Crypt(?) -

Objectives
Requirements / Goals / Attributes / ...

CIA Triad
- Confidentiality
- Integrity
- Availability

Authentication

Non-repudiation

Terminology

Cryptology Cryptography Cryptanalysis Cryptogram	Attack Adversary Corrupted / Malicious Party	Defences Mitigations Countermeasures

Cryptology

Security

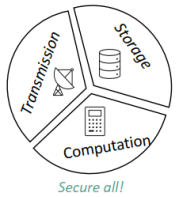
1

Electronical computation

Quantum computation

Pen & paper

Mechanical computation



Secure all!

Attacks		
Passive	vs.	Active
Outsider		Insider

Deteriorate

Corrupt

Impersonate

Principles -

Kerckhoffs's principle
Only keep hidden the key.
(e.g., make the design public)

Principle of sufficient keys
The number of possible keys must be large.
(e.g., avoid brute force)

Principle of (key) separation
Use different keys for different contexts, compartmentalize.
(e.g., minimise the damage of a leak)

2

Principle of simplicity
Keep everything simple.
(e.g., unnecessary complexity brings in risks)

Principle of diversity
Use different types of ... e.g., cryptographic algorithms.
(e.g., avoid same attacks against all)

Security by default
Keep default configuration as secure as possible.
(e.g., deny access by default)

Principle of minimal trust
Minimise the number of trusted entities, don't trust easily.
(e.g., do not say your secrets to anyone)

Principle of the weakest link
A system cannot be more secure than its weakest component (link).
(e.g., secure all components)

Principle of least privilege
Grant the exact privileges required to perform the job.
(e.g., do not grand less or more privileges)

Security by design
Build in security from start.
(e.g., integrate security in all design and development stages)

Principle of modularization
Keep things modular.
(e.g., easily change one component with another)

Defence in depth
Use diverse security strategies at different layers.
(e.g., use physical and technological security)

Ethics!

Security through obscurity (?)
Oblivious Transfer, Obfuscation, Covert Channels, ... ; Kleptography; Standardisation ...

Unconditional vs. Conditional Security -

Unconditional (Information-theoretic)	Conditional (Computational)
... security	
An *adversary* with **no restrictions** (unbounded computational resources - time, memory) **cannot break** the scheme.	An *adversary* with **computational restrictions** (**bounded** time, memory) **can break** the scheme with **some (negligible) probability**.

4

Stands against brute force ⊕
Good in theory, poor in practice ⊖

⊕ Suitable for practice
⊖ Weaker than unconditional security

Adversary $\mathcal{A}$

Cryptographic scheme

A cryptographic construction satisfies *computational security* if any adversary $\mathcal{A}$ that runs the attack in a time $t(n)$ succeeds the attack with probability at most $\varepsilon(n)$; t and ε are functions of a *computational security parameter* n.

Statistical Security

A cryptographic construction satisfies $\varepsilon(\lambda)$ *statistical security* if any unbounded adversary $\mathcal{A}$ succeeds the attack with probability at most $\varepsilon(\lambda)$; ε is function of a *statistical security parameter* λ.
- Introduces a *small* advantage $\varepsilon(\lambda)$ wrt the *a-priory probability* of winning

Statistical and *computational* security are both *relaxations* of information-theoretical security.

PPT (Probabilistic Polynomial in Time) Adversary

- $t(n)$ is **polynomial** in n
- $\varepsilon(n)$ is **negligible** in n

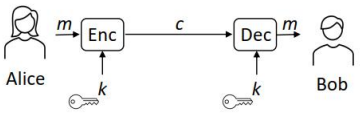
Negligibility:
$\forall p(n), \exists n_d$ such that $\forall n \geq n_d$ it holds $\varepsilon(n) < 1/p(n)$
$p(n) = n^d$ and d constant

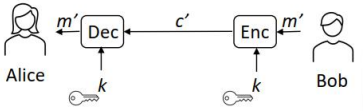
Examples:
✗ $1/2$, $1/n^{100}$
✓ $1/2^n$, $p(n)/2^n$

Symmetric vs. Asymmetric Encryption -

Symmetric	Asymmetric
... encryption	

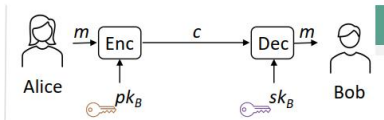
3

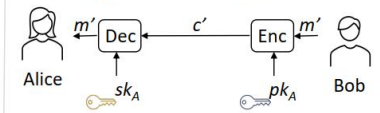




Encryption: $c = \text{Enc}(k,m)$
Decryption: $m = \text{Dec}(k,c)$
Correctness: $\forall m \in \mathcal{M}, k \in \mathcal{K}$
$\text{Dec}(k,\text{Enc}(k,m)) = m$

Shorter keys ⊕
Key establishment ⊖





⊕ Private keys never leave the owner
⊖ Computational cost & speed

Encryption: $c = \text{Enc}(pk_B,m)$
Decryption: $m = \text{Dec}(sk_B,c)$
Correctness: $\forall m \in \mathcal{M}, (pk_B, sk_B) \in \mathcal{K}$
$\text{Dec}(sk_B,\text{Enc}(pk_B,m)) = m$

Terminology

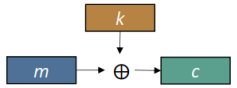
k: symmetric key	*m*: plaintext
pk: public key	*c*: ciphertext
sk: private (secret) key	Enc: encryption alg.
(pk,sk): public-private key pair	Dec: decryption alg.
	Cryptanalysis

Confidentiality

No. of keys
for N bi-directional communicating parties

Each: N-1 [k]	vs.	Each: 1 [sk], N-1 [pk]
Total: N(N-1)/2 [k]		Total: N [sk], N [pk]

One Time Pad (OTP) - Cryptographic Hash Function - Stream Ciphers -



Encryption: $c = k \oplus m$
Decryption: $m = k \oplus c$

m(c) \ k	0	1
0	0	1
1	1	0

(⊕ (XOR))

The key *k*:
- is as long as the plaintext *m* and the ciphertext *c*
- is uniformly random chosen in $\mathcal{K}$
- must be used only once

Examples

k: 0 1 1 0 1 1 0 0 ⊕
m: 1 0 1 1 1 0 0 1
c: 1 1 0 1 0 1 0 1

k: G F N O M ⊕
m: P A G E S (mod 26)
c: V F T S E

Perfect Secrecy

For all *m* possible plaintext (i.e., all *m* in $\mathcal{M}$) and any *c* ciphertext (i.e., all *c* in $\mathcal{C}$) such that $Pr[C=c]>0$, it holds:

$$Pr[M=m \mid C=c] = Pr[M=m]$$

Theorem (key length bounding):
Let (Enc, Dec) be a perfectly-secret encryption scheme over a plaintext space $\mathcal{M}$ and a key space $\mathcal{K}$. Then it holds that $|\mathcal{K}| \geq |\mathcal{M}|$ (i.e., the length of the key is larger or equal to the length of the message).

⊕ Easy, fast encryption and decryption
⊖ Long key length

Multiple use of the same key k

$c_1 = k \oplus m_1$, $c_2 = k \oplus m_2$, ...

Attack 1. $\mathcal{A}$ knows the ciphertexts c_1, c_2
$\mathcal{A}$ finds a relation between the plaintexts: $m_1 \oplus m_2 = c_1 \oplus c_2$

Attack 2: $\mathcal{A}$ knows (at least) the pair (m_1, c_1)
$\mathcal{A}$ finds the key $k = m_1 \oplus c_1$, then decrypts $m_2 = k \oplus c_2$

Perfect secrecy

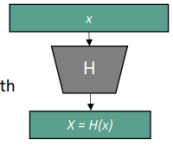
5

Pages on SecuRity by Ruxandra F. Olimid
https://pagesonsecurity.blogspot.com/

Cryptographic Hash Function

$H: \{0,1\}^* \rightarrow \{0,1\}^{l(n)}$

- arbitrary input length, fixed output length
- deterministic
- "easy" to compute, "difficult" to invert



⊖ *Attacks:* *Birthday attack*

l(n) = poly(n), with n the security parameter
$\{0,1\}^*$: sequence on bits, regardless its size
s.t.: such that
$\mathcal{A}$: adversary

6

Security

Collision resistance

$\text{Hash}^{coll}_{\mathcal{A},H}(n)=1$ if
$\mathcal{A}$ outputs $x,y \in \{0,1\}^*$ s.t.
$x \neq y$ and $H(x) = H(y)$
$\text{Hash}^{coll}_{\mathcal{A},H}(n)=0$, otherwise

H is *collision resistant* if
$\forall \mathcal{A}$ PPT, $\exists\ \varepsilon(n)$ negligible s.t.:
$Pr[\text{Hash}^{coll}_{\mathcal{A},H}(n)=1] \leq \varepsilon(n)$

Second pre-image resistance

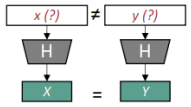
$\text{Hash}^{2nd\text{-}pre\text{-}img}_{\mathcal{A},H}(n)=1$ if
given $x \leftarrow^R \{0,1\}^*$,
$\mathcal{A}$ outputs $y \in \{0,1\}^*$ s.t.
$x \neq y$ and $H(x) = H(y)$
$\text{Hash}^{2nd\text{-}pre\text{-}img}_{\mathcal{A},H}(n)=0$, otherwise

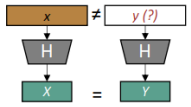
H is *second pre-image resistant* if
$\forall \mathcal{A}$ PPT, $\exists\ \varepsilon(n)$ negligible s.t.:
$Pr[\text{Hash}^{2nd\text{-}pre\text{-}img}_{\mathcal{A},H}(n)=1] \leq \varepsilon(n)$

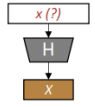
First pre-image resistance

$\text{Hash}^{1st\text{-}pre\text{-}img}_{\mathcal{A},H}(n)=1$ if
given $X = H(x')$, $x' \leftarrow^R \{0,1\}^*$,
$\mathcal{A}$ outputs $x \in \{0,1\}^*$ s.t.
$H(x) = X$
$\text{Hash}^{1st\text{-}pre\text{-}img}_{\mathcal{A},H}(n)=0$, otherwise

H is *first pre-image resistant* if
$\forall \mathcal{A}$ PPT, $\exists\ \varepsilon(n)$ negligible s.t.:
$Pr[\text{Hash}^{1st\text{-}pre\text{-}img}_{\mathcal{A},H}(n)=1] \leq \varepsilon(n)$







one-way function

higher security ← → lower security

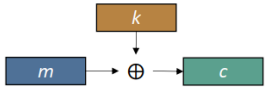
Pages on SecuRity by Ruxandra F. Olimid
https://pagesonsecurity.blogspot.com/

Stream Ciphers

One Time Pad (OTP)

Perfect secrecy

Encryption: $c = k \oplus m$
Decryption: $m = k \oplus c$

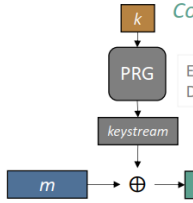


Stream Ciphers

Computational secrecy

Encryption: $c = PRG(k) \oplus m$
Decryption: $m = PRG(k) \oplus c$

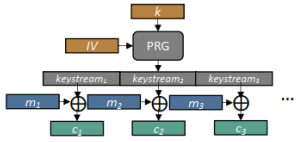
PRG: Pseudo-Random Generator
IV: Initialization Vector



7

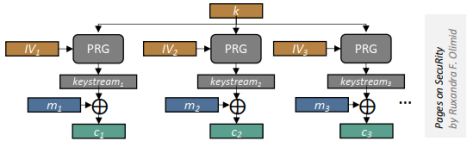
Synchronized Mode

Encryption: $c_1 \,||\, c_2 \,||\, c_3 \ldots = (IV,\ PRG(k,IV) \oplus m_1 \,||\, m_2 \,||\, m_3 \ldots)$
Decryption: $m_1 \,||\, m_2 \,||\, m_3 \ldots = PRG(k,IV) \oplus c_1 \,||\, c_2 \,||\, c_3 \ldots$
IV chosen uniformly at random



Unsynchronized Mode

Encryption: $c_i = (IV_i,\ PRG(k,IV_i) \oplus m_i)$
Decryption: $m_i = PRG(k,IV_i) \oplus c_i$
$IV_1, IV_2, \ldots$ chosen uniformly at random (and thus independent)



Pages on SecuRity by Ruxandra F. Olimid
https://pagesonsecurity.blogspot.com/

Obiectivele criptografiei

Confidențialitate: *păstrarea secretului informației, accesul la informația sensibilă fiind disponibilă doar persoanelor autorizate.*

Integritate (a datelor): *eliminarea posibilității de modificare (schimbare, inserare, ștergere) neautorizată a informației.*

Disponibilitate: *permiterea entităților autorizate să acceseze în timp util și fiabil informația.*

Autentificare: *identifică o entitate sau atestă sursa datelor.*

Non-repudiere: *previne negarea unor evenimente anterioare.*

Securitate perfectă (Shannon 1949)

Definiție

O schemă de criptare peste un spațiu al mesajelor $\mathcal{M}$ este perfect sigură dacă pentru orice probabilitate de distribuție peste $\mathcal{M}$, pentru orice mesaj $m \in \mathcal{M}$ și orice text criptat c pentru care $Pr[C = c] > 0$, următoarea egalitate este îndeplinită:

$$Pr[M = m|C = c] = Pr[M = m]$$

- $Pr[M = m]$ - probabilitatea *a priori* ca Alice să aleagă mesajul m;
- $Pr[M = m|C = c]$ - probabilitatea *a posteriori* ca Alice să aleagă mesajul m, chiar dacă textul criptat c a fost văzut;
- securitate perfectă - dacă Oscar afla textul criptat nu are nici un fel de informație în plus decât dacă nu l-ar fi aflat.

Teoremă

Schema de criptare OTP este perfect sigură.

- securitatea perfectă nu este imposibilă dar...
- cheia trebuie să fie la fel de lungă precum mesajul
- incoveniente practice (stocare, transmitere)
- cheia trebuie să fie folosită o singură dată - one time pad - de ce?

Exercițiu Ce se întâmplă dacă folosim o aceeași cheie de două ori cu sistemul OTP ?

Limitările securității perfecte

Teoremă

Fie (Enc, Dec) o schemă de criptare perfect sigură peste un spatiu al mesajelor $\mathcal{M}$ și un spațiu al cheilor $\mathcal{K}$. Atunci $|\mathcal{K}| \geq |\mathcal{M}|$.

Sau altfel spus:

Teoremă

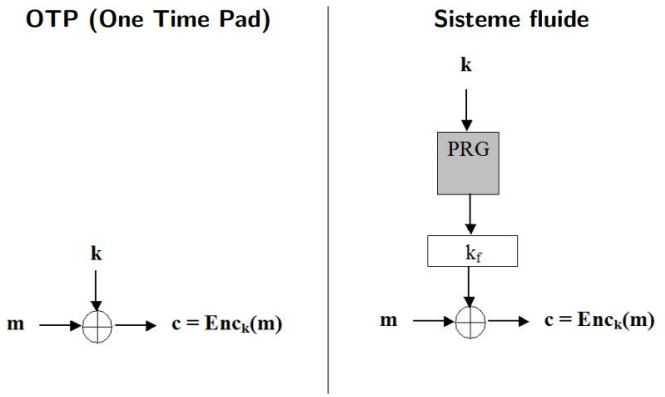
Nu există nici o schemă de criptare (Enc, Dec) perfect sigură în care mesajele au lungimea n biți iar cheile au lungimea (cel mult) $n - 1$ biți.

Neglijabil și ne-neglijabil

- Întrebare: de ce această definiție și nu alta?
  $\epsilon(n)$ negl. în $n \Leftrightarrow \forall p(n), \exists n_d$ 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 Polinomial în Timp)

Sisteme fluide



Sisteme fluide

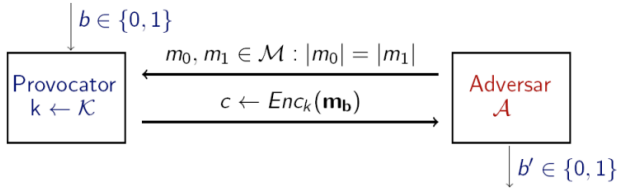
- Am văzut că securitatea perfectă există, dar nu este practic accesibilă - OTP;
- Facem un compromis de securitate, dar obținem o soluție utilizabilă în practică - sisteme de criptare fluide;
- Sistemele fluide sunt similare OTP, cu diferența că secvența **perfect aleatoare** de biți cu care se XOR-ează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

Securitate - interceptare unică

Teorema

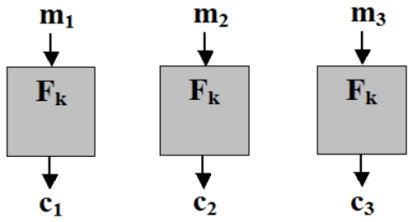
Dacă G este PRG, atunci sistemul fluid definit anterior este un sistem de criptare simetric de lungime fixă computațional sigur pentru un atacator pasiv care poate intercepta un mesaj.

Experimentul $Priv^{eav}_{\mathcal{A},\pi}(n)$

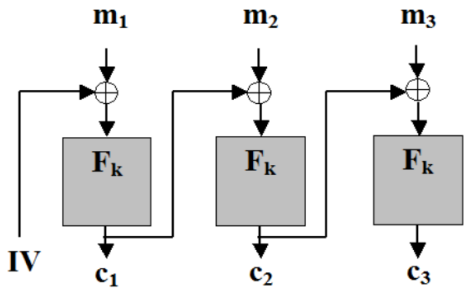


- Output-ul experimentului este 1 dacă $b' = b$ și 0 altfel. Dacă $Priv^{eav}_{\mathcal{A},\pi}(n) = 1$, spunem că $\mathcal{A}$ a efectuat experimentul cu succes.

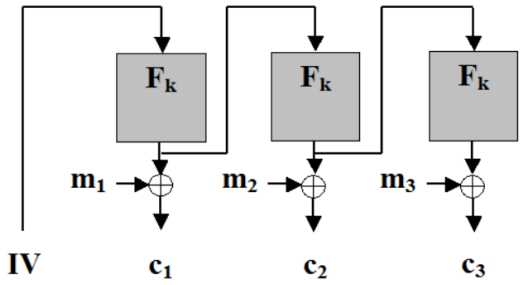
Modul ECB (Electronic Code Book)



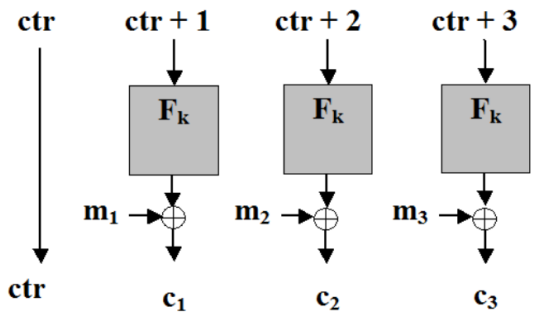
Modul CBC (Cipher Block Chaining)



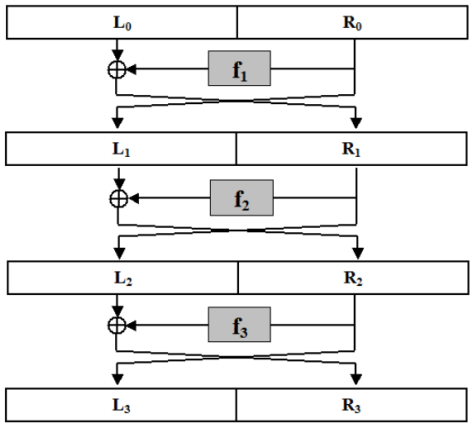
Modul OFB (Output FeedBack)



Modul CTR (Counter)



Rețele Feistel



Principul 2: Efectul de avalanșă

- Un singur bit modificat la intrare **trebuie** să afecteze toți biții din secvența de ieșire;
- Efectul de avalanșă apare într-o rețea de substituție-permutare dacă:
  1. S-box-urile sunt proiectate a.î. un singur bit schimbat la intrare să schimbe cel puțin 2 biți de la ieșire;
  2. Permutarea este proiectată a.î. biții de la ieșirea unui S-box să fie împărțiți între intrările în S-box-uri diferite la runda următoare.
- Principiul 2 - necesitate de securitate.

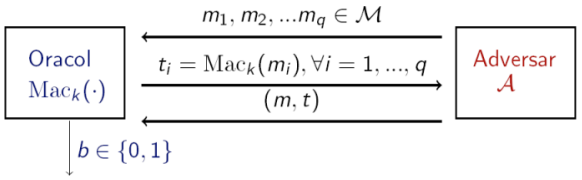
Noțiuni de securitate

- Definim astfel 2 noțiuni de securitate:
  - CPA (Chosen-Plaintext Attack): adversarul poate să obțină criptarea unor mesaje alese de el;
  - CCA (Chosen-Ciphertext Attack): adversarul poate să obțină criptarea unor mesaje alese de el și decriptarea unor texte criptate alese de el.

Securitate CPA

- Întrebare: Un sistem de criptare CPA-sigur este întotdeauna semantic sigur?
- Răspuns: DA! Experimentul $Priv^{eav}_{\mathcal{A},\pi}(n)$ este $Priv^{cpa}_{\mathcal{A},\pi}(n)$ în care $\mathcal{A}$ nu folosește oracolul de criptare.
- Întrebare: Un sistem de criptare determinist poate fi CPA-sigur?
- Răspuns: NU! Adversarul cere oracolului criptarea mesajului m_0. Dacă textul criptat este egal cu c, atunci $b' = 0$, altfel $b' = 1$. În concluzie, $\mathcal{A}$ câștigă cu probabilitate 1.

Experimentul $\mathrm{Mac}^{forge}_{\mathcal{A},\pi}(n)$



- Output-ul experimentului este 1 dacă și numai dacă: (1) $\mathrm{Vrfy}_k(m, t) = 1$ si (2) $m \notin \{m_1, ..., m_q\}$;
- Dacă $\mathrm{Mac}^{forge}_{\mathcal{A},\pi}(n) = 1$, spunem că $\mathcal{A}$ a efectuat experimentul cu succes.

Securitate CCA

- Întrebare: Un sistem de criptare CCA-sigur este întotdeauna CPA-sigur?
- Răspuns: DA! Experimentul $Priv^{cpa}_{\mathcal{A},\pi}(n)$ este $Priv^{cca}_{\mathcal{A},\pi}(n)$ în care $\mathcal{A}$ nu folosește oracolul de decriptare.
- Întrebare: Un sistem de criptare determinist poate fi CCA-sigur?
- Răspuns: NU! Sistemul nu este CPA-sigur, deci nu poate fi CCA-sigur.

CBC-MAC

Definiție

Fie F o funcție pseudoaleatoare. Un CBC-MAC este format dintr-o pereche de algoritmi polinomiali probabiliști $(\mathrm{Mac}, \mathrm{Vrfy})$*:*

1. Mac*: pentru o cheie $k \in \{0,1\}^n$ și un mesaj m de lungime l:*
   - *Sparge m in $m = m_1, ..., m_l$, $|m_i| = n$ și notează $t_0 = 0^n$;*
   - *Pentru $i = 1, ..., l$, calculează $t_i = F_k(t_{i-1} \oplus m_i)$;*

   Intoarce t_l ca tag-ul rezultat;
2. Vrfy : *pentru o cheie $k \in \{0,1\}^n$, un mesaj m de lungime l, și un tag t de lungime n:*
   întoarce 1 dacă și numai dacă $t = \mathrm{Mac}_k(m)$.

Rămâne valabilă condiția de corectitudine:
$\forall m \in \mathcal{M}$, $k \in \mathcal{K}$, $\mathrm{Vrfy}_k(m, \mathrm{Mac}_k(m)) = 1$.

Securitatea funcțiilor hash

- În practică, rezistența la coliziuni poate fi dificil de obținut;
- Pentru anumite aplicații sunt utile noțiuni mai relaxate de securitate;
- Există 3 nivele de securitate:
  1. Rezistența la coliziuni: este cea mai puternică noțiune de securitate și deja am definit-o formal;
  2. Rezistența la a doua preimagine: presupune că fiind dat x este dificil de determinat $x' \neq x$ a.î. $H(x) = H(x')$
  3. Rezistența la prima preimagine: presupune că fiind dat $H(x)$ este imposibil de determinat x.

Important de reținut!

- În criptografia cu cheie publică:
  - NU există securitate perfectă
  - securitate semantică = securitate CPA

Atacul "zilei de naștere"

- Generalizând, considerăm o mulțime de dimensiune n și q elemente uniform aleatoare din această mulțime $y_1, \ldots, y_q$;
- Atunci pentru $q \geq 1.2 \times 2^{n/2}$ probabilitatea să existe $i \neq j$ a.î. $y_i = y_j$ este $\geq 1/2$.
- Aceast rezultat conduce imediat la un atac asupra funcțiilor hash cu scopul de a determina coliziuni:
  - Adversarul alege $2^{n/2}$ valori x_i;
  - Calculează pentru fiecare $y_i = H(x_i)$;
  - Caută $i \neq j$ cu $H(x_i) = H(x_j)$;
  - Dacă nu găsește nici o coliziune, reia atacul.
- Cum probabilitatea de succes a atacului este $\geq 1/2$, atunci numărul de încercări este ≈ 2.

Padded RSA

1. Se rulează GenRSA pentru a determina N, e, d.
   - Cheia publică este: (N, e);
   - Cheia privată este (N, d);
2. **Enc**: dată o cheie publică (N, e) și un mesaj $m \in \{0,1\}^{l(n)}$, alege $r \leftarrow^R \{0,1\}^{|N|-l(n)-1}$, interpretează $r||m$ ca un element în $\mathbb{Z}_N$ și întoarce $c = (r||m)^e \mod N$;
3. **Dec**: dată o cheie secretă (N, d) și un mesaj criptat $c \in \mathbb{Z}_N$, calculează $c^d \mod N$ și întoarce ultimii $l(n)$ biți.

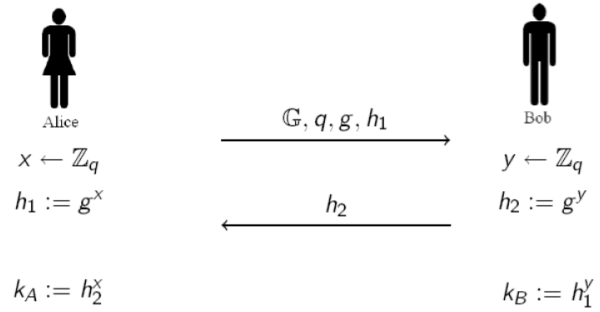
Padded RSA

- Pentru $l(n)$ foarte mare, atunci este posibil un atac prin forță brută care verifică toate valorile posibile pentru r;
- Pentru $l(n)$ mic se obține securitate CPA:

Teoremă

Dacă problema RSA este dificilă, atunci Padded RSA cu $l(n) = O(\log n)$ este CPA-sigură.

Schimbul de chei Diffie-Hellman



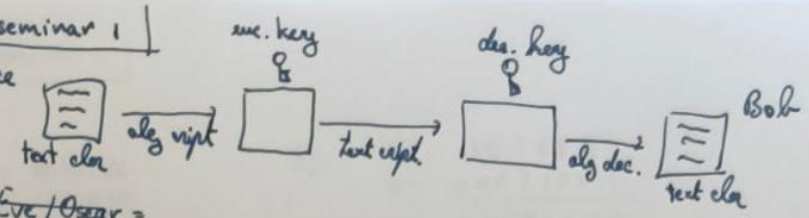
Securitate - Problema 3

Problema 3: Proprietatea de homomorfism

- Fie m_1, m_2 2 texte clare și $c_1 = (c_{11}, c_{12}), c_2 = (c_{21}, c_{22})$ textele criptate corespunzătoare;
- Atunci:
  $c_1 \cdot c_2 = (c_{11} \cdot c_{21}, c_{12} \cdot c_{22}) = (g^{y_1} \cdot g^{y_2}, m_1 h^{y_1} \cdot m_2 h^{y_2})$
- Întrebare: Dacă un adversar cunoaște c_1 și c_2 criptările lui m_1, respectiv m_2, ce poate spune despre $c_1 \cdot c_2$?
- Răspuns: $c_1 \cdot c_2$ este criptarea lui $m_1 \cdot m_2$ folosind $y = y_1 + y_2$:
  $c_1 \cdot c_2 = (g^{y_1+y_2}, m_1 m_2 h^{y_1+y_2})$
- Un sistem de criptare care satisface
  $Dec_s k(c_1 \cdot c_2) = Dec_{sk}(c_1) \cdot Dec_{sk}(c_2)$ se numește sistem de criptare **homomorfic**.
  (homomorfismul este deseori o proprietate utilă în criptografie)

SSI - seminar 1

1) a)



b) ~~Eve / Oscar~~ =
Eve = ascultă mesajele (eavesdropper)
Oscar = oponent (nu neapărat rău intenționat)

c) Alg., cu cheia trebuie asigurată

d) Scopul lor este să afle cheia / descifrarea mesajul

3) cheie de cript. pe 128 biți 256 biți
chei exist. 2^{128} 2^{256}
timp, 2^{10} dec. pe sec (Brute) 2^{128} s 2^{256} s

4) Sistemul cavalerilor din Malta

A:	B:	C:
D:	E:	F:
G:	H:	I:

J:	K:	L:
M:	N:	O.
P.	Q.	R.

S	T	U
V	W	X
Y	Z	

5) Sistemul Polybius (I = J)

A	B	C	D	E
F	G	H	I/J	K
L	M	N	O	P
Q	R	S	T	U
V	W	X	Y	Z

SECRET ⇒ 43 15 13 ...

$|K| = \frac{26 \cdot 25}{2}$

8) Sist. de substit. simplă
K = BROWSER →
A B C D E F G ...
B R O W S E R A C D ...

9) ~~Sist. de transp.~~

11) Playfair
m = The circle
k = album

A	L	B	U	M
C	D	E	F	G
H	I/J	K	N	O
P	Q	R	S	T
V	W	X	Y	Z

TH EC IR CL EX
PO FD KQ DA KB

m = TAAK SUCP
k = ATAC

ATCBA
EFGHI
KLMNO
PQRSU
VWXYZ

TA AK SU CP
AOVE RS AR

SSI – seminar 2

1) OTP

a) $m = 00101101$
$k = 10110110$
$\overline{10011011}$

b) $c = 11010101$
$k = 00011001$
$\overline{11001100}$

2) Eve intere. 0x A617 , OTP
$m_1 =$ 0x 4441 sau $m_2 =$ 0x 4E55
$\overline{\quad\quad}$
$m = m_1$ sau $m = m_2$?

R: nu poate det.

3) a) sist corect : $c = m \oplus k \Leftrightarrow c \oplus k = m \oplus k \oplus k \Leftrightarrow c \oplus k = m$

b) XOR și nu AND, OR, NOT ? → nu ar mai fi simetric

4) OTP dar fără $k = 0^l$ nu e perfect sigur.
Contrazice teorema C5 : $|K| \geq |M|$

5) $1Gb = 2^{10} Mb = 2^{20} kb = 2^{30} B = 2^{33}$ biți

SSI - seminar 3

1) $G: \{0,1\}^k \to \{0,1\}^k$, $k<n$ definit mai jos. Este G PRG?

a) $msb(G(s)) = 1$, $\forall s$

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

$D(r) = \begin{cases} 1, & msb(r)=1 \\ 0, & msb(r)=0 \end{cases}$

$Pr[D(r)=1] = \frac{1}{2}$
$Pr[D(G(s))=1] = Pr[1=1] = 1$ $\Big| \Rightarrow \frac{1}{2} \leq negl\ n$ Exclus

b) $msb(G(s)) = 1$ cu prob $\frac{1}{n^{100}}$

$D(r) = msb(r)$
$Pr[D(r)=1] = \frac{1}{2}$
$Pr[D(G(s))=1] = \frac{1}{n^{100}}$ $\Big| \Rightarrow \frac{1}{2} - \frac{1}{n^{100}} \leq negl\ n$ Exclus

c) $G(s) = G_0(s) \| G_1(s) \| G_2(s)$, $|G_0(s)| = |G_1(s)| = |G_2(s_0)|$, $G_2(s) = G_1(s) \oplus G_0(s)$

$r = r_0 \| r_1 \| r_2$
$s = s_0 \| s_1 \| s_2$

$D(r) = \begin{cases} 1, & r_0 = r_1 \oplus r_2 \\ 0, & altfel \end{cases}$

$Pr[D(r)=1] = Pr[r_0 = r_1 \oplus r_2] = \frac{1}{2^{|r_0|}}$ $\Big| \Rightarrow$ Exclus

$Pr[D(G(s))=1] = Pr[G_0(s) = G_1(s) \oplus G_2(s)] = 1$

2) $\hat{G}$ PRG $\Rightarrow \hat{G}'(s) = \hat{G}(s_{n/2} \dots s_n)$ PRG, unde $s = s_1 \dots s_n$

Fie G PRG, $G'(s) = G(s0^{|s|})$ PRG?

$G'(s) = G(s0^{|s|})$ PRG $\Rightarrow G''(s) = G(0^{|s|})$ PRG Exclus

3) $F' : \{0,1\}^k \times \{0,1\}^k \to \{0,1\}^m$ PRF. Este F PRF?

a) $F_k(x) = \begin{cases} F'_k(x) & , x \text{ par} \\ F'_k(x+1) & , x \text{ impar} \end{cases}$

$\Delta(r) = \begin{cases} 1, & r(1) = r(2) \\ 0, & \text{altfel} \end{cases}$

$Pr[\Delta(r) = 1] = Pr[r(1) = r(2)] = \frac{1}{2^m}$

$Pr[\Delta(C_k(F_k)) = 1] = Pr[F_k(1) = F_k(2)] = 1$ | Exista un Δ negl.

b)

4) Fie $F' : K \times X \to \{0,1\}^{128}$ PRF. Este F PRF?

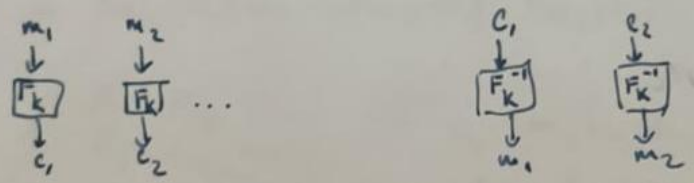
$F_k(x) = \begin{cases} 0^{128} & , x = 0 \\ F'_k(x) & , x \neq 0 \end{cases}$

$\Delta(r) = \begin{cases} 1, & r F_k(0) = 0 \\ 0, & \text{altfel} \end{cases}$

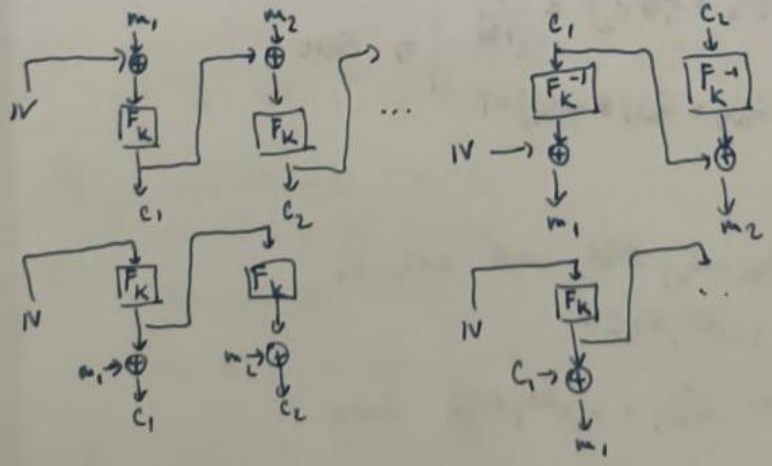
$Pr[\Delta(r) = 1] = Pr[r(0) = 0] = \frac{1}{2^{128}}$

$Pr[\Delta(F_k(\cdot)) = 1] = Pr[F_k(0) = 0] = 1$ | un Δ negl.

5) ECB:



CBC:



OFB
OBE:

SSI - seminar 3 - cont

7) - DES 56 biți
- cheia gen. pe baza unei parole de 8 caractere char (0-255) : 8·8 = 64 biți, din care 8 ne ignoră (biții)
- cale fortează - 10^6 chei pe s.

a) spațiul cheilor? : $128^8 = 2^{7 \cdot 8} = 2^{56}$
timp căutare exhaustivă? $\frac{2^{56}}{10^6} \simeq 2^{36}$ secunde

b) spațiul cheilor (0-127) : $64^8 = 2^{6 \cdot 8} = 2^{48}$
$\frac{2^{48}}{10^6} \simeq 2^{28}$ secunde

c) litere mari | spațiul cheilor : 13^8
$\frac{13^8}{10^6}$ secunde

8) AES : intrare runda

$$\begin{bmatrix} 04 & 07 & E2 & 49 \\ F2 & 78 & 2F & C5 \\ CA & 28 & 01 & A7 \\ 97 & 45 & 96 & 10 \end{bmatrix}$$

cheia de runda

$$\begin{bmatrix} 21 & 35 & AC & 6C \\ 75 & 50 & A1 & 1B \\ 17 & 62 & 6B & F0 \\ 87 & 0B & 3C & 9B \end{bmatrix}$$

lung. cheie	128	192	256
runde	10	12	14

1. XOR cu cheia de runda
2. SubBytes (conform Sbox)
3. ShiftRows, MixColumns

SSI - seminar 5

1) CBC, IV incrementat cu 1 de fiecare dată - când cript un mesaj:
→ Schema nu e CPA sigură.

Adv. știe IV la fiecare pas (după primul)

1. - aflu IV
2. $m = IV+1 \Rightarrow c = F_k(m \oplus (iv+1)) = F_k(0)$
3. $m_1 = IV+2,\ m_2 = 0 \Rightarrow$

2) $Enc'_k(m_1 \| m_2) = (Enc_k(m_1), Enc_k(m_2))$ nu e CCA sigur

1. $M_0 = m_{0,0} \| m_{0,1}$, $M_1 = m_{1,0} \| m_{1,1}$ $\overset{cript}{\Rightarrow}$ $c_b = (Enc_k(m_{b,0}), Enc_k(m_{b,1}))$

2. $(Enc_k(m_{b,0}), Enc_k(m_{b,0})) \overset{dec.}{\Rightarrow} m_{0,0} \| m_{0,0}$ sau $m_{1,0} \| m_{1,1} \Rightarrow b$

3) $m_1 = m \oplus IV \rightarrow c_1$
$m_2 = c_1 \oplus m' \rightarrow c_2$
~~$(m_1, m_2) \rightarrow$~~
$(m \oplus c_2, m' \oplus c_2) = c \quad c = c_1 / c_2$

4) $M = \{0,1\}^n,\ T = \{0,1\}^{128}$
$Mac'(k,m) = Mac(k,m)$
$Vrfy'(k,m,t) = \begin{cases} Vrfy(k,m,t), & m \neq 0^n \\ 1, & \text{altfel} \end{cases}$

→ orice t întoarce 1

5) F PRF. $Mac(m_0 \| m_1, k) = F_k(0 \| m_0) \| F_k(1 \| m_1)$, $|m_0| = |m_1| = n-1$, $k \in \{0,1\}^n$

Nu: $Mac(m_0 \| m_0, k) = F_k(0 \| m_0) \| F_k(1 \| m_0)$
~~$\Rightarrow Mac(m$~~
$Mac(m_1 \| m_1, k) = F_k(0 \| m_1) \| F_k(1 \| m_1)$
$\Rightarrow Mac(m_0 \| m_1, k)$