 \Rightarrow 44$ apare în sumă și $k = 58 - 44 = 14$
  $k = 14 < 21 \Rightarrow 21$ nu apare în sumă
  $k = 14 > 11 \Rightarrow 11$ apare în sumă și $k = 14 - 11 = 3$
  $k = 3 < 5 \Rightarrow 5$ nu apare în sumă
  $k = 3 = 3 \Rightarrow 3$ apare în sumă și $k = 3 - 3 = 0$
- S-a obținut deci $k = 44 + 11 + 3$.

Problema rucsacului

- Transformăm o problemă **simplă** a rucsacului într-o problemă **dificilă** pe baza unei informații secrete și obținem astfel o **funcție cu trapă secretă**;

Problema rucsacului

- Transformăm o problemă **simplă** a rucsacului într-o problemă **dificilă** pe baza unei informații secrete și obținem astfel o **funcție cu trapă secretă**;
- Fie un vector supercrescător $A = (a_1, a_2, \ldots, a_n)$;

Problema rucsacului

- Transformăm o problemă **simplă** a rucsacului într-o problemă **dificilă** pe baza unei informații secrete și obținem astfel o **funcție cu trapă secretă**;
- Fie un vector supercrescător $A = (a_1, a_2, \ldots, a_n)$;
- Se aleg un **modul** m și un **multiplicator** t a.î. $gcd(c, m) = 1$;

Problema rucsacului

- Transformăm o problemă **simplă** a rucsacului într-o problemă **dificilă** pe baza unei informații secrete și obținem astfel o **funcție cu trapă secretă**;
- Fie un vector supercrescător $A = (a_1, a_2, \ldots, a_n)$;
- Se aleg un **modul** m și un **multiplicator** t a.î. $gcd(c, m) = 1$;
- Se calculează $B = (b_1, b_2, \ldots, b_n)$, unde $b_i = a_i \cdot t \pmod m$;

Problema rucsacului

- Transformăm o problemă **simplă** a rucsacului într-o problemă **dificilă** pe baza unei informații secrete și obținem astfel o **funcție cu trapă secretă**;
- Fie un vector supercrescător $A = (a_1, a_2, \ldots, a_n)$;
- Se aleg un **modul** m și un **multiplicator** t a.î. $gcd(c, m) = 1$;
- Se calculează $B = (b_1, b_2, \ldots, b_n)$, unde $b_i = a_i \cdot t \pmod{m}$;
- Cunoscând A problema este simplă, dar cunoscând B problema este dificilă.

Problema rucsacului

- Pentru exemplul anterior: $A = \{1, 3, 5, 11, 21, 44, 87\}$, fie $t = 43$ și $m = 1590$;

Problema rucsacului

- Pentru exemplul anterior: $A = \{1, 3, 5, 11, 21, 44, 87\}$, fie $t = 43$ și $m = 1590$;
- Se obține $B = \{43, 129, 215, 473, 903, 302, 561\}$;

Problema rucsacului

- Pentru exemplul anterior: $A = \{1, 3, 5, 11, 21, 44, 87\}$, fie $t = 43$ și $m = 1590$;
- Se obține $B = \{43, 129, 215, 473, 903, 302, 561\}$;
- Se cere rezolvarea problemei rucsac pentru $k = 904$ și B, care este dificilă (facem abstracție de dimensiunea lui n);

Problema rucsacului

- Pentru exemplul anterior: $A = \{1, 3, 5, 11, 21, 44, 87\}$, fie $t = 43$ și $m = 1590$;
- Se obține $B = \{43, 129, 215, 473, 903, 302, 561\}$;
- Se cere rezolvarea problemei rucsac pentru $k = 904$ și B, care este dificilă (facem abstracție de dimensiunea lui n);
- Pentru deținătorul trapei secrete $(t, m) = (43, 1590)$ problema devine ușoară;

Problema rucsacului

- Pentru exemplul anterior: $A = \{1, 3, 5, 11, 21, 44, 87\}$, fie $t = 43$ și $m = 1590$;
- Se obține $B = \{43, 129, 215, 473, 903, 302, 561\}$;
- Se cere rezolvarea problemei rucsac pentru $k = 904$ și B, care este dificilă (facem abstracție de dimensiunea lui n);
- Pentru deținătorul trapei secrete $(t, m) = (43, 1590)$ problema devine ușoară;
- Aceasta se rezumă la rezolvarea problemei pentru $k = 904 \cdot 43^{-1} \pmod{1590} = 58$ și A pe care am rezolvat-o anterior.

Algoritmul lui Euclid extins

- Pentru calculul $43^{-1} \pmod{1590}$ am folosit algoritmul lui Euclid extins;

Algoritmul lui Euclid extins

- Pentru calculul $43^{-1} \pmod{1590}$ am folosit algoritmul lui Euclid extins;
- Se fac împărțiri cu rest repetate (împărțitorul se împarte la rest) până se obține restul 1:
  $1590 = 43 \cdot 36 + 42$
  $43 = 42 \cdot 1 + 1$

Algoritmul lui Euclid extins

- Pentru calculul $43^{-1} \pmod{1590}$ am folosit algoritmul lui Euclid extins;
- Se fac împărțiri cu rest repetate (împărțitorul se împarte la rest) până se obține restul 1:
  $1590 = 43 \cdot 36 + 42$
  $43 = 42 \cdot 1 + 1$
- Se înlocuiesc valorile restului în sens invers:
  $1 = 43 - 42 \pmod{1590}$
  $1 = 43 - (1590 - 43 \cdot 36) \pmod{1590} = 43 \cdot 37 \pmod{1590}$

Algoritmul lui Euclid extins

- Pentru calculul $43^{-1} \pmod{1590}$ am folosit algoritmul lui Euclid extins;
- Se fac împărțiri cu rest repetate (împărțitorul se împarte la rest) până se obține restul 1:
  $1590 = 43 \cdot 36 + 42$
  $43 = 42 \cdot 1 + 1$
- Se înlocuiesc valorile restului în sens invers:
  $1 = 43 - 42 \pmod{1590}$
  $1 = 43 - (1590 - 43 \cdot 36) \pmod{1590} = 43 \cdot 37 \pmod{1590}$
- Cum $43 \cdot 37 \pmod{1590} = 1 \Rightarrow 43^{-1} \pmod{1590} = 37$.

Construcția sistemelor de criptare asimetrice

- Folosim TDP pentru construcția sistemelor de criptare asimetrice;

Construcție

Fie (Gen, F, F^{-1}) *TDP cu* $F : \mathcal{X} \rightarrow \mathcal{Y}$*,* (Enc, Dec) *un sistem de criptare simetric sigur cu autentificarea mesajelor definit peste* $(\mathcal{X}, \mathcal{Y})$ *și* $H : \mathcal{X} \rightarrow \mathcal{K}$ *o funcție hash. Definim un sistem de criptare asimetrică peste* $(\mathcal{K}, \mathcal{X}, \mathcal{Y})$ *în felul următor:*

- $\mathrm{Enc_{pk}(m)} = (y, c) = (F_{pk}(x), Enc_k(m))$*, unde* $k = H(x)$ *și* $x \leftarrow^R \mathcal{X}$*;*
- $\mathrm{Dec_{sk}(y, c)} = Dec_k(c)$*, unde* $k = H(x)$ *și* $x = F_{sk}^{-1}(y)$*;*

Exemple

- Merkle-Hellman
  - definit în 1978 de R.Merkle și M.Hellman
  - bazat pe problema rucsacului
  - spart la numai câțiva ani de la publicare

Exemple

- Merkle-Hellman
  - definit în 1978 de R.Merkle și M.Hellman
  - bazat pe problema rucsacului
  - spart la numai câțiva ani de la publicare
- RSA
  - definit în 1977 de R.Rivest, A.Shamir și L.Adleman
  - bazat pe problema RSA și indirect a factorizării numerelor mari
  - cel mai cunoscut sistem de criptare cu cheie publică

Important de reținut!

- Noțiunea de permutare cu trapă secretă (TDP)
- Construcția sistemelor de criptare asimetrice folosind TDP