RSA.

algoritm de criptare pe blocuri

Generarea cheilor:

1. Se selectează două numere întregi prime p şi q.

2. Se calculează produsul n=p*q.

3. Se calculează indicatorul lui Euler Φ(n)=(p-1)*(q-1).

4. Se selectează un număr întreg e astfel încât

c.m.m.d.c.(Φ(n),e)=1, 1<e< Φ(n).

5. Se calculează d astfel încât d = e^{-1} mod Φ(n).

6. Cheia publică este (e,n), iar cheia privată este d.

Algoritmul de criptare:

• Presupunem că un utilizator A are cheia publică (e,n) şi cheia privată d.

• Utilizatorul B cripteaz ă mesajul M pentru a fi transmis la A astfel:

1. Obține cheia publică (e,n) a lui A.

2. Transformă mesajul ce va fi criptat într-un număr întreg M în intervalul [0, n-1].

3. Calculează C = M^e (mod n).

4. Trimite textul cifrat C la utilizatorul A.

Algoritmul de decriptare:

• Pentru a determina textul clar M din textul cifrat C, utiliz. A calc.:

M = C^d (mod n).

• Numai utilizatorul A cunoaste cheia privata d.

DSA - este un standart pentru semnaturile digitale.

Algoritmul este format din trei proceduri: generarea cheii, semnarea, verificarea semnaturii.

Generarea cheilor

Fiecare entitate generează cheia publică şi cheia privată corespunzătoare. Entitatea A execută următoarele:

1. Generează un număr prim q astfel încât $2^{159} < q < 2^{160}$.

2. Generează un număr prim p astfel încât $2^{512} \leq p < 2^{1024}$ şi q|p-1

3. Selectează un generator α pentru grupul ciclic Z^{p^*} de ordin q

    3.1 Alege un element $g \in Z^{p^*}$ şi calculează $\alpha = g^{(p-1)/q} \bmod p$

    3.2 Dacă α = 1, atunci se execută pasul 3.1.

4. Se selectează un număr întreg a astfel încât $1 \leq a \leq q-1$.

5. Se calculează $y = \alpha^a \bmod p$.

6. Cheia publică a entității A este (p,q,α,y), iar cheia privată este a.

Generarea semnăturii

Entitatea A semnează un mesaj m astfel:

• Selectează aleator un număr întreg k astfel încât 0<k<q.

• Calculează $r = (\alpha^k \bmod p) \bmod q$.

• Calculează $k^{-1} \bmod q$.

• Calculează $s = k^{-1}(H(m)+a*r) \bmod q$, unde H este o funcție hash.

• Semnătura mesajului m este perechea (r,s).

Verificarea semnăturii

Pentru a verifica semnătura (r,s) a mesajului m, entitatea B execută următoarele:

• Obține cheia publică autentică (p,q,α,y) a entității B.

• Verifică dacă 0 < r< q şi 0 < s< q. Dacă aceste inegalități nu au loc, semnătura (r,s) nu e validă.

• Calculează $w = s^{-1} \bmod q$ şi H(m).

• Calculează $u^1 = w*H(m) \bmod q$ şi $u^2 = r*w \bmod q$.

• Calculează $v = (\alpha^{u1} y^{u2} \bmod p) \bmod q$.

Acest algoritm este considerat imposibil de spart, datorita sigurantei mari asigurate de câteva puncte, cum ar fi generarea aleatoare a lui p, q, a si k. Pentru a se afla k, de exemplu, ar trebui rezolvata o problema de tipul logaritmilor discreti, care este o problema „dificila”, in sensul ca ajungerea la o solutie poate dura câteva luni.

Algoritmul de distribuție a cheilor Diffie-Hellman

Metoda schimbului de chei Diffie-Hellman, cunoscută şi ca metoda de distribuţie a cheilor publice

Metoda Diffie-Hellman se bazează pe conceptul perechii de chei publică privată.

Protocolul începe cu fiecare parte care generează independent câte o cheie privată. În pasul următor, fiecare calculează câte o cheie publică, aceasta fiind o funcţie matematică a cheilor private respective. Urmează schimbul de chei publice.

În final, fiecare dintre cele două persoane calculează o funcţie a propriei chei private şi a cheii publice a celeilalte persoane. Matematica este cea care va face să se ajungă la aceeaşi valoare, care este derivată din cheile lor private. Ele vor folosi valoarea ca pe cheie a mesajului.

Diffie şi Hellman folosesc exponenţierea în aritmetica modulară pentru a calcula cheile publice şi cheia mesajului. Aritmetica modulară este ca şi aritmetica standard, cu excepţia faptului că foloseşte numere numai în intervalul 0 la N, numit modulo. Atunci când o operaţie produce un rezultat care este mai mare sau egal cu N, N este scăzut repetat din rezultat până când valoarea se încadrează în intervalul 0 la N-1 (ca şi cum s-ar împărţi la N şi se ia în seamă restul). De exemplu, 3+4 mod 5 = 2. Dacă rezultatul este negativ, N se adaugă acestuia până când se va încadra în intervalul 0 la N-1. De exemplu, 3-8 mod 7 = -5 mod 7 = 2.

În aritmetica modulară, exponenţierea este o funcţie într-un singur sens.

Aceasta înseamnă că este uşor de calculat un număr y = gx mod N pentru o valoare secretă x, însă este mult mai dificil să se calculeze x din y, dacă numerele sunt suficient de mari, ca de exemplu o lungime de câteva sute de cifre (noi presupunem că g şi N sunt cunoscute). Aceasta este referită ca şi problema logaritmului discret pentru că x este logaritm din y în baza g (mod N), iar numerele sunt finite şi întregi.

Cu metoda Diffie-Hellman a schimbului de chei publice, Alice şi Bob stabilesc cheia mesajului secret după cum urmează. Alice generează o cheie secretă xa şi Bob o cheie secretă xb. După aceasta, Alice calculează o cheie publică ya, care este g ridicat la puterea xa modulo p, unde p este un număr prim (adică nu poate fi descompus în produsul a două numere), g fiind mai mic decât p. Identic, Bob calculează o cheie publică yb, prin ridicarea lui g la puterea xb modulo p. Ei vor schimba valorile publice ale acestora. Apoi, Alice ridică cheia publică a lui Bob la puterea exponentului său, xa modulo p, în timp ce Bob ridică cheia publică a lui Alice la exponentul său, xb modulo p. Amândoi vor obţine acelaşi rezultat, g ridicat la puterea xa şi xb, iar rezultatul obţinut va fi folosit de amândoi drept cheia K a mesajului. Matematic, totul se va exprima astfel:

ya = gxa mod p

yb = gxb mod p

K = yaxb mod p = ybxa mod p = gxa*xb mod p

Deşi în practică se folosesc numere foarte lungi, de câteva sute de cifre, pentru a ajuta la înţelegerea modului de funcţionare, vom folosi numere mici.

Functii Hash

Definiție. *O funcție hash este o funcție care mapează un şir binar de o lungime arbitrară finită la un şir binar de o lungime fixată l :*

Proprietati:

• Rezistență la coliziune slabă: pentru un x dat, este greu de găsit un x'≠x astfel încât H(x) = H(x')

• Rezistență la coliziune puternică: este greu de găsit o pereche (x,x') cu x ≠ x' astfel încât H(x) = H(x'), dacă H este aleasă aleator dintr-o familie de funcții hash

• Greu inversabilă: pentru un c dat, este greu de găsit un x astfel încât H(x) = c.

• Una din cerințele fundamentale pentru o astfel de funcție este ca, modificând un singur bit la intrare, să producă o avalanşă de modificări în biții de la ieşire.

SHA-1

Algoritmul procesează un mesaj de lungime maximă 264 biți şi produce un rezumat de 160 de biți.

• Mai întâi mesajul este completat la multiplu de 512 biți, adică se adaugă 1 şi atâția de 0 până la 448 biți, iar ultimii 64 de biți memorează lungimea mesajului înainte de completare.

• Rezumatul MD de 160 de biți, văzut ca 5 regiştri A, B, C, D, E de 32 de biți, se inițializează cu următoarea constantă MD0:

• Apoi se prelucrează fiecare bloc Mj de 512 biți al mesajului.

• Fiecare prelucrare are 4 runde de câte 20 de operații fiecare.

• Funcția neliniară F, care se modifică la fiecare rundă, este definită astfel:

Runda 1: $Ft(B,C,D) = (B \wedge C) \vee (B' \wedge D)$, pentru t= 0,...,19

Runda 2: $Ft(B,C,D) = B \oplus C \oplus D$, pentru t=20,...,39

Runda 3: $Ft(B,C,D) = (B \wedge C) \vee (B \wedge D) \vee (C \wedge D)$, pentru t=40,...,59

Runda 4: $Ft(B,C,D) = (B \oplus C \oplus D)$, pentru t=60,...,79

• Se notează cu t numărul operației (t=0,...,79).

• Fiecare bloc Mj, j=0,...,15 ,de 16 cuvinte de 32 de biți este transformat în 80 de subblocuri Wj, j=0,...,79, folosind următorul algoritm:

$Wt = Mt$, t = 0,...,15

$Wt = (Mt-3 \oplus Mt-8 \oplus Mt-14 \oplus Mt-16) <<< 1$, t = 16,...,79

- Unde <<< K semnifică deplasarea circulară la stânga a cuvântului cu K poziții.
- S-a notat cu A' complementul de 1 al lui A, cu ∧ funcția AND, cu ∨ funcția SAU
- S-a notat cu ⊕ suma modulo 2 (XOR).
- În fiecare rundă se execută următoarele operații (t=0,...,79):

  TEMP = (A<<<5) +Ft(B,C,D)+E+Wt+Kt
  E = D
  C = B<<<30
  B = A
  A = TEMP

- Unde Kt este o constantă unică aditivă.
- În final avem: MDj = MDj + MDj-1.

MD5 (*Message Digest Algorithm 5*) este o funcţie criptografică de tip hash unidirecțional, care livrează ca rezultat o valoare fixă ca lungime de 128 Biți.

Este utilizată drept componentă în unele scheme de semnătură electronică, deși tinde să fie înlocuită în acest scop de SHA-1 sau RIPEMD-160, funcții mai puțin sensibile la coliziuni. Valoarea calculată cu ajutorul funcției MD5 (pe scurt md5sum), este folosită însă pe scară largă drept sumă de control, la verificarea integrității fișierelor.

un certificat digital creat cu ajutorul funcției MD5 încă nu poate să fie falsificat

n=pq ; Φ(n)=(p-1)(q-1); 1<e<Φ(n); (e, Φ(n))=1;

ed=1mod(Φ(n)); $d=e^{-1}$mod(Φ(n)); $c=m^e$(mod n); $m=c^d$(mod n).

1. Realizați criptarea şi decriptarea utilizând algoritmul RSA pentru următoarele date: a) p=3, q=11, d=7, M=5

n=3*11=33
Φ(n)=2*10=20
7*e=1 mod20
7*3 = 1 mod 20 =>e=3
C=5^3(mod 33) = 125*mod33=26 **=> c=26**

b) p=5, q=11, e=3, M=9

n=5*11=55 Φ(n)=4*10=40
d=3^-1 mod 40 = 27
c=9^3 (mod55)= 729(mod55)=14

c) p=7, q=11, e=17,M=8

n=7*11=77 Φ(n)=6*10=60
d=17^-1 mod 60 =53
c=8^17(mod77)=57

d) p=11,q=13,e=11,M=7

n=11*13=143 Φ(n)=10*12=120
d=11^-1 (mod 120)=11
c=7^11(mod 143)=106

e) p=17,q=31, e=7, M=2.

n=17*31=527 Φ(n)=16*30=480
d=7^-1 mod 480=343
c=2^7(mod527)= 128

2. Intr-o criptosistema RSA interceptați textul cifrat c=10 care este trimis la un utilizator ce are cheia publică e=5 şi n=35. Care este textul clar M?

Deoarece n=35 => p=5 q=7 si Φ(n)=4*6=24
d=5^-1 mod 24=5
m=c^d(mod n) = 10^5(mod 35) =5

3.Intr-o criptosistema RSA, cheia publică a unui utilizator este e=31 şi n=3599. Care este cheia privată a utilizatorului(d)?

P=59 și q=61 N=3599 Φ(n)=3480

D=e^-1 mod Φ(n)

31^-1 mod 3480