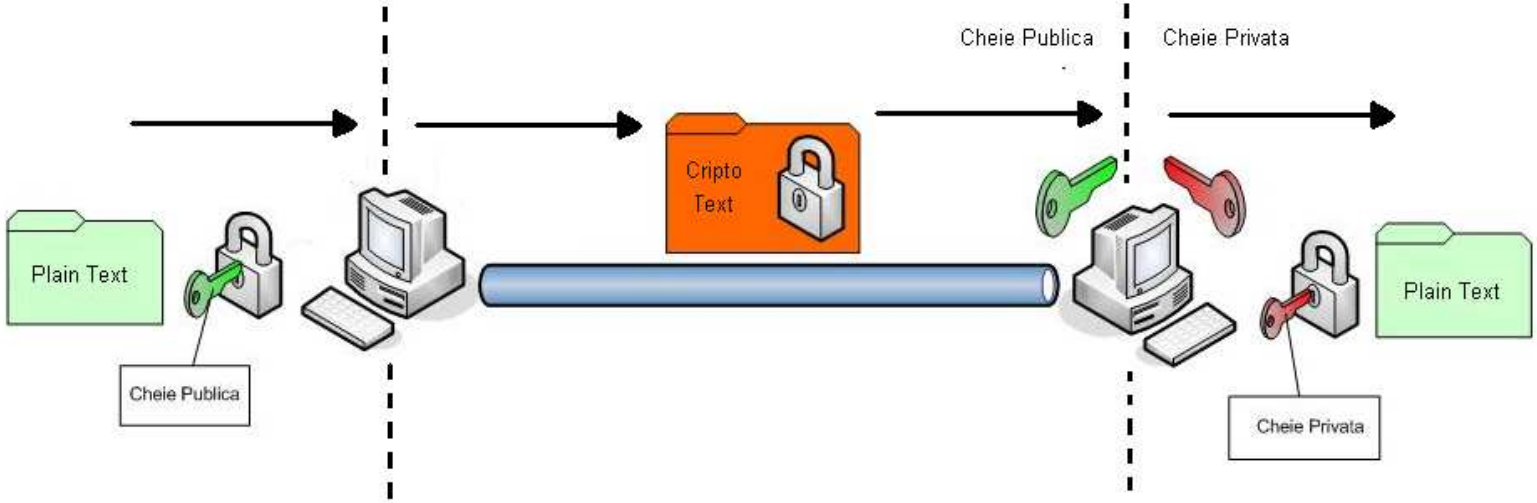
UNIUNEA EUROPEANĂ
GUVERNUL ROMÂNIEI MINISTERUL MUNCII, FAMILIEI ŞI PROTECŢIEI SOCIALE AMPOSDRU
Fondul Social European POSDRU 2007-2013
Instrumente Structurale 2007-2013
ORGANISMUL INTERMEDIAR REGIONAL PENTRU POS DRU REGIUNEA BUCUREŞTI ILFOV

Proiect cofinanțat din Fondul Social European prin
Programul Operațional Sectorial Dezvoltarea Resurselor Umane 2007-2013.
"Investește în oameni!"

Cap. 2.2. Functii Criptografice Asimetrice. Functii de criptarea cu cheie publica si semnaturi digitale (RSA, Diffie-Hellman-Merkle, ElGamal, DSA, ECC).

Proiect cofinanțat din Fondul Social European prin Programul Operațional Sectorial Dezvoltarea Resurselor Umane 2007-2013.
"Investește în oameni!"

Criptari asimetrice



Descrierea criptosistemului RSA

- **Generarea cheii**

1. Genereaza doua numere prime p,q
2. Calculeaza $n=pq,\ \Phi(n)=(p-1)(q-1)$
3. Genereaza e relativ prim la $\Phi(n)$
4. Calculeaza d $a.i.$ $ed \equiv 1 mod\ \Phi(n)$
5. Cheia Publica este (n,e) si Cheia Privata (n,d)

- **Criptarea**

1. Obtine cheia publica (e,n)
2. Calculeaza $c=m^e mod n$, m este mesajul iar c este mesajul criptat

- **Decriptarea**

1. Receptioneaza mesajul criptat c
2. Calculeaza $m=c^d mod n$

Proiect cofinanțat din Fondul Social European prin Programul Operațional Sectorial Dezvoltarea Resurselor Umane 2007-2013. "Investește în oameni!"

- Observatie: cel mai costisitor pas in implementarea RSA este generarea celor doua numere prime p si q, generarea cheii se face insa doar o data urmand ca aceeasi cheie sa fie utilizata pentru oricate criptari/decriptari. Generarea celor doua numere prime se face in mod eficient prin generarea unor numere aleatoare si aplicarea unor teste de primalitate asupra acestora (testele de primalitate sunt usor de efectuat si sunt probabilistice, deci se aplica pana cand se stie cu o probabilitate suficient de mare ca numarul este prim)

Exemplu RSA

- ***Exemplul 1:*** (numere artificiale, in practica se folosesc numere de sute-mii de biti)
- Generarea cheii $p = 11,\ q = 13,\ n = p \cdot q = 143, \phi(n) = (p-1)\cdot(q-1) = 120$

  $e = 7,\ d = 103,\ e \cdot d \equiv 1 \bmod 120$

  $Cheia\ Publica(7,143)$

  $Cheia\ Privata(103,143)$
- Criptarea $m = 5$

  $c = m^e \bmod n = 5^7 \bmod 143 = 47$
- Decriptarea $c = 47$

  $m = c^d \bmod n = 47^{103} \bmod 143 = 5$

Exemplul 2. RSA Challenge 100.000 $ oferiti de RSA (premiu retras in 2007)

RSA-1024 =
135066410865995223349603216278805969938881475
605667027524485143851526510604859533833940287
150571909441798207282164471551373680419703964
191743046496589274256239341020864383202110372
958725762358509643110564073501508187510676594
629205563685529475213500852879416377328533906
109750544334999811150056977236890927563

Analiza Securitatii

- ***Relatia intre RSA si Factorizarea Intregilor:***
- Nu exista nici o demonstratie ca problema RSA este echivalenta cu factorizarea
- Factorizarea lui n duce la "spargerea" sistemului
- Calculul (sau aflarea) unei perechi de exponenti public-privat este echivalent factorizarii. Observatie:

$$a^{e\cdot d-1} \equiv 1 \bmod n, \forall\, a \in Z_n^*, e\cdot d - 1 = 2^s \cdot t$$

$$daca \exists i\ a.i. a^{2^{i-1}\cdot t} \neq 1 \bmod n\ si\ a^{2^i \cdot t} \equiv 1 \bmod n$$

$$\Rightarrow cmmdc\left(a^{2^{i-1}\cdot t} - 1, n\right) este\ p\ sau\ q$$

- ***Forward search*** – daca mesajul este relativ mic atunci el poate fi subiectul unei cautari exhaustive
- ***Exponenti mici de decriptare*** – pot fi si ei subiectul unei cautari exhaustive
- ***Mesaje necriptate*** – exista mesaje care nu pot fi criptate (Ex: 0,1) numarul lor este redus si nu afecteaza securitatea, numarul de mesaje care au valoarea criptotextului egala cu a plain-textului este exact $(\gcd(\varepsilon-1, p-1)+1)(\gcd(\varepsilon-1, q-1)+1)$
- ***Utilizarea modulelor comune:*** a fost sugerat ca mai multe entitati din acelasi sistem sa foloseasca un modul comun acest lucru duce la pierderea totala a securitatii deoarece fiecare entitate poate calcula cheia celeilalte entitati

Schimbul de cheie Diffie-Hellman-(Merkle)

- Schimbul de cheie Diffie-Hellman:

$$A \rightarrow B : a^{\alpha} \bmod p$$

$$B \rightarrow A : a^{\beta} \bmod p$$

$$cheie \quad comuna : a^{\alpha\beta} \bmod p$$

- A il cunoaste α pe iar B il cunoaste pe β deci dupa acest schimb A si B pot calcula in particular $a^{\alpha\beta} \, mod \, p$ ca fiind cheia comuna (pentru ca $(a^{\alpha} mod p)^{\beta} \, mod \, p = (a^{\beta} mod p)^{\alpha} \, mod \, p = a^{\alpha\beta} \, mod \, p$)
- Deoarece logaritmul discret nu poate fi calculat un adversar nu poate sa il afle pe α sau β deci nu poate efectua calculul facut de A si B pentru a obtine cheia
- Atentie: schimbul de cheie Diffie-Helman este un schimb de cheie neautentificat si poate fi fraudat cu un ***atac de tip man-in-the-middle*** (lucru valabil si pentru orice alt schimb de cheie fara autentificare din partea participantilor)

Criptarea asimetrica ElGamal [ElGamal, 1985]

- ***Generarea cheilor:***

1) Genereaza un numar prim p
2) Alege un generator a al grupului Z_p
3) Genereaza un intreg aleator α
4) Calculeaza $a^{\alpha} \bmod p$
5) Cheia publica este $a, a^{\alpha} \bmod p, p$ iar cea privata este α

- ***Criptarea unui mesaj:***

1) Obtine cheia publica a entitatii $a, a^{\alpha} \bmod p, p$
2) Reprezinta mesajul ca intreg in intervalul (1,p)
3) Genereaza un intreg aleator $1<k<p-2$
4) Calculeaza $\gamma = a^k \bmod p$
5) Calculeaza $\delta = m(a^{\alpha})^k \bmod p$
6) Trimite catre posesorul cheii private γ, δ

- ***Decriptarea unui mesaj:***

1) Decripteaza mesajul ca $m = \delta\gamma^{-\alpha} \bmod p$

- Rezumat al criptarii ElGamal

$$A \to B : a, a^{\alpha} \bmod p$$

$$B \to A : \delta = m \cdot \left(a^{\alpha}\right)^{k} \bmod p, \gamma = a^{k} \bmod p$$

  - A decripteaza mesajul:

$$m = \delta \cdot \left(\gamma^{-1}\right)^{\alpha} \bmod p$$

- ***Observatie:*** numarul k este ales aleator de B la fiecare criptare, altfel criptosistemul poate fi spart

Semnaturi digitale

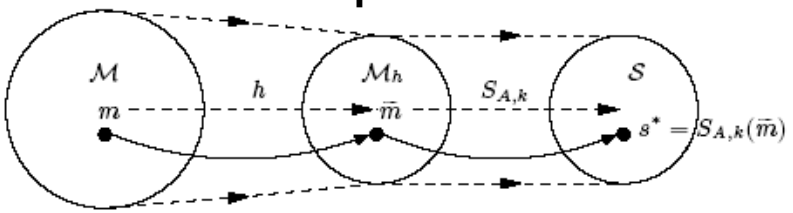
- Sunt echivalentul electronic al semnaturilor de mana, se folosesc pentru a asigura non-repudierea informatiei (impiedica o entitate in a nega ulterior ca transmis o anume informatie)
- **Principiu**: inverseaza rolul cheii publice si private ale unui sistem criptografic asimetric si foloseste cheia privata pentru a semna mesaje iar cheia publica pentru a verifica semnatura
- ***Definitie:*** *O schema (algoritm) pentru semnatura digitala este un set de 3 elemente {Gen, Sig, Ver} care satisface urmatoarele conditii: 1. Gen este algoritmul de generare a cheilor care primeste ca intrare un parametru de securitate k si returneaza o pereche (Pb, Pv) de chei publica, privata 2. Sig este algoritmul de semnare digitala care primeste ca intrare cheia privata Pv si mesajul m si returneaza semnatura 3. Ver – este algoritmul de verificare care primeste ca intrare cheia publica Pb, semnatura si mesajul (uneori nu e necesar si mesajul) si returneaza 1/0 dupa cum semnatura este adevarata sau falsa 4. pentru orice pereche de chei publica/privata si orice mesaj este adevarat ca Ver(Pb, m, Sig(Pv,m))=1*

Clasificare

1) semnaturi digitale cu apendice (frecvent utilizate in practica) – sunt semnaturi digitale pentru a caror verificare este necesara si prezenta mesajului original care a fost semnat

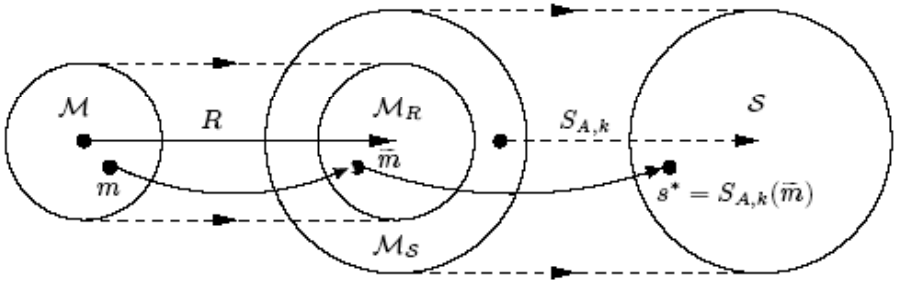
2) semnaturi digitale care permit recuperarea mesajului (fara apendice) – sunt semnaturi digitale care nu necesita prezenta mesajului original pentru a fi verificate, mai mult decat atat ele permit recuperarea mesajului din semnatura (este usor de observat ca orice semnatura digitala care face posibila recuperarea mesajului poate fi transformata intr-o semnatura digitala cu apendice)

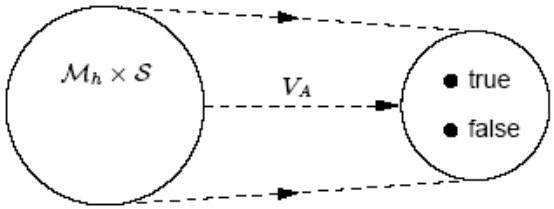
Cu apendice



Semnarea unui mesaj

Cu recuperarea mesajului





Verificarea unei semnaturi

figura din [Menezes et. al, 1996, p. 429, p. 431]

- In cazul semnaturii fara apendice R este o functie de redundanta, la cea cu apendice h este *o functie hash*
- ***Observatie:*** Orice semnatura fara apendice poate fi transformata in semnatura cu appendice

Semnatura RSA (cu recuperarea mesajului)

- Genereaza o pereche de chei RSA
- Semnarea digitala:

1) Calculeaza $\overline{m} = R(m)$
2) Calculeaza $s = \overline{m}^{\delta} \bmod n$
3) Semnatura digitala a mesajului *m* este *s*

- Verificarea semnaturii:

1) Obtine cheia publica a entitatii n, ε
2) Calculeaza $\overline{m} = s^{\varepsilon} \bmod n$
3) Verifica ca mesajul contine redundanta $\overline{m} \in M_R$
4) Recupereaza mesajul ca $m = R^{-1}(\overline{m})$

$Exemplu\ (\ fara\ functie\ de\ redundanta)$

$n = 143, e = 7, d = 103$

$s = 47\ este\ semnatura\ digitala\ \ a\ \ lui\ m = 5$

$m = s^d \bmod n = 47^{103} \bmod 143 = 5$

Semnatura ElGamal (cu apendice)

- **Generarea cheilor**:

1) Genereaza un numar prim p
2) Genereaza un generator α al grupului Z_p
3) Genereaza un intreg aleator $1=<a=<p-2$
4) Calculeaza $\alpha^a \bmod p$
5) Cheia publica este α ,$\alpha^a \bmod p$, p iar cea privata este a

- **Semnarea digitala:**

1) Genereaza un intreg aleator $1=<k=<p-2$ $cu\ cmmdc(k,p-1)=1$
2) Calculeaza $r = \alpha^k \bmod p$
3) Calculeaza $k^{-1} mod(p-1)$
4) Calculeaza $s=k^{-1}\{h(m)-ar\} mod(p-1)$
5) Semnatura digitala asupra lui m este perechea (r, s)

- **Verificarea semnaturii** digitale implica urmatorii pasi:

1) Obtine cheia publica α ,$\alpha^a \bmod p$, p
2) Verifica ca $1=<r=<p-1$ si in caz contrar respinge semnatura
3) Calculeaza $v_1=y^r r^s \bmod p$
4) Calculeaza $h(m)$, $v_2=\alpha^{h(m)} \bmod p$
5) Accepta semnatura daca si numai daca $v_1 = v_2$

Nivele de fraudare pentru semnaturi digitale

1) ***Falsificare existentiala*** (existential forgery) – un adversar poate falsifica cel putin o semnatura dar nu are control total (sau deloc) asupra mesajului semnat
2) ***Falsificare selectiva*** (selective forgery) – un adversar poate falsifica o semnatura asupra unui anumit tip de mesaje la alegerea sa
3) ***Falsificare universala*** (universal forgery) – adversarul poate calcula semnaturi digitale asupra oricarui mesaj cu toate ca nu cunoaste cheia cu care mesajele sunt semnate
4) ***Spargere totala*** (total break) - adversarul este in posesia cheii private si poate falsifica orice semnatura

Tipuri de atac asupra semnaturilor digitale

1) ***key-only*** – este atacul in care adversarul are acces doar la cheia publica, de verificare, a semnaturii digitale

2) ***message attacks*** – sunt atacuri in care atacatorul are acces la mesaje semnate de entitatea posesoare a cheii private, se clasifica dupa cum urmeaza:

2.1) ***known-message*** (sau known signature) – adversarul are acces la mesaje semnate de posesorul cheii private, dar aceste mesaje nu sunt la alegerea lui

2.2) ***chosen-message*** – adversarul poate obtine semnatura pentru un anumit numar de mesaje la alegerea sa

2.3) ***adaptive chosen-message*** – adversarul obtine semnaturi digitale la alegerea sa pentru un anume set de mesaje iar acest lucru se desfasoara interactiv (posesorul cheii private nu vrea sa semneze mesajul pentru care atacatorul doreste sa obtina semnatura dar este dispus sa semneze orice alte mesaje pe baza carora atacatorul reuseste sa construiasca semnatura)

Notiuni slabe de securitate

- Schemele asimetrice anterior prezentate au doar securitate de tipul:
- ***Confidentialitate totul-sau-nimic (All-or-nothing secrecy):*** pastreaza confidentialitatea doar in masura in care mesajul nu putea fi recuperat integral si nici cheia nu putea fi sparta de catre un adversar
- ***Securitate in fata adversarilor pasivi:*** pastreaza securitatea doar in fata adversarilor pasivi care nu are acces la masina de criptare/decriptare (de exemplu, exercitiile 5 de la L1 si 7 de la L2 arata cum un adversar activ poate frauda RSA si ElGamal)

Insuficienta notiunilor slabe de securitate

- In lumea reala, un adversar nu are nevoie de a recupera tot mesajul pentru a frauda un mecanism de securitate (poate fi suficient si sa recupereze un singur bit din informatie) iar adversari pasivi nu exista (adversarul are acces cel putin la masina de criptare a unui sistem asimetric)
- Astfel un sistem criptografic asimetric trebuie sa fie rezistent in fata urmatoarelor atacuri:
  - Plaintext ales (chosen plaintext attack) **CPA**
  - Criptotext ales (chosen ciphertext) **CCA**
  - Criptotext ales adaptiv (adaptive chosen ciphertext) **CCA2** (cel mai important atac)

Notiuni puternice de securitate

- ***Securitate polinomiala (SP sau mai comun IND)****:* o schema asimetrica are securitate polinomiala daca nici un adversar nu poate selecta doua mesaje m_1 si m_2 astfel incat avand ulterior acces la masina de criptare sa poata distinge intre c_1 si c_2 (acestea fiind criptotextele corespunzatoare) cu probabilitate mai mare de ½
- ***Securitate semantica (SS)****:* o schema asimetrica are securitate semantica daca ceea ce se poate calcula eficient despre plaintext avand criptotextul se poate calcula eficient si fara criptotext – adica criptotextul nu spune nimic despre mesaj
- ***Criptare non-maleabila (NM)****:* o schema asimetrica este non-maleabila daca avand un criptotext este imposibil de generat un alt criptotext astfel incat valorile plaintextelor aferente sa aiba vreo legatura cunoscuta de adversar
- ***Criptare plaintext-aware(PA)****:* o schema asimetrica este plaintext-aware daca un adversar nu poate construi criptotextul unui plaintext pe care nu il cunoaste
- Relatii:

1) O schema asimetrica are securitate polinomiala daca si numai daca are securitate semantica (SP<=>SS)
2) O schema non-maleabila are si securitate polinomiala (NM=>SS)
3) O schema plaintext-aware si cu securitate semantica este non-maleabila (NM = PAW + SS) si sigura in fata atacurilor de tip CCA2

O defintie completa a proprietatilor de securitate pentru scheme asimetrice

- Coreland proprietatile IND si NM cu tipurile de atac CPA, CCA si CCA2 se obtin urmatoarele tipuri de rezistenta pentru o schema asimetrica

1) IND-CPA
2) IND-CCA
3) **IND-CCA2**

1) NM-CPA
2) NM-CCA
3) **NM-CCA2**

RSA - Optimal Asymmetric Encription (RSA-OAEP) [Bellare & Rogaway, 1995]

- Prin OAEP se doreste:

1) Transformarea criptarii asimetrice (in particular RSA) in criptare non-determinista prin adaugarea unui elemnt aleator
2) Prevenirea aflarii unui bit de informatie fara a inversa toata functia one-way

Proiect cofinanțat din Fondul Social European prin Programul Operațional Sectorial Dezvoltarea Resurselor Umane 2007-2013. "Investește în oameni!"

- OAEP se poate aplica asupra RSA si prin RSA-OAEP se obtine o schema rezistenta IND-CCA2, NM-CCA2
- Transformarea OAEP avand o functie de criptare asimetrica E (in cazul nostru E este chiar functia RSA) se defineste ca:

$$E(m) = f\left(\left\{m0^{k_1} \oplus G(r)\right\} \| \left\{r \oplus H\left(m0^{k_1} \oplus G(r)\right)\right\}\right)$$

$$G:\{0,1\}^{k_0} \to \{0,1\}^{n+k_1}, H:\{0,1\}^{n+k_1} \to \{0,1\}^{k_0}$$

- G si H sunt functii aleatoare (in practica ambele pot fi derivate dintr-o functie hash, de exemplu SHA-x, in acest caz functiile devin pseudoaleatoare, dar proprietatile de securitate par sa fie conservate)
- f este o functie pe k biti, iar lungimea plaintextului m este $n=k-k_0-k_1$

OAEP

figura din [Menezes et. al, 1996, p. 312]