CRIPTARE

1. INTRODUCERE

Întotdeauna informația (religioasă, militară, economică, etc.) a însemnat putere, prin urmare dorința de a o proteja, de a o face accesibilă doar unor elite, unor inițiați, s-a pus din cele mai vechi timpuri. Primele texte cifrate descoperite până în prezent datează de circa 4000 de ani şi provin din Egiptul antic, dar existența acestora datează fără doar şi poate de la apariția scrierii în toate civilizațiile umane.

În Grecia antică scrierile cifrate erau folosite încă din secolul al V-lea î.e.n. Pentru cifrare se folosea un baston în jurul căruia se înfăşura, spirală lângă spirală, o panglică îngusta de piele, papirus sau pergament, pe care paralel cu axa se scriau literele mesajului. După scriere panglica era derulată, mesajul devenind astfel indescifrabil. Citirea mesajului putea fi făcută doar de persoana posesoare a unui baston identic cu cel utilizat în cifrare. În secolul al IV-lea î.e.n. în Grecia se cunoşteau 16 scrieri cifrate. Istoricul grec Polybius (sec ÎI î.e.n.) este inventatorul unui tabel de cifrare pătrat de dimensiune 5x5, tabel aflat la baza elaborării unui număr mare de sisteme de cifrare utilizate şi astăzi.

În Roma antică secretul informațiilor politice şi militare se făcea utilizând scrierea secretă. Amintim cifrul lui Cesar, utilizat încă din timpul războiului galic.

Documentele atestă chiar existența încă din antichitate a scrierilor secrete în Asia. Astfel literatura indiană dă o serie de indicii dintre care Artha-sastra (321-300 î.e.n.), Lalita-Vistara şi Kamasutra sunt cele mai cunoscute.

Stenografia, ştiința scrierilor secrete insesizabile camuflate în texte în clar, constituie o forma particulară de secretizare.

De o importanță remarcabilă la dezvoltarea criptologiei este contribuția arabă. David Kahn, unul din cei mai de seamă istoriografi ai domeniului, sublinia în cartea să *The Codebreakers* că, de fapt criptologia s-a născut în lumea arabă. Primele trei secole ale civilizației islamice (700-1000 e.n) au constituit, pe lângă o mare extindere politică şi militară şi o epoca de intense traduceri în limba arabă ale principalelor opere ale antichității greceşti, romane, indiene, armene, ebraice, siriene. Unele cărți sursă erau scrise în limbi deja moarte, deci prezentau în fapt texte cifrate. Astfel încât primii paşi în traducerea lor au fost traducerea acestora, deci originile criptologiei pot fi atribuite arabilor. Cartea lui Ahmad ibn Wahshiyyah (cca. 900 e.n.) conține 93 de alfabete ale diferitelor limbi, moarte sau vii.

Cartea a fost tradusă în limba engleză şi publicată la Londra nouă secole mai târziu sub numele *Ancient Alfabets and Hieroglyphic Caracters Explained*, iar aceasta din urmă a fost tradusă şi publicată la Paris în 1810 şi este posibil să-l fi ajutat pe celebrul arheolog J. F. Champollion în descifrarea hieroglifelor (scrierea de la Rosseta, aflată la British Museum). Dezvoltările criptanalizei au fost sprijinite de studiile lingvistice ale limbii arabe care în primele patru secole ale

imperiului islamic a constituit limba oficială unificatoare pentru un imperiu de o uriaşă întindere şi, în acelaşi timp, şi limba ştiinţifică. Arabii au preluat cunoştinţele matematice ale civilizaţiilor greceşti şi indiene. Arabii sunt cei care au introdus sistemele de numeraţie zecimal şi cifrele arabe drept urmare unii termeni cum ar fi ”zero”, ”algebră” li se datorează tot lor. Ultimele descoperiri demonstrează faptul ca manuscrise referitoare la probabilităţi şi statistici erau realizate cu 800 de ani înaintea celor scrise de Pascal şi Fermat şi însuşi termenul de ”cifru” provine de la arabi originea lui fiind reprezentată de cuvântul arab ”şifr” care reprezintă traducerea din sanscrită a cifrei zero.

Odată trezit interesul pentru civilizaţia antică, în perioada Renaşterii s-au descoperit lucrările criptografiei din antichitate. Extinderea relaţiilor diplomatice dintre diferitele state feudale a determinat o dezvoltare puternică a secretizării informaţiei. Curţile regale şi în special statul papal dispuneau de criptologi de mare valoare cum ar fi Giambattista della Porta, Vigénère, Cardan (secolul XIV) şi Rossignol (secolul XVIII), cel mai abil criptolog din Europa regilor Ludovic al XIII-lea şi Ludovic al XIV-lea.

Dezvoltare sau recesiunea criptografiei / criptologiei conchide cu evoluţia marilor imperii şi civilizaţii. Ea nu apare şi nu se dezvoltă decât acolo unde puterea trebuie protejata, drept urmare, apariţia telegrafului şi a radioului în secolul al XIX-lea precum şi cele două războaie mondiale din secolul XX au fost stimulente puternice în dezvoltarea metodelor şi tehnicilor de criptare.

Apariţia şi dezvoltarea continua a utilizării calculatoarelor în practic toate domeniile de activitate, existenta şi evoluţia puternică a reţelelor teleinformatice la nivel naţional, globalizarea comunicaţiilor, existenta unor baze de date puternice, apariţia şi dezvoltarea comerţului electronic, a poştei electronice, constituie premisele societăţii informatice în care păşim. Toate acestea indică o creştere extraordinară a volumului şi importanţei datelor transmise sau stocate şi implicit a vulnerabilităţii acestora. Protecţia în aceste sisteme vizează:

- eliminarea posibiltăţilor de distrugere voită sau accidentală.
- asigurarea caracterului secret al comunicării pentru a preveni posibilitatea ca persoane neautorizate să extragă informaţii în sistem.
- în situaţii cum ar fi transferurile electronice de fonduri , negocierile contractuale, este importantă existenţa unei semnături electronice pentru a evita dispute între emiţător şi receptor cu privire la mesajul transmis.

Toate aceste obiective arată o lărgire puternică a domeniului de aplicare al criptografiei de la domeniul diplomatic, militar, politic, la cel cu caracter economic şi social. La ora actuală, 99% nu este utilizată pentru protejarea secretelor militare ci pentru carduri bancare, plăţi de taxe radio/TV, taxe de drumuri, acces autorizat în clădiri sau la calculatoare, terminale de loterie, instrumente electronice de plăţi anticipate. În aceste aplicaţii rolul criptografiei este de a face furturile dacă nu imposibil cel puţin mai greu de realizat.

1.1 TERMINOLOGIE DE BAZĂ

Criptografia (cryptography) este ştiinţa creării şi menţinerii mesajelor secrete, în sensul imposibiltăţii citirii lor de către neautorizaţi (ştiinţa mesajelor secrete).

Mesaj (text) în clar (M) (plain / clear text) este mesajul ce urmează a fi secretizat; în criptografie M se numeşte scriere chiar dacă este un document de alta natura, de exemplu voce, imagine, date.

Mesaj cifrat (criptograma) (C) (cipher text) este mesajul secretizat, inaccesibil neavizaţilor.

Criptare / cifrare (E) (encryption / enciphering) este procedeul de ”ascundere” a unui mesaj în clar în mesajul secretizat.

E(M)=C

Decriptare / descifrare (D) (decryption / deciphering) este procedeul de regăsire a mesajului în clar M din mesajul cifrat C.

D(C) = D(E(M))=M

Observaţii:

- decriptarea este operaţia inversă criptării;
- ISO 7498-2 foloseşte termenii de cifrare / descifrare (encipher / decipher), probabil pentru ca primul set de termeni aminteşte de morţi (criptă).

Criptograf (cryptographer) este persoana care se ocupă cu criptografia.

Algoritm criptografic / cifru (cryptographic algorithm / cipher) este funcţia sau funcţiile matematice utilizate pentru criptare / decriptare; în general exista două funcţii: una pentru criptare (E) şi alta pentru decriptare (D).

Cheia criptografica (K) (key) este mărimea (în majoritatea cazurilor secretă) necesară realizării criptării şi decriptării.

Criptosistem (cryptosistem) este sistemul format din:

- algoritm
- toate mesajele în clar (M)
- toate textele cifrate (C)
- toate cheile (K).

Criptanaliza (cryptanalysis) este ştiinţa spargerii cifrurilor, deci a obţinerii mesajelor în clar (M) sau a cheii (K) din mesajul cifrat(C).

Criptanalist (cryptanayst) este persoana care se ocupă cu criptanaliza.

Atac(attack) este încercarea / tentativa criptanalitică.

Criptologie(cryptology) este ştiinţa care se ocupă atât de criptografie cât şi de criptanaliză.

Criptolog (cryptologist) este persoana care se ocupă cu criptologia.

Steganografia(steganography) este tehnica ascunderii mesajelor secrete în alte mesaje, în aşa fel încât existenta mesajelor secrete să fie invizibilă.

I.2 CRIPTOSISTEME ŞI ROLUL ACESTORA

Schema bloc a unui criptosistem este prezentată în figura alăturată:

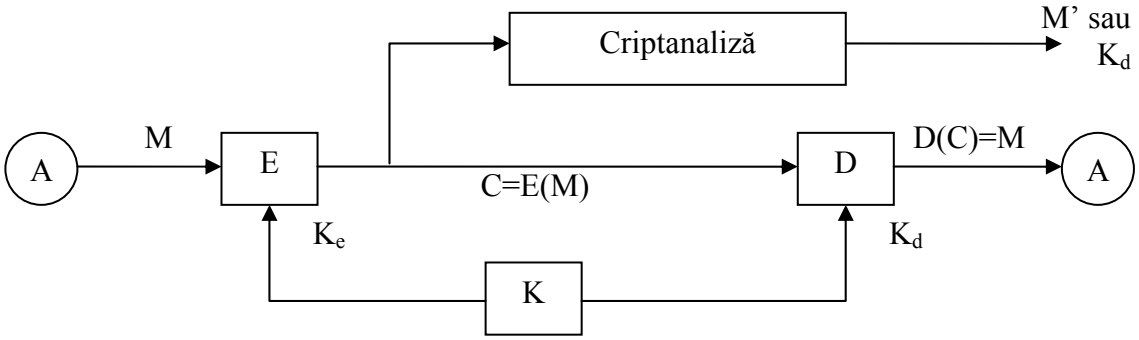


Figura I.1 Schema bloc a unui criptosistem

A,B - entităţi ce emit, recepţionează sau manipulează informaţia

E - funcţia de criptare(cifrare)

D - funcţia de decriptare(descifrare)

M - spaţiul mesajelor în clar

C - spaţiul criptogramelor(text cifrat)

K - spaţiul cheilor

Ke - spaţiul cheilor de criptare

Kd - spaţiul cheilor de decriptare

I.2.1 ROLUL UNUI CRIPTOSISTEM

Rolul unui criptosistem este de a preveni sau detecta activităţile nepermise dintr-un sistem informatic, cum ar fi:

- consultarea neavizată a datelor transmise sau stocate.
- inserarea de mesaje false.
- modificarea, ştergerea sau distrugerea mesajelor existente.
- analiza traficului.
- conectările neautorizate.

deci, acela de "poliţie informatică"

Îndeplinirea acestui rol presupune ca un criptosistem să realizeze:

- *Confidentialitatea / secretul / protecţia datelor (confidentiality / secrecy / privacy*) asigura inaccesibilitatea conţinutului informaţiei transmise / stocate pentru un utilizator neavizat (în cazul canalelor / mediilor nesigure în care pot apărea intruşi)
- *Autentificarea(autentification)* se aplica la :
  - entităţi (persoane, terminale, cărţi de credit, etc.) şi în acest caz se numeşte *autentificarea entităţilor / identificare (entity autentification / identification)*
  - informaţie: informaţia transmisă / stocată trebuie să fie autentificată în sensul

originii, conținutul datelor, timpul de transmisiune / stocare şi defineşte *autentificarea originii datelor / integritatea datelor (data origin autentification / data integrity)*

- *Semnătura digitală (digital signature)* reprezintă autentificarea reciprocă a datelor şi entităților şi se referă la faptul că un utilizator B este sigur că mesajul recepționat vine de la A prin semnarea mesajelor sale (SA), iar A este sigur ca nimeni nu-i poate atribui un mesaj fals deoarece mesajul transmis de către el îi poartă semnătura.

Observație **:** Aceste trei cerințe sunt vitale pentru interacțiunea "socială" în calculatoare şi sunt echivalente cu interacțiunile umane față în față.

Pe lângă cele trei obiective principale anterior enumerate, anumite Criptosisteme pot asigura şi alte obiective printre care amintim: *nerepudierea (non- repudiation), autorizarea (authorization), validarea (validation), controlul accesului (acces control), certificarea (certification), certificarea temporală (time-stamping), mărturia (witnessing), confirmarea (confirmation), dreptul de proprietate (ownership), anonimatul (anonimity), revocarea(revocation).*

Obiectivele pe care le poate realiza un criptosistem sunt cuprinse în tabelul următor:

Nr. crt.	**Obiectivul**		**Definirea**
1	Confidențialitatea		păstrarea secretului pentru neavizați
2	autentificarea	Emițător / identificare	legată de identitatea unei entități (persoană, terminal, carte de credit, etc.)
		originii datelor	legată de sursa de informație (sursa informație)
		integrității datelor	asigură că informația nu a fost modificată de surse neautorizate
3	Semnătură		mijlocul de a lega informația de entitate
4	Nerepudierea		prevenirea nerecunoaşterii unor acțiuni comise anterior
5	Autorizare		transferul către altă entitate a unei împuterniciri oficiale de a fi sau de a face ceva
6	Validare		mijlocul de a permite (pentru o anumită perioadă de timp) autorizarea de a folosi sau manipula informația sau resursele
7	Revocare		retragerea certificatului sau autorizației
8	Controlul		restricționarea accesului la resurse pentru

	accesului	entități privilegiate (cu drept de acces)
9	Certificare	informația de aprobare data de o entitate de încredere (arbitru)
10	Certificare temporala	înregistrarea timpului de creare sau de existență a informației
11	Mărturia	verificarea creării sau existenței unei informații de către o entitate diferită decât cea creatoare
12	confirmarea	recunoaşterea faptului ca serviciile au fost efectuate
13	dreptul de proprietate	mijlocul de a înzestra o entitate cu dreptul legal de a utiliza sau transfera o sursa la alții
14	anonimatul	ascunderea (tăinuirea) identității unei entități implicate într-un anumit proces

Observații:

- toate aceste obiective există natural în interacțiunile umane față în față;
- apariția rețelelor de calculatoare prin care au loc interacțiunile interumane pune o serie de probleme în plus: hârtia se înlocuieşte cu discul, unda radio, firul; scrisul are un format electronic şi anume "biți" cu posibilitate extrem de uşoară de copiere (copiile digitale sunt identice); utilizatorii se află la distanță, nu se pot vedea față în față şi deci se lucrează în condiții de neîncredere.
- criptografia este chemată să soluționeze o diversitate de probleme, multe şi de mare dificultate.

1.2.2 CLASIFICAREA CRIPTOSISTEMELOR

După tipul cheilor folosite pentru criptare / decriptare, criptosistemele se clasifică în două categorii şi anume în sisteme: cu chei simetrice şi cu chei publice.

1.2.2.1 Criptosisteme cu chei simetrice / cu cheie secretă / criptosisteme convenționale

Criptosistemele cu chei simetrice sunt criptosistemele pentru care cheile folosite la criptare şi decriptare sunt identice, de unde şi denumirea de criptosisteme cu chei simetrice.

$$K_e = K_d = K \tag{1.1}$$

Cheia K este secretă, fapt ce conduce la denumirea de *sisteme cu cheie secretă.* Criptarea şi decriptarea se realizează extrem de simplu dacă se cunoaşte K :

$$E_K(M) = C \tag{1.2}$$

$$D_K(C) = D_K(E_K(M)) = M \tag{1.3}$$

Problema ce se pune în cazul acestor sisteme este distribuția cheii K, care necesită un canal sigur.

După tipul algoritmului folosit, criptosistemele cu chei simetrice se clasifică în două categorii:

- *cu cifruri bloc (block ciphers)* sunt cifrurile care acționează asupra unei diviziuni a textului în clar; blocurile de la intrare se cifrează independent, lungimea tipică a blocurilor fiind $n=32 \div 128$ biți. Transformările de bază folosite pentru criptare şi decriptare sunt substituțiile şi transpozițiile, repetate iterativ.
- *cu cifruri secvențiale (stream ciphers)*: mesajul de la intrare este considerat ca o succesiune (şir) de simboluri. Criptarea operează asupra textului în clar simbol cu simbol. Cheia K este generată de un registru de deplasare cu reacție (RDR) având starea inițială S_0 controlată de o cheie compactă.

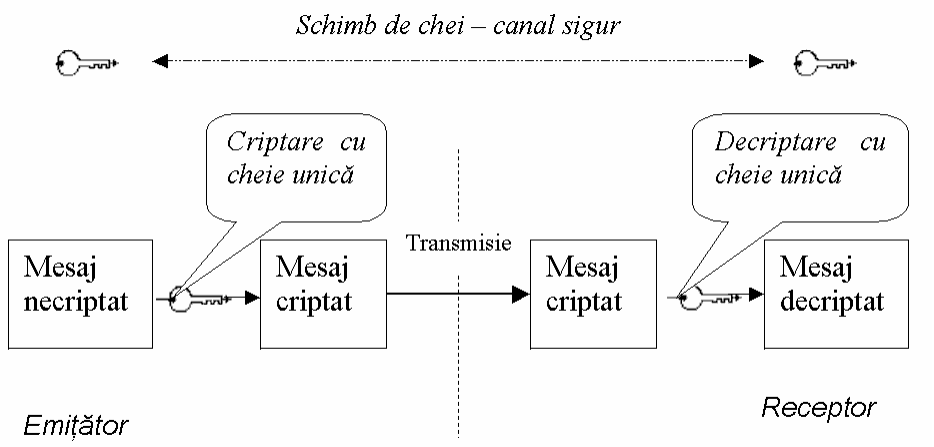


Fig.1.3. Schema de aplicare a unui algoritm simetric

Deoarece algoritmul este valid în ambele direcții, utilizatorii trebuie să aibă încredere reciprocă. Securitatea acestui tip de algoritm depinde de lungimea cheii şi posibilitatea de a o păstra secretă. Când comunicațiile dintre numeroşi utilizatori trebuie să fie criptate, apare o mare problemă a managementului cheilor; pentru n utilizatori sunt posibile $n(n-1)/2$ legături bidirecționale, fiind necesare tot atâtea chei. Aceasta implică în general probleme dificile în generarea, distribuția şi memorarea cheilor. Utilizarea calculatoarelor electronice a permis folosirea unor chei de dimensiuni mai mari, sporindu-se astfel rezistența la atacuri criptoanalitice. Când cheia secretă are o dimensiune convenabilă şi este suficient de frecvent schimbată, devine practic imposibilă spargerea cifrului, chiar dacă se cunoaşte algoritmul de cifrare.

I.2.2.2 Criptosistemele cu chei publice (asimetrice)

În locul unei singure chei, secrete, criptografia asimetrică foloseşte două chei, diferite, una pentru cifrare, alta pentru descifrare. Deoarece este imposibilă deducerea unei chei din cealaltă, una din chei este făcută publică fiind pusă la dispoziția oricui doreşte să transmită un mesaj cifrat.

Doar destinatarul, care deține cea de-a doua cheie, poate descifra şi utiliza mesajul. În sistemele cu chei publice, protecția şi autentificarea sunt realizate prin transformări distincte.

Criptosistemele cu chei publice sunt criptosistemele pentru care cheile folosite la criptare şi decriptare diferă, de unde derivă şi denumirea de sisteme asimetrice:

$$K_e \neq K_d \tag{1.4}$$

Cheia publică - aşa cum îi spune şi numele - poate fi accesata public (asemănător unei cărți de telefon), iar cheia privată este secretă. Caracteristic acestor sisteme este ca funcția de criptare şi cea de decriptare se realizează uşor dacă se cunosc K_d şi K_e:

$$\begin{cases} E_{K_e}(M) = C \\ D_{K_d}(C) = M \end{cases} \tag{1.5}$$

dar aflarea lui M, dacă se cunosc C şi K_e este computațional nefezabilă.

Tehnica cheilor publice poate fi folosită şi pentru autentificarea mesajelor, prin aşa numita semnătură digitală, fapt care i-a sporit popularitatea. Folosind *algoritmi cu cheie publică (asimetrici*), se creează criptosisteme cu două chei, prin care doi utilizatori (procese) pot comunica cunoscând fiecare doar cheia publică a celuilalt.

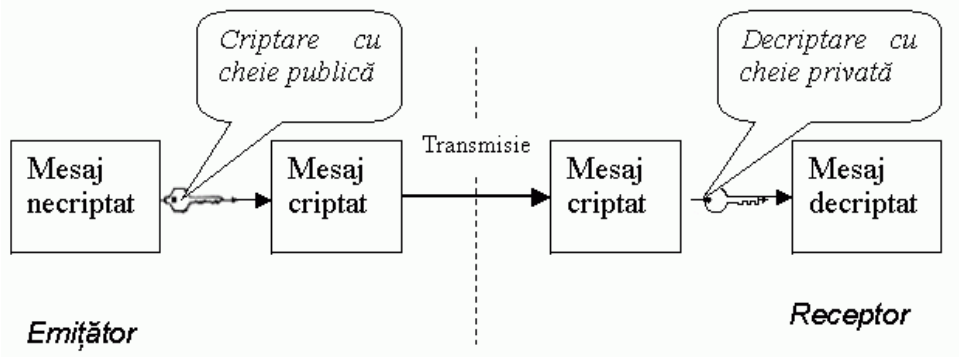


Fig.1.4. Schema de aplicare a unui algoritm asimetric

Observație: E este o funcție neinversabilă cu trapă. K_d este trapa care furnizează informația necesară calculării funcției inverse D. Dintre funcțiile neinversabile cu trapă amintim: factorizarea unui produs de numere prime mari cu peste 100 de cifre zecimale (folosit în algoritmul RSA), găsirea logaritmului modulo un număr prim într-un câmp Galois GF(q) cu q foarte mare (folosit în algoritmii Rabin şi Diffie-Hellman), problema rucsacului folosită în algoritmul (Markle-Helman).

2. ATACURI ŞI MODELE DE SECURITATE ÎN SISTEMELE INFORMATICE

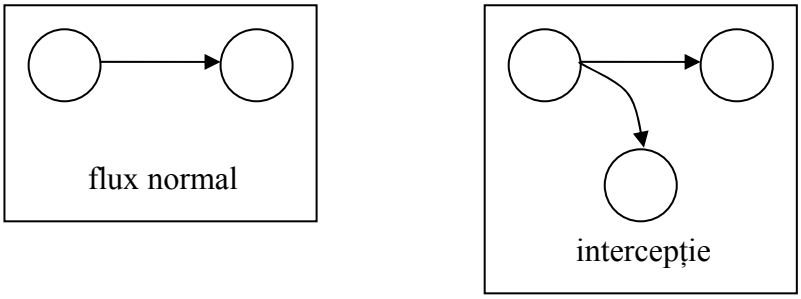
2.1 ATACURI ASUPRA SECURITĂȚII SISTEMELOR CRIPTOGRAFICE

Atacul asupra securității unui sistem criptografic defineşte orice acțiune ce compromite securitatea acelui sistem.

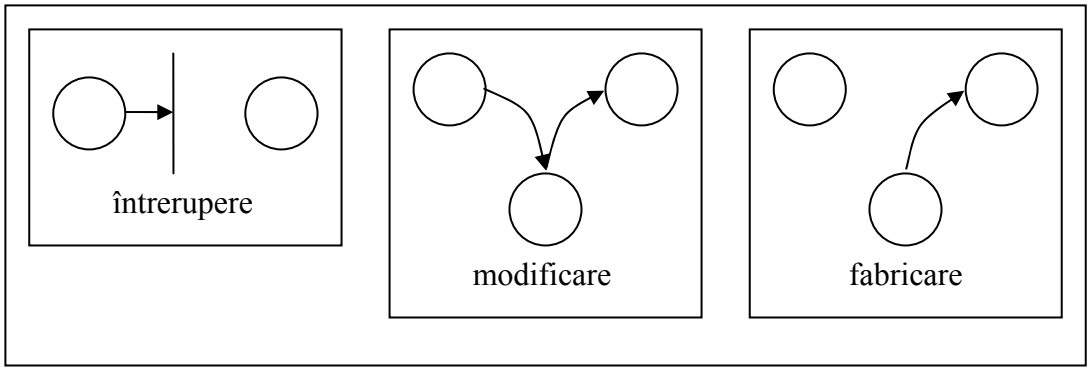
Atacurile criptografice pot fi îndreptate împotriva :

- algoritmilor criptografici
- tehnicilor utilizate pentru implementarea algoritmilor protocoalelor
- protocoalelor

O ilustrare sugestivă a principalelor tipuri de atacuri asupra unui sistem informatic este prezentată succint în figura alăturată:



Atac pasiv
(atac la confidențialitate)



Atacuri active

După modelul de atacare al unui *atacator / intrus / persoană neautorizată / pirat (attacker / intruder / pirat*), aceste atacuri se pot clasifica după cum urmează:

- atacuri pasive (de intercepție):
  - de înregistrare a conținutului mesajelor
  - de analiză de trafic;
- atacuri active :
  - de întrerupere (atac la disponibilitate)

- de modificare (atac la integritate)
- de fabricare(atac la autenticitate).

Atacurile pasive sunt atacuri în care intrusul (persoana, calculator, program) doar ascultă, monitorizează transmisia, deci sunt *atacuri de interceptie*. Ele pot fi de două feluri:

- *de înregistrare a conținutului mesajelor(release of message contents)*, de exemplu în convorbirile telefonice, în poşta electronică, fapt pentru care dacă mesajele nu sunt criptate, se violează caracterul confidențial al comunicației.
- *de analiză a traficului (trafic analysis)*: în cazul în care mesajele sunt criptate şi nu se poate face rapid criptanaliza, prin analiza traficului se pot afla o serie de date utile criptanalizei precum identitatea părților ce comunica între ele , frecvența şi lungimea mesajelor

Caracteristicile atacurilor pasive:

- sunt greu de detectat pentru că datele nu sunt alterate ;
- măsurile ce pot fi luate pentru evitarea acestor atacuri sunt acelea care fac criptanaliza extrem de grea, dacă nu imposibilă;
- este necesară prevenirea şi nu detecția lor.

Atacurile active sunt atacuri în care intrusul are o intervenție activă atât în desfăşurarea normală a traficului, cât şi în configurarea datelor (modificarea datelor, crearea unor date false). Dintre atacurile active amintim :

- *întreruperea / refuzul serviciului (denial of service)* un bloc funcțional este distrus, sau se inhibă funcționarea normală sau managementul facilitaților de comunicație; acest tip de atac este un *atac la disponibilitate(attack on availability)*
- *modificarea:* mesajul inițial este întârziat, alterat, reordonat pentru a produce efecte neautorizate cum ar fi :
  - schimbare de valori în fişiere de date;
  - modificări în program astfel încât acesta va lucra diferit;
  - modificarea conținutului mesajelor transmise în rețea
- *fabricarea:* un neavizat inserează informații false în sistem; acest atac este un *atac de autenticitate*. Din această categorie fac parte şi:
  - *masacrarea (masaquerade):* o entitate pretinde a fi altă entitate.

  Exemplu: secvențele de autentificare pot fi capturate şi după validarea unei autentificări se înlocuiesc, permițând astfel unei entități să obțină privilegii pe care nu le are de drept.
  - *reluarea(replay)* constă în capturarea prin atac pasiv a unei cantități de informație şi transmiterea să ulterioară pentru a produce efecte neautorizate

Caracteristicile atacurilor active:

-deşi pot fi detectate, prevenirea lor este foarte grea, deoarece ar însemna protecție fizică permanentă a întregului sistem.

2.2 ATACURI CRIPTANALITICE

Atacurile criptanalitice sunt atacuri asupra textelor cifrate în vederea obținerii textului în clar sau a cheilor folosite pentru decriptare .

Există mai multe tipuri de asemenea atacuri dintre care amintim:

1) *Atac asupra textului cifrat (cipher text-only attack)*: criptanalistul are criptograma $C_i = E_k(M_i)$ şi trebuie să obțină mesajul în clar (M_i) sau cheia k.

2) *Atac asupra unui text cunoscut (chiper text - only attack):* criptanalistul are criptogramele C_i şi mesajele în clar M_i corespunzătoare şi trebuie să determine cheia k sau algoritmul de determinare a lui M_{i+1} din $C_{i+1} = E_k(M_{i+1})$

3) A*tac cu text în clar ales (chosen plain-text attack)* : se p*ot alege o serie de mesaje* M_i *după dorința şi se cunosc criptogramele corespondente :* $C_i = E_k(M_i)$. *Criptanalistul trebuie să determine cheia k sau algoritmul de determinare a lui* M_{i+1} din $C_{i+1} = E_k(M_{i+1})$.

4) *Atac cu text în clar ales adaptiv (adaptive chosen plain-text attack)*: mesajele în clar M_i se pot alege după dorința şi sunt adaptabile în funcție de rezultatele criptanalizelor anterioare, iar criptogramele $C_i = E_k(M_i)$ se cunosc. Se cere cheia k sau algoritmul de determinare a lui M_{i+1} din $C_{i+1} = E_k(M_{i+1})$.

Aceste patru atacuri constituie atacurile criptanalitice de bază. În afara acestora mai putem aminti:

5) *Atac cu text cifrat la alegere (chosen cipher-text attack*) în care se aleg C_i şi $M_i = D_k(C_i)$, sarcina criptanalistului fiind determinarea cheii k. Acest atac se aplică mai ales algoritmilor cu chei publice.

6) *Atacul de "cumpărare" a cheii*(*rubber-hose cryptanalysis / purchase-key attack),* în care aflarea cheii se face fără mijloace criptanalitice (se apelează la şantaj , furt , etc.) şi este unul dintre cele mai puternice atacuri.

<u>*Observație*</u>: Atacurile 3) şi 4) se numesc *atacuri cu text ales (chosen text attack)* şi au fost folosite cu succes în cel de-al doilea război mondial pentru spargerea codurilor german şi japonez.

2.3 SECURITATEA ALGORITMILOR

În secolul al XIX-lea, olandezul A. Kerckhoff a enunțat conceptul fundamental al criptanalizei: Criptanaliza *se rezumă în întregime la cheie,* algoritmul criptografic şi implementarea considerându-se cunoscute.

Diferiții algoritmi pot asigura diferite grade de securitate, funcție de dificultatea cu care pot fi sparți:

- dacă costul spargerii unui algoritm este mai mare decât valoarea datelor criptate, algoritmul este probabil sigur (PS);
- dacă timpul necesar spargerii este mai mare decât valabilitatea datelor criptare, algoritmul

este PS;

- dacă mulțimea datelor necesare spargerii este mai mare decât mulțimea datelor criptate la un moment dat de o cheie, algoritmul este PS;

Lars Knudsen - în teza de doctorat susținută în 1944 - a clasificat diferitele categorii de spargere a unui algoritm în ordine crescătoare a securității:

1)*Spargere totală (total break) / securitate zero:* un criptanalist găseşte cheia, deci orice criptograma va fi decriptată: $D_k(C) = M$.

2)*Deducție globală (global deduction)*: criptanalistul găseşte un algoritm alternativ echivalent cu $D_k(C)$ fără a cunoaşte cheia k.

3)D*educție locală (local deduction)*: un criptanalist găseşte textul în clar al unui text cifrat interceptat.

4)*Deducția informațională (informațional deduction)*: criptanalistul capătă unele informații privitor la cheie sau la textul în clar (de exemplu câțiva biți ai cheii, anumite informații privitoare la M etc.)

5)*Algoritm computațional puternic (computațional strong)* este algoritmul care poate fi spart cu resursele existente, atât la momentul curent, cât şi într-un viitor predictibil.

6)*Algoritm necondiționat sigur (unconditional secure)* este algoritmul pentru care indiferent cât text cifrat are criptanalistul, informația nu este suficientă pentru a deduce textul în clar. Privitor la aceşti din urmă termeni trebuie atenționat că sunt extrem de expuşi interpretărilor.

Observații:

- Doar *cheia de unică folosință (one time pad),* inventată în 1917 de Major J. Maubergue şi GilbertVernon*,* având aceeaşi lungime cu a textului în clar, este de nespart.
- Toți ceilalți algoritmi pot fi sparți cu ajutorul unui atac cu text cifrat, prin încercarea tuturor cheilor, până când textul descifrat are sens. Acest atac se numeşte *atac prin forță brută (brute force attack).*

Complexitatea unui atac (complexity) se manifesta în mai multe feluri:

a) Complexitatea *datelor (data complexity)* este volumul de date necesar pentru un atac;

b) *Complexitatea procesării / factorul de lucru (procesing complexity / work factor)* este timpul necesar realizării atacului;

c) *Complexitatea stocării (storage complexity)* este cantitatea de memorie necesară atacului.

Regulă: *Complexitatea unui atac = max {a, b, c}.*

Exemplu: complexitatea de 2^{128} indică 2^{128} operații necesare spargerii cifrului(de obicei aceasta reprezintă complexitatea atacului prin forța brută, deci în cazul unui algoritm necondiționat sigur, 128 indicând lungimea cheii în biți).

Observație: Multe atacuri se pretează paralelismului, deci complexitatea scade față de regula anterior enunțată.

3. CRIPTOGRAFIE CLASICĂ (PRECOMPUTAȚIONALĂ, SIMETRICĂ)

Criptografia clasică este criptografia dinaintea calculatorului, de unde şi denumirea de *criptografie precomputațională*.

În criptografia clasică, algoritmii erau bazați pe caracter şi constau dintr-o serie de transformări elementare (substituții, transpoziții) ale caracterelor textului în clar. Unii algoritmi aplicau aceste transformări în mod repetat, *îmbunătățind* în acest fel securitatea algoritmului.

În criptografia modernă bazată pe calculator (criptografie computațională), lucrurile s-au complicat, dar multe din ideile criptografiei clasice au rămas nemodificate.

Criptografia clasică se încadrează în clasa criptografiei cu chei simetrice.
Criptografia cu chei simetrice se referă la sistemele care folosesc aceiaşi cheie atât la criptare cât şi la decriptare. Modelul unui sistem cu chei simetrice este prezentat în figura 3.1.

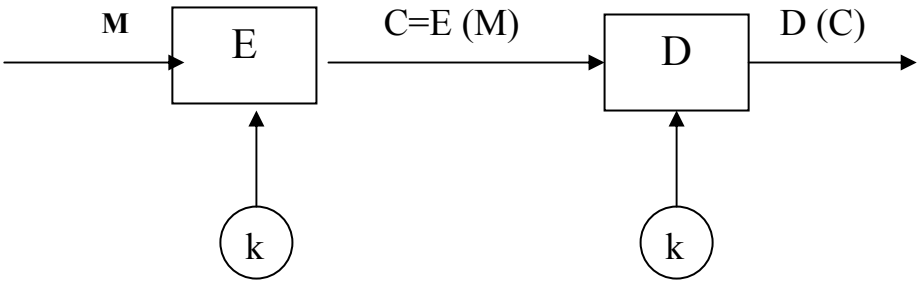


Figura 3.1 Schema bloc a unui sistem de criptare cu chei simetrice

k - cheie de criptare ; E - bloc de criptare (encryption) ; D - bloc de decriptare (decryption) ;
M - mesaj în clar ; C - mesaj criptat .

Clasificarea metodelor simetrice

În cadrul metodelor de criptografie simetrică distingem 3 categorii diferite :

1. Cifruri substituție;
2. Cifruri transpoziție;
3. Cifruri combinate.

3.1 CIFRURI DE SUBSTITUȚIE

Cifrul de substituție (substituion cipher) este cifrul bloc la care fiecare caracter sau grup de caractere ale textului în clar (M) este substituit cu un alt caracter sau grup de caractere ale textului cifrat (C), descifrarea făcându-se prin aplicarea substituției inverse asupra textului cifrat.

În criptografia clasică există patru tipuri de cifruri de substituție:

3.1.1) Cifruri de substituție monoalfabetică (monoalphabetic ciphers) sunt cifruri în care fiecare caracter al textului în clar (M) este înlocuit cu un caracter corespondent al textului cifrat (C).Vom aminti câteva dintre cifrurile de substituie cele mai cunoscute:

A. Cifrul Caesar

În cazul sistemului CAESAR substitutul unei litere se obține prin translarea să în față k paşi în alfabet. În sistemul CAESAR şi în alte sisteme naturale se foloseşte *codificarea numerică* naturală:

A	B	C	D	E	F	G	H	I	J	K	L	M
0	1	2	3	4	5	6	7	8	9	10	11	12
N	O	P	Q	R	S	T	U	V	W	X	Y	Z
13	14	15	16	17	18	19	20	21	22	23	24	25

Notând literele de la 0 la 25, în CAESAR fiecare litera α devine $\alpha + k$, calculele efectuându-se modulo 26.

Numărul cheilor posibile în CAESAR este foarte mic şi pe lângă aceasta un alt mare dezavantaj, din punct de vedere al securității, este acelă că secvența substitutelor păstrează ordinea alfabetică; doar poziția inițială se schimbă.

B. Cifrul lui Polybius

Este un cifru de substituție. Literele alfabetului latin sunt aşezate într-un pătrat de dimensiune 5x5. Literele I şi J sunt combinate pentru a forma un singur caracter, deoarece alegerea finală (între I şi J) poate fi uşor decisă din contextul mesajului. Rezultă 25 de caractere aşezate într-un pătrat 5x5 cifrarea oricărui caracter făcându-se alegând perechea potrivită de numere (intersecția liniei şi a coloanei) corespunzătoare dispunerii caracterului în pătrat.

	1	2	3	4	5
1	A	B	C	D	E
2	F	G	H	IJ	K
3	L	M	N	O	P
4	Q	R	S	T	U
5	V	W	X	Y	Z

Tabelul I.2 – Pătratul lui Polybius

Exemplu: Mesajul: ”AM CASTIGAT LUPTA”, se transformă după cifrare în: ”11233111344443221144 1354534411”.

Observație: Codul poate fi schimbat prin rearanjarea literelor în pătratul 5x5.

În sistemele UNIX, programul de criptare ROT 13 este un cifru de substituție monoalfabetică; fiecare literă, în textul cifrat se rotește cu 13 caractere, de unde și denumirea de ROT 13:

C = ROT 13 (M)

iar decriptarea se face aplicând de două ori ROT 13, dat fiind că alfabetul latin conține N = 26 litere:

M = ROT 13 (ROT 13(C))

acest cifru nu este în realitate un cifru de securitate; el se utilizează adesea în posturile de utilizator de rețea pentru a ascunde texte potențial ofensive.

Concluzie: Cifrurile de substituție monoalfabetică pot fi sparte cu ușurință deoarece frecvențele literelor alfabetului nu se schimbă în textul cifrat față de textul în clar.

3.1.2. Cifruri de substituție omofonica (homophonic substitution ciphers) sunt cifrurile de substituție în care un caracter al alfabetului mesajului în clar (alfabet primar) poate să aibă mai multe reprezentări.

Ideea utilizată în aceste cifruri este uniformizarea frecvențelor de apariție a caracterelor textului cifrat (alfabet secundar), pentru a îngreuna atacurile criptanalitice.

Astfel, litera A – cu cea mai mare frecvență de apariție în alfabetul primar – poate fi înlocuită cu F, * sau K.

Concluzii:

- deși mai greu de spart decât cifrurile de substituție simple (monoalfabetice), ele nu maschează total proprietățile statistice ale mesajului în clar.
- în cazul unui atac cu text în clar cunoscut, cifrul se sparge extrem de ușor.
- atacul cu text cifrat este mai dificil, dar unui calculator îi va lua doar câteva secunde pentru al sparge.
-

3.1.3. Cifrul de substituție poligramică (polygram substitution ciphers) se obțin substituind blocuri de caractere ale alfabetului primar – numite poligrame - cu alte blocuri de caractere, de exemplu:

ABA → RTQ

SLL → ABB

Cifrurile bazate pe substituția poligramică realizează substituirea unor blocuri de caractere (poligrame) din textul clar, distrugând astfel semnificația, atât de utilă în criptanaliză, a frecvențelor diferitelor caractere.

Considerăm un mesaj $M=m_1m_2...m_dm_{d+1}...$ și un cifru care prelucrează poligrame de lungime d. Criptograma rezultată este $C=c_1c_2...c_dc_{d+1}...c_{d+d}$. Fiecare poligramă $m_{id+1}...m_{id+d}$ va fi prelucrată în poligrama $c_{id+1}...c_{id+d}$ prin funcția de substituție f_i astfel : $C_{id+j}=f_j (m_{id+1}, ..., m_{id+d})$.

În cazul cifrării literelor singulare frecvența de apariție a acestora în textul cifrat este aceeaşi cu frecvența de apariție a literelor corespondente din textul clar. Aceast lucru furnizează o cantitate de informație suficientă criptanalistului pentru spargerea sistemului. Pentru minimizarea informației furnizate de frecvența de apariție a literelor s-a procedat la cifrarea grupurilor de litere (n-grame). În cazul în care un grup de n litere este substituit printr-un alt grup de n litere, substituția se numeşte poligramică; cel mai simplu caz se obține pentru n=2, când diagrama m_1m_2 din textul clar se substitue cu diagrama c_1c_2 din textul cifrat.

Un exemplu clasic pentru substituția diagramelor este cifrul lui **Playfair**.

Metoda constă în dispunerea literelor alfabetului latin de 25 de litere într-un pătrat de 5 linii şi 5 coloane (i=j) de forma :

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

De regulă, în prima linie a pătratului se scrie un cuvânt cheie şi apoi se completează celelalte linii cu literele alfabetului, fără repetarea literelor din prima linie. Cifrarea se execută după următoarele reguli :

- dacă m_1, m_2 sunt dispuse în vârfurile opuse ale unui dreptunghi, atunci c_1, c_2 sunt caracterele din celelalte vârfuri ale dreptunghiului, c_1 fiind în aceeaşi linie cu m_1. De exemplu GS devine MN.

- dacă m_1 şi m_2 se găsesc într-o linie, atunci c_1 şi c_2 se obțin printr-o deplasare ciclică spre dreapta a literelor m_1 şi m_2. De exemplu AD devine BF sau CF, DA.

- dacă m_1 şi m_2 se află în aceeaşi coloană atunci c_1 şi $c_2$se obțin prin deplasarea ciclică a lui m_1, m_2 de sus în jos. De exemplu UO devine BW, iar EZ devine FE. Descifrarea se execută după reguli asemănătoare cu cele de cifrare.

3.1.4. Cifruri de substituție polialfabetice sunt formate din mai multe cifruri de substituție simple. Au fost inventate de Leon Battista, în 1568. Cifrurile bazate pe substituția polialfabetică constau din utilizarea unor substituții monoalfabetice diferite.

Fie d alfabete de cifrare c_1, c_2, ..., c_d, şi d funcții f_i care realizează substituția de forma:

$f_i : A \rightarrow C_i,\ 1 \le i \le d$.

Un mesaj clar $M = m_1m_2...m_dm_{d+1}...m_{2d}...$ va fi cifrat prin repetarea secvențelor de funcții f_1, f_2, ..., f_d: $E_k(M) = f_1(m_1) ...f_d(m_d)\ f(m_{d+1})$.

Utilizarea unei secvențe periodice de substitiții ale alfabetului măreşte în mod semnificativ securitatea criptogramei prin nivelarea caracteristicilor statistice ale limbii. Aceeaşi literă din textul cifrat poate reprezenta mai multe litere din textul clar, cu diferite frecvențe de apariție. În acest caz numărul cheilor posibile se măreşte de la 26!, câte erau la substituția monoalfabetică, la $(26!)^n$.

A. cifrul lui Vigenere,

In acest caz cheia k este o secvență de litere de forma : $k=k_1k_2...k_d$.

Funcțiile f_i de substituție se definesc astfel : $f_i(a) = (a+k_i) \bmod n$, unde n este lungimea alfabetului .

Ca un exemplu considerăm cheia de 8 litere ACADEMIE care va fi utilizată repetitiv pentru cifrarea mesajului SUBSTITUȚIE POLIALFABETICĂ. Folosind o corespondență biunivocă între literele alfabetului şi elementele claselor de resturi modulo 26 (A=0, B=1, ..., Z=25), substituția 8-alfabetică conduce la următorul text:

- text clar : SUBSTITUȚIE POLIALFABETICĂ ;
- cheia : ACADEMIEACADEMIEACADEMIEA;

S + A = 18 + 0 (mod 26) = 18 (mod 26) = 18 = S

U + C = 20 + 2 (mod26) = 22 (mod 26) = 22 = W

B + A = 1 + 0 (mod26) = 1 (mod26) = 1 = B

...

C + E = 2 + 4 (mod 26) = 6 (mod26) = 6 = G

A + A = 0 + 0 (mod26) = 0 (mod26) = 0 = A

- text cifrat : SWBVXUBYTKESSXQELHAEIFQGA .

B. Sistemul Autoclave

O modificare suplimentară a sistemului **Vigenere** este sistemul **Autoclave** (cifrul **Vigenere** cu cheie în clar sau cheie de încercare). Caracterele textului în clar servesc drept cheie de criptare cu o anumită deplasare.

Decriptarea se face în modul următor: din începutul criptotextului, pe baza cuvântului cheie, se obține începutul textului în clar, după care se foloseşte drept cheie textul clar deja disponibil .

Într-o altă variantă a lui **Autoclave** (cifrul lui **Vigenere** cu autocheie sau cu cheie cifrată) criptotextul deja creat foloseşte drept cheie adăugându-se în continuarea cuvântului cheie.

Observație: Deşi fiecare caracter utilizat drept cheie poate fi găsit din caracterul anterior al textului cifrat, el este funcțional dependent de toate caracterele anterioare ale mesajului, inclusiv de cheia de încercare. Urmare a acestui fapt este efectul de fuziune a proprietăților statistice ale textului în clar asupra textului cifrat, ceea ce face ca analizele statistice să devină foarte grele pentru criptanalist.

Schemele de cifrare Vigénère nu sunt foarte sigure având în vedere standardele actuale însă contribuția importantă a lui Vigénère constă în faptul ca a descoperit ca pot fi generate secvențe nerepetitive drept cheie, prin utilizarea a însuşi mesajului sau a unor părți ale acestuia.

C. Cifrul Vernam

Cifrul **Vigenere** cu o perioadă n, deşi mult mai puternic decât un cifru bazat pe substituția monoalfabetică, poate fi spart dacă criptanalistul dispune de cel puțin 2n caractere din textul cifrat.
O variantă mai nouă a acestui cifru peste un alfabet binar, este cifrul **Vernam** care se diferențiază prin cheia de cifrare, care este reprezentată de o secvență de caractere aleatoare care nu se repetă.
Fie $M=m_1m_2...m_n$ un mesaj clar binar şi $K=k_1k_2...k_n$ un şir binar care reprezintă cheia. Criptograma $C = E_k (M) = c_1c_2...c_n$ se obține determinând fiecare caracter c_i astfel : $C_i = (m_i+k_i) \pmod n$,
$i=1, 2, ... ,n$.

Utilizarea o singură dată a cheii ne dă un exemplu de criptosistem care face ca mesajul să fie foarte rezistent la criptanaliză, practic imposibil de spart; textul clar este o secvență finită de biți, să zicem cel mult 20. Cheia este şi ea o secvența de 20 de biți. Ea este utilizată atât pentru criptare, cât şi pentru decriptare şi este comunicată receptorului printr-un canal sigur. Să luăm cheia 11010100001100010010;
un text clar, să zicem 010001101011, este criptat prin sumare bit cu bit cu biții cheii, pornind cu începutul acesteia.

11010100001100010010 ⊕
010001101011
_____
100100101000

Deci criptotextul este 100100101000. Acesta nu dă informații criptanalistului, deoarece el nu ştie nici ce biți vin din textul clar, nici care au fost schimbați de către cheie. Aici este esențial faptul că o cheie este folosită o singură dată, altfel, un text clar anterior împreună cu criptotextul corespunzător oferă posibilitatea aflării cheii sau cel puțin prefixul ei.
Dezavantajul evident al acestei tehnici este gestionarea dificilă a cheilor. Cheia, cel puțin la fel de lungă ca şi textul clar, trebuie comunicată printr-un canal sigur. Aceste cifruri au o largă utilizare în comunicațiile diplomatice şi militare.

D.Cifrul lui Trithemius

Este un cifru polialfabetic. Alfabetul este dispus pe 26 de linii numerotate de la 0 la 25**,** unde numărul de ordine al liniei indică numărul de caractere cu care se deplasează ciclic alfabetul spre dreapta. Linia numerotată cu 0 constituie tocmai alfabetul în ordinea inițială. Acest cifru poate fi utilizat astfel: primul caracter se cifrează selectându-l din linia 1, al doilea din linia a 2-a şi aşa mai departe.

Exemplu: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17
Mesajul: ”T R E B U I E S Ă I N V I N G E M”,
se cifrează: ”UTHFZOL AJ SYHVBVUD”.

3.2 CIFRURI DE TRANSPOZIȚIE

Spre deosebire de cifrurile cu substituție, care păstrează ordinea literelor din textul sursă dar le transformă, cifrurile cu transpoziție ("transposition ciphers") reordonează literele, fără a le "deghiza".

Un exemplu simplu este transpoziția pe coloane, în care textul sursă va fi scris literă cu literă şi apoi citit pe coloane, în ordinea dată de o anumită cheie. Ca şi cheie se poate alege un cuvânt cu litere distincte, de o lungime egală cu numărul de coloane folosite în cifru. Ordinea alfabetică a literelor din cuvântul cheie va da ordinea în care se vor citi coloanele.

Exemplu:

Cifrul simplu cu transpunere în coloane: textul în clar se scrie orizontal într-o anumită formă, ca la Polybius sau ceva asemănător, iar textul cifrat se citeşte pe coloane:

N	E	E	D
H	E	L	P
F	A	S	T

Atfel textul rezultat este: ”NHFEEAELSDPT”

O simplă transpoziție permite păstrarea proprietăților statistice ale textului în clar şi în textul cifrat; o nouă transpoziție a textului cifrat măreşte securitatea cifrului.

Spargerea unui cifru cu transpoziție începe cu verificarea dacă acesta este într-adevăr de acest tip prin calcularea frecvențelor literelor şi compararea acestora cu statisticile cunoscute. Dacă aceste valori coincid, se deduce că fiecare literă este "ea însăşi", deci este vorba de un cifru cu transpoziție.

Următorul pas este emiterea unei presupuneri în legătură cu numărul de coloane. Acesta se poate deduce pe baza unui cuvânt sau expresii ghicite ca făcând parte din text. Considerând sintagma "să prezinte", cu grupurile de litere (luate pe coloane) "si", "ăn", "pt", "re", se poate deduce numărul de litere care le separă, deci numărul de coloane. Notăm în continuare cu k acest număr de coloane.

Pentru a descoperi modul de ordonare a coloanelor, dacă k este mic, se pot considera toate posibilitățile de grupare a câte două coloane (în număr de $k(k\text{-}1)$). Se verifică dacă ele formează împreună un text corect numărând frecvențele literelor şi comparându-le cu cele statistice. Perechea cu cea mai bună potrivire se consideră corect poziționată. Apoi se încearcă, după acelaşi principiu, determinarea coloanei succesoare perechii din coloanele rămase iar apoi - a coloanei predecesoare. În urma acestor operații, există şanse mari ca textul să devină recognoscibil.

Unele proceduri de criptare acceptă blocuri de lungime fixă la intrare şi generează tot un bloc de lungime fixă. Aceste cifruri pot fi descrise complet prin lista care defineşte ordinea în care

caracterele vor fi trimise la ieşire (şirul poziţiilor din textul de intrare pentru fiecare caracter din succesiunea generată).

3.3 MAŞINI ROTOR

Având în vedere faptul ca metodele de substituţie şi permutări repetate sunt destul de complicate s-a încercat mecanizarea lor primele maşini fiind puse la punct în jurul anului 1920 acestea fiind bazate pe pricipiul de rotor.

O maşină rotor (rotor machine) are o tastatură şi o serie de rotoare ce permit implementarea unei versiuni a cifrului Vigénère. Fiecare rotor face o permutare arbitrară a alfabetului, are 26 de poziţii şi realizează o simplă substituţie. Deoarece rotoarele se mişcă cu viteze de rotaţie diferite, perioada unei maşini cu n rotoare este 26^n.

Cel mai celebru cifru bazat pe o maşină rotor este Enigma, utilizată de germani în cel de-al doilea război mondial. El a fost inventat de Arthur Scherbius şi Arvid Gerhard Damm în Europa şi a fost patentat în SUA. Germanii au îmbunătăţit considerabil proiectul inventatorilor săi, dar a fost spart de criptanaliştii polonezi care au explicat atacul englezilor.

4. Criptografie computationala cu cheie secreta

Cifrul DES: Standard de încriptare a datelor (Data Encryption Standard)

4.1.Scurt istoric al dezvoltării DES

Primele versiuni de DES au fost dezvoltate de IBM; în prezent, algoritmul este utilizat pe majoritatea platformelor UNIX, pentru criptarea parolelor. DES este autorizat şi de NBS (National Bureau of Standards-Biroul Naţional de Standarde) şi de NSA (National Security Agency-Agenţia Naţională de Securitate). De fapt începând cu 1977, DES a reprezentat metoda general acceptată pentru protejarea datelor confidenţiale. Figura următoare prezintă un scurt istoric al dezvoltării DES:

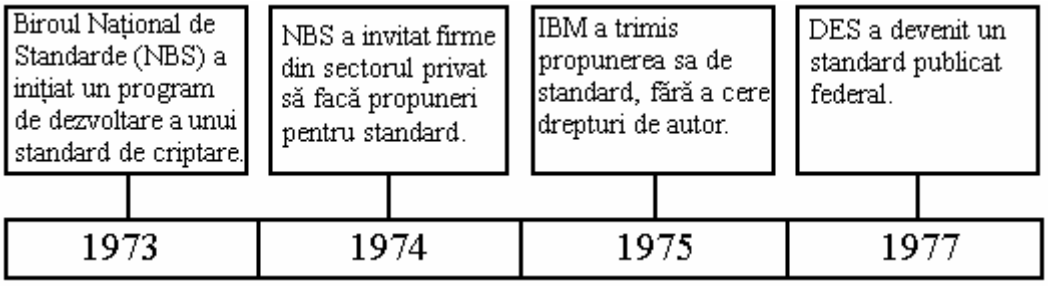


Fig.4.1. Scurt istoric al dezvoltării DES

DES a fost proiectat în principal pentru protejarea anumitor informaţii secrete ce puteau exista în birourile federale.

Datorită inexistenței unei tehnologii criptografice generale în afara domeniului securității naționale şi deoarece produse de securitate, cum ar fi cele de criptare, erau necesare în aplicațiile secrete folosite pe calculatoarele federale şi guvernamentale, NBS (National Bureau of Standards) a inițiat în 1973 un proiect pentru asigurarea securității sistemelor de calcul, proiect care cuprindea şi dezvoltarea unui standard pentru criptarea datelor din calculatoare.

4.2. Prezentarea algoritmului DES

DES oferă un algoritm implementabil în dispozitivele electronice pentru criptarea–decriptarea datelor. Ideea de "standardizare" în criptografie este revoluționară. Înaintea publicării lui DES, nu se cunoştea o altă publicație conținând descrierea unui algoritm criptografic în uz.

În general, cei mai mulți dintre proiectanții de criptosisteme au încercat să ascundă detaliile algoritmilor folosiți. DES constituie o excepție, algoritmul este publicat!. Aceasta poate constitui o veritabilă provocare pentru oricine încearcă spargerea acestuia.

Caracteristici:

- lungimea unui bloc este de 64 de biți;
- cheia este pe 64 de biți dintre care 8 sunt biți de paritate;
- flexibilitatea implementării şi utilizării în diferite aplicații;
- fiecare bloc cifrat este independent de celelalte;
- nu este necesară sincronizarea între operațiile de criptare/decriptare ale unui bloc;
- pentru creşterea securității se poate aplica algoritmul T-DES (triplu DES) care constă în iterarea de trei ori a algoritmului DES.

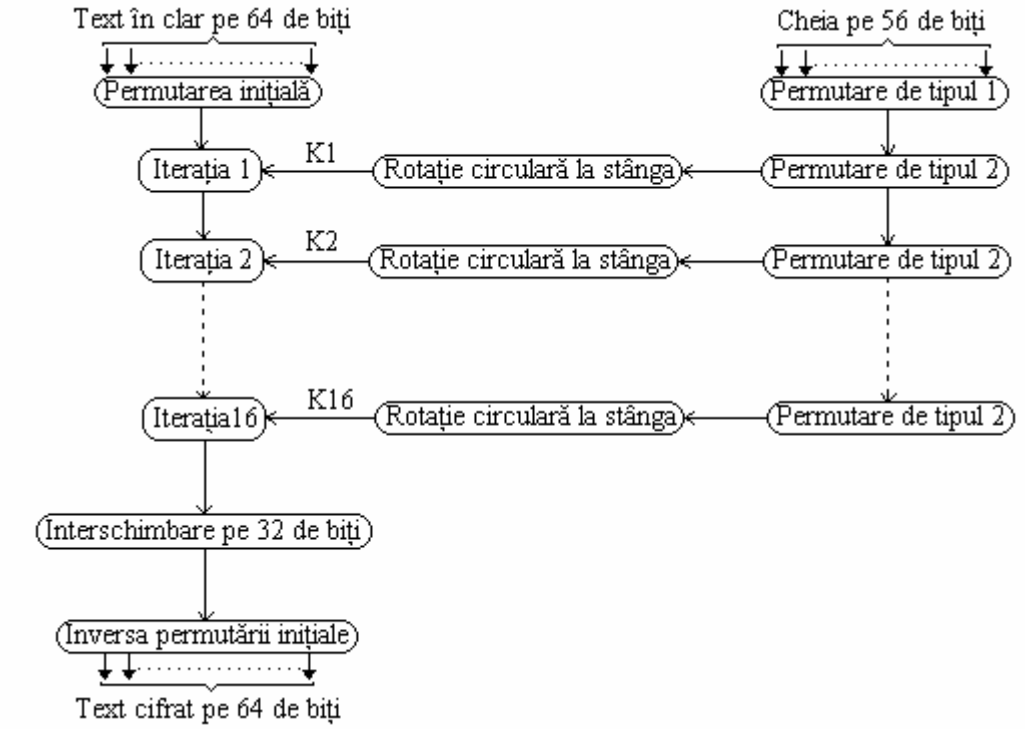


Fig.4.2. Prezentarea generală a algoritmului DES

În figura 4.2. se dă o schemă generală pentru algoritmul DES. Există două intrări în algoritm şi anume un bloc de text în clar de 64 de biţi şi o cheie de 56 de biţi.

Din figură se poate observa că în partea dreaptă a desenului avem generarea cheilor iar în partea stângă se prelucrează textul. Intrarea în bloc se va supune întâi unei permutări iniţiale după care au loc 16 iteraţii succesive, o interschimbare pe 32 de biţi şi în încheiere se va aplica inversa permutării iniţiale.

În cazul algoritmului DES, şi cheia de 56 de biţi va fi supusă unei permutări de tipul 1 după care va fi împărţită în două blocuri de 28 de biţi iniţial. După aceasta, asupra ambelor părţi astfel obţinute se va efectua o rotaţie circulară spre stânga cu un număr de biţi corespunzător numărului iteraţiei. Prezentarea unei singure iteraţii a algoritmului DES se dă în figura 4.3.

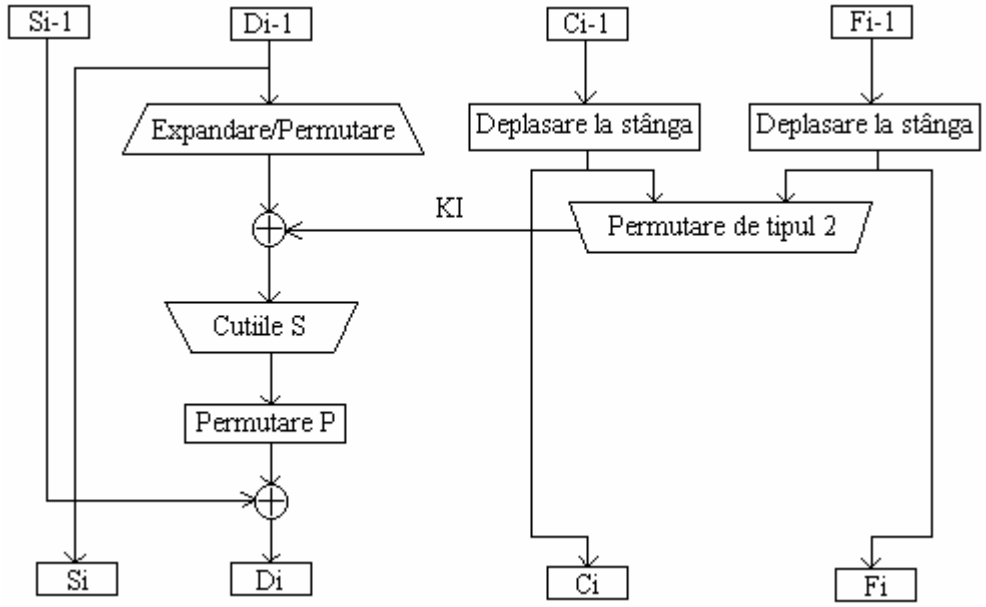


Fig.4.3 Descrierea unei singure iteraţii a algoritmului DES

Din figură se observă faptul că algoritmul împarte blocul de 64 de biţi în două blocuri de 32 de biţi pe care le denumeşte *stâng* şi *drept* şi cu care va lucra în continuare separat. Chiar din schemă se poate observa că blocul *stâng* care va fi regăsit la ieşirea dintr-o iteraţie va fi identic cu cel *drept* de la intrare deci $S_i = D_{i-1}$. Blocul iniţial *drept* va fi supus întâi unei permutaţii de expandare care îi va mări dimensiunea de la 32 la 48 de biţi pentru ca apoi să poată fi supus unei operaţii de SAU EXCLUSIV cheia intermediară corespunzătoare iteraţiei.

Rezultatul obţinut în urma acestor calcule va fi introdus în cutiile S de unde se va obţine un bloc de 32 de biţi care la rândul său va fi supus unei permutări P şi apoi unei operaţii de SAU EXCLUSIV cu blocul *stâng* iniţial obţinându-se blocul drept final al iteraţiei. Toate operaţiile de mai sus se pot rezuma în formula $D_i = S_{i-1} \otimes f(D_{i-1},S_i)$ unde modulul de calculare al funcţiei *f* se poate observa în figura 4.4:

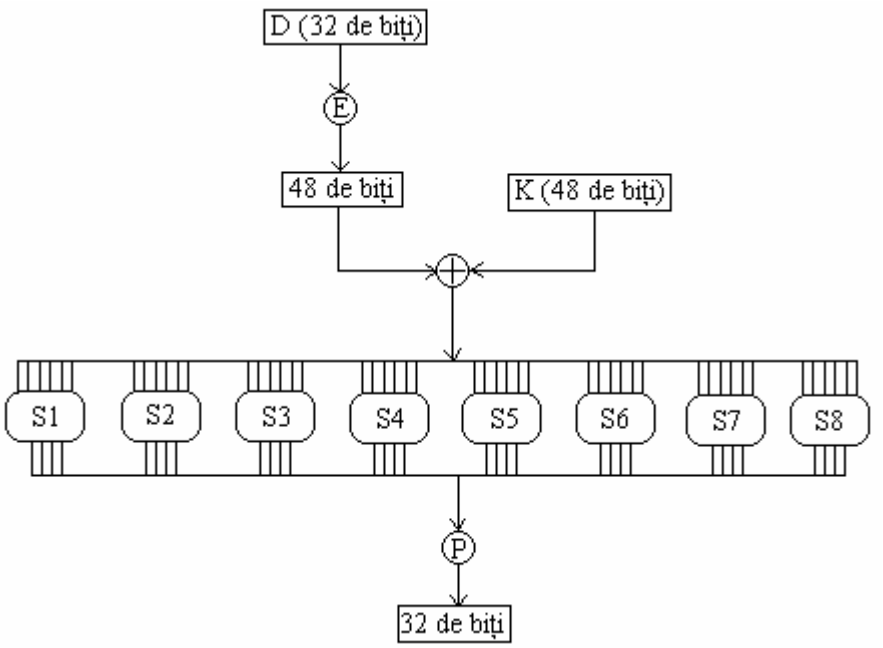


Fig.4.4. Operațiile care se efectuează pentru calcularea funcției *f*

O prezentare mai amănunțită a algoritmul DES este prezentată în continuare.

Operațiile de criptare şi decriptare, potrivit algoritmului DES, se desfăşoară după cum urmează. Mai întâi utilizatorul alege o cheie – o succesiune aleatoare de 56 biți – folosită atât la criptare, cât şi la decriptare, pe care, bineînțeles, o păstrează secretă. Opt biți, în pozițiile 8,16,....., 64, sunt adăugați la cheie pentru a asigura faptul că fiecare octet este de paritate impară. Aceasta foloseşte la detectarea erorilor în timpul stocării şi distribuirii cheilor. Astfel, biții adăugați sunt determinați de cei 56 biți inițiali, acum în pozițiile 1, 2, ..., 7, 9, ..., 15, ...,57, ..., 63 ale cheii. Apoi, cei 56 de biți sunt permutați în modul următor:

Bit ieşire	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Bit intrare	57	49	41	33	25	17	9	1	58	50	42	34	26	18
Bit ieşire	15	16	17	18	19	20	21	22	23	24	25	26	27	28
Bit intrare	10	2	59	51	43	35	27	19	11	3	60	52	44	36
Bit ieşire	29	30	31	32	33	34	35	36	37	38	39	40	41	42
Bit intrare	63	55	47	39	31	23	15	7	62	54	46	38	30	22
Bit ieşire	43	44	45	46	47	48	49	50	51	52	53	54	55	56
Bit intrare	14	6	61	53	45	37	29	21	13	5	28	20	12	4

Tabelul 4.1. - Permutarea de tipul 1

obținându-se două blocuri C_0 şi D_0 a câte 28 de biți fiecare. Prin urmare, primii trei biți ai lui C_0 (respectiv, ultimii trei ai lui D_0) sunt biții 57, 49, 41 (respectiv, 20, 12, 4) ai cheii. Având construite blocurile C_{n-1} şi $D_{n-1,}$ n=1,..., 16, putem construi blocurile C_n şi D_n, printr-o *deplasare la stânga* cu una sau două poziții, din blocurilor C_{n-1} şi D_{n-1}, în conformitate cu următorul tabel:

Iterația	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Nr. biți	1	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1

Tabelul 4.2. - Programul deplasărilor spre stânga pe paşi

O deplasare la stânga cu o poziție înseamnă că secvența 1, 2, ..., 28, devine 2, 3, 28,1. Deci, potrivit tabelului, C_6 şi D_6 sunt obținute din C_5 şi, respectiv, din D_5, printr-o deplasare la stânga cu două poziții. În acest moment se pot defini 16 selecții K_n, $1 \leq n \leq 16$ de biți din cheie. Fiecare K_n constă din 48 de biți, obținuti din biții lui C_nD_n, în următoarea ordine:

Bit ieşire	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Bit intrare	14	17	11	24	1	5	3	28	15	6	21	10	23	19	12	4
Bit ieşire	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
Bit intrare	26	8	16	7	27	20	13	2	41	52	31	37	47	55	30	40
Bit ieşire	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48
Bit intrare	51	45	33	48	44	49	39	56	34	53	46	42	50	36	29	32

Tabelul 4.3. - Permutarea de tipul 2

Primii (respectiv, ultimii) trei biții în K_n sunt biții 14, 17, 11 (respectiv, 36, 29, 32) din C_nD_n au fost omişi în K_n.

Calculele de până acum sunt preliminare: plecând de la cheie, am produs 16 secvențe K_n, constând fiecare din câte 48 de biți. Vom arăta acum cum se criptează un bloc *w* de 64 de biți din textul clar. Mai întâi blocul *w* este supus următoarei *permutări inițiale*:

Bit ieşire	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Bit intrare	58	50	42	34	26	18	10	2	60	52	44	36	28	20	12	4
Bit ieşire	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
Bit intrare	62	54	46	38	30	22	14	6	64	56	48	40	32	24	16	8
Bit ieşire	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48
Bit intrare	57	49	41	33	25	17	9	1	59	51	43	35	27	19	11	3
Bit ieşire	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64
Bit intrare	61	53	45	37	29	21	13	5	63	55	47	39	31	23	15	7

Tabelul 4.4. - Permutarea inițială

Bit ieşire	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Bit intrare	40	8	48	16	56	24	64	32	39	7	47	15	55	23	63	31
Bit ieşire	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
Bit intrare	38	6	46	14	54	22	62	30	37	5	45	13	53	21	61	29
Bit ieşire	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48
Bit intrare	36	4	44	12	52	20	60	28	35	3	43	11	51	19	59	27
Bit ieşire	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64
Bit intrare	34	2	42	10	50	18	58	26	33	1	41	9	49	17	57	25

Tabelul 4.5. - Inversa permutării inițiale

rezultând un nou cuvânt w', primii 3 biți ai acestuia fiind 58, 50, 42 ai lui w. Scriem acum $w'=L_0R_0$, unde L_0 şi R_0 sunt cuvinte de câte 32 de biți. Având definite L_{n-1} şi R_{n-1}, $1 \leq n \leq 16$, definim L_n şi R_n prin:

$$L_n = R_{n-1}$$

$$R_n = L_{n-1} \oplus f(R_{n-1}, K_n)$$

unde $\oplus$ semnifică adunarea bit cu bit modulo 2, iar *f* va fi definită mai jos. Criptarea *c* a blocului original *w* se va obține prin aplicarea inversei permutări inițiale, blocului $R_{16}L_{16}$ de câte 64 de biți.

Încă n-am definit funcția *f*, dar, înainte de a o face, să vedem cum se procedează pentru decriptare. Scriind ecuațiile de mai sus sub forma:

$$R_{n-1} = L_n$$

$$L_{n-1} = R_n \oplus f(L_n, K_n)$$

putem afirma că am indicat o cale de deducere recursivă a lui L_0 şi R_0 din L_{16} R_{16}, după care decriptarea este clară.

Funcția *f* realizează un bloc de 32 de biți din două blocuri, unul de 32 biți, R_{n-1} sau L_n, şi altul de 48 biți, K_n, în felul următor:

Bit ieşire	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Bit intrare	32	1	2	3	4	5	4	5	6	7	8	9	8	9	10	11
Bit ieşire	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
Bit intrare	12	13	12	13	14	15	16	17	16	17	18	19	20	21	20	21
Bit ieşire	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48
Bit intrare	22	23	24	25	24	25	26	27	28	29	28	29	30	31	32	1

Tabelul 4.6. - Permutarea cu expansiune

Aşadar, primul bit din blocul inițial de 32 biți apare de două ori în pozițiile 2 şi 48 în noul bloc de 48 biți. După această expandare, cele două blocuri a 48 biți sunt sumate bit cu bit modulo 2. Blocul rezultat B (48 biți) este împărțit în blocuri $B_1, B_2, \ldots, B_8$ a câte 6 biți fiecare. Fiecare din aceste 8 blocuri B_i este apoi transformat într-un nou bloc de 4 biți, B_i', utilizând tabele corespunzătoare S_I afişate mai jos.

Cutie	Rând	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
S1	0	14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
	1	0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
	2	4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
	3	15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13
S2	0	15	1	8	14	6	11	3	4	9	7	2	13	12	0	5	10
	1	3	13	4	7	15	2	8	14	12	0	1	10	6	9	11	5
	2	0	14	7	11	10	4	13	1	5	8	12	6	9	3	2	15
	3	13	8	10	1	3	15	4	2	11	6	7	12	0	5	14	9
S3	0	10	0	9	14	6	3	15	5	1	13	12	7	11	4	2	8
	1	13	7	0	9	3	4	6	10	2	8	5	14	12	11	15	1
	2	13	6	4	9	8	15	3	0	11	1	2	12	5	10	14	7
	3	1	10	13	0	6	9	8	7	4	15	14	3	11	5	2	12
S4	0	7	13	14	3	0	6	9	10	1	2	8	5	11	12	4	15
	1	13	8	11	5	6	15	0	3	4	7	2	12	1	10	14	9
	2	10	6	9	0	12	11	7	13	15	1	3	14	5	2	8	4
	3	3	15	0	6	10	1	13	8	9	4	5	11	12	7	2	14

S5	0	2	12	4	1	7	10	11	6	8	5	3	15	13	0	14	9
	1	14	11	2	12	4	7	13	1	5	0	15	10	3	9	8	6
	2	4	2	1	11	10	13	7	8	15	9	12	5	6	3	0	14
	3	11	8	12	7	1	14	2	13	6	15	0	9	10	4	5	3
S6	0	12	1	10	15	9	2	6	8	0	13	3	4	14	7	5	11
	1	10	15	4	2	7	12	9	5	6	1	13	14	0	11	3	8
	2	9	14	15	5	2	8	12	3	7	0	4	10	1	13	11	6
	3	4	3	2	12	9	5	15	10	11	14	1	7	6	0	8	13
S7	0	4	11	2	14	15	0	8	13	3	12	9	7	5	10	6	1
	1	13	0	11	7	4	9	1	10	14	3	5	12	2	15	8	6
	2	1	4	11	13	12	3	7	14	10	15	6	8	0	5	9	2
	3	6	11	13	8	1	4	10	7	9	5	0	15	14	2	3	12
S8	0	13	2	8	4	6	15	11	1	10	9	3	14	5	0	12	7
	1	1	15	13	8	10	3	7	4	12	5	6	11	0	14	9	2
	2	7	11	4	1	9	12	14	2	0	6	10	13	15	3	5	8
	3	2	1	14	7	4	10	8	13	15	12	9	0	3	5	6	11

Tabelul 4.7. – Cutiile S din algoritmul DES

Transformarea se face după cum urmează. Să luăm, de exemplu, blocul B_6=110010. Primul bit împreună cu ultimul constituie un număr x, $0 \leq x \leq 3$. Similar, restul de 4 biți constituie un alt număr y, $0 \leq y \leq 15$. În exemplul nostru, x=2 şi y=9. Dacă privim componentele perechii (x,y) ca indicii de linii şi coloane ai lui S_6, acestea vor determina în mod unic un număr, în cazul nostru 15. Reprezentând binar pe 15, obținem B_7'=1111. Valoarea lui *f* se obține acum prin aplicarea permutării:

Bit ieşire	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Bit intrare	16	7	20	21	29	12	28	17	1	15	23	26	5	18	31	10
Bit ieşire	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
Bit intrare	2	8	24	14	32	27	3	9	19	13	30	6	22	11	4	25

Tabelul 4.8. - Permutarea P

blocului de 32 biți rezultat B_1', B_2', B_3', B_4', B_5', B_6', B_7', B_8'. Cu aceasta definiția funcției f, ca şi descrierea algoritmului de criptare şi decriptare DES sunt încheiate.

Cu un hard adecvat algoritmul DES este foarte rapid. Pe de altă parte, criptanaliza lui conduce la numeroase sisteme de ecuații neliniare. Construirea unei maşini care să analizeze toate cele 2^{56} chei posibile, cu o viteză de 10^{12} chei pe secundă, ar necesita un echipament special, cu 10^6 chip-uri, care să cerceteze diferite porțiuni din spațiul cheii cu o viteză de o cheie pe micro-secundă. Estimările privind costul unei asemenea maşini variază considerabil.

Sistemul DES posedă şi o altă caracteristică foarte apreciată din punctul de vedere al siguranței lui: o mică schimbare în textul clar sau în cheie provoacă, o mare schimbare în criptotext.

5. Criptografia cu chei publice (asimetrică)

Într-un sistem cu chei simetrice, pentru un criptanalist care cunoaşte metoda de criptare nu există nici un fel de dificultăți în procesul de decriptare. Cheile de criptare şi decriptare coincid, chiar şi în cele mai sofisticate sisteme, cum ar fi DES-ul. Există criptosisteme în care metoda de criptare poate fi făcută publică. Aceasta înseamnă că şi criptanalistul cunoaşte metoda de criptare şi, totuşi, el nu este în măsură să decripteze criptotextul. În aceasta constă ceea ce este cunoscut sub numele de *criptografie cu chei publice*: metoda de criptare poate fi făcută publică.

Ideea a fost realizată de către Diffie şi Hellman, şi deşi revoluționară, ea este cât se poate de simplă. O astfel de idee simplă a fost luată în considerare târziu - la mijlocul anilor ’70 deoarece teoria complexității calculului s-a dezvoltat abia în ultima perioadă de timp. Teoria dă informații privind complexitatea diferitelor calcule, de pildă privind timpul de calcul necesar atunci când se folosesc cele mai bune calculatoare. O astfel de informație este crucială în criptografie.

Deşi metoda de criptare este făcută publică, criptarea se poate face în siguranță. Dacă totuşi unui criptanalist îi vor fi necesari sute de ani pentru a deduce metoda de decriptare prin dezvăluirea metodei de criptare, în această situație nimic nu va fi compromis.

Criptarea se poate face într-o singură direcție (“sens unic”). Deşi este uşor de mers în această direcție, totuşi este imposibil de urmat direcția opusă: decriptarea. Pentru exemplificare să considerăm cartea de telefon dintr-un mare oraş. Este foarte uşor să stabilim numărul de telefon al unei persoane precizate. Pe de altă parte, este foarte greu (chiar imposibil) de stabilit persoana care are un număr de telefon specificat.

Acest exemplu oferă şi o idee pentru un criptosistem cu chei publice. Criptarea este de tip context-free: literă cu literă. Pentru fiecare literă a textului clar este ales, din cartea de telefon, în mod aleator, un nume care începe cu acea literă. Numărul de telefon corespunzător constituie criptarea apariției respective a literei în cauză. Spre exemplificare să criptăm textul ACUMSTAU.

Tabelul 513. - Criptarea unui text

Text clar	Nume ales	Text criptat
A	Asiminei	7134267
C	Cazacu	3214327
U	Ursanu	2653598
M	Moldovan	9767121
S	Scutaru	4002132
T	Tudor	2147877
A	Aliuță	1671873
U	Ursachi	2355510

Criptotextul se obține deci scriind, unul după altul, toate numerele care apar în coloana din dreapta. Un receptor legal al mesajului trebuie să aibă o carte de telefon, cu numere aranjate în ordine crescătoare. O astfel de listă de numere face ca decriptarea să fie mai simplă. Lista numerelor de telefon ordonate crescător constituie *trapa secretă*, cunoscută numai de utilizatorii legali ai acestui sistem.

Fără cunoaşterea acestei trape, adică fără a poseda o copie a cărții de telefon, criptanalistul va avea nevoie de un timp foarte mare pentru decriptare. Aceasta în ciuda faptului că metoda de criptare a fost făcută publică şi astfel criptanalistul cunoaşte, în principiu, cum trebuie să intercepteze secvența numerică interceptată.

Căutarea exhaustivă este în mod cert mult prea lungă în timp. Criptanalistul poate, de asemenea, să apeleze numerele din criptotext şi să ceară numele corespunzătoare acestora. Succesul metodei este cel puțin discutabil în foarte multe cazuri, criptanalistul poate primi un răspuns curios sau nici un răspuns. De fapt metoda devine de neaplicat dacă se foloseşte o carte de telefon mai veche.

Principiul acestui sistem este ilustrat în figura 5.1:

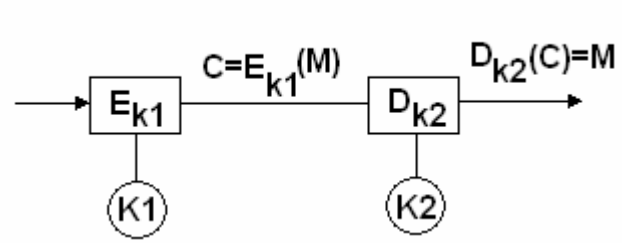


Fig. 3.1 Schema bloc a unui sistem de criptare cu cheie publică

Fiecare utilizator are o transformare de cifrare publică E_{k1} care poate fi memorată într-un fişier public şi o transformare de cifrare critică D_{k2}.

Proprietățile unui sistem de criptare cu cheie publică sunt:

1. pentru orice pereche de chei (K_1,K_2) algoritmul de descifrare cu cheia K_2 : D_{k2} este inversul algoritmului de descifrare cu cheia K_1 : E_{k1};

2. pentru orice K_1, K_2 şi orice M, algoritmii de calcul pentru E_{k1} şi D_{k2} sunt simpli şi rapizi

3. pentru orice K_1,K_2 algoritmul de calcul al lui D_{k2} nu poate fi obținut într-un interval de timp rezonabil plecând de la E_{k1}

4. orice pereche K_1,K_2 trebuie să fie calculabilă uşor plecând de la o cheie unică şi secretă.

În sistemele cu cheie publică, protecția şi autentificarea sunt realizate prin transformări distincte.

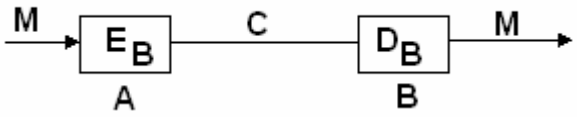


Fig. 5.2 Protecția criptosistemului cu chei publice

Presupunem că utilizatorul A doreşte să transmită un mesaj M unui utilizator B. În acest caz, A, cunoscând cheia publică a lui B (E_B), transmite criptograma $C=E_B(M)$, asigurând astfel funcția de protecție (Fig. 5.2).

La recepție, B descifrează criptograma C utilizând transformarea D_B:

$D_B(C)=D_B(E_B(M))=M$.

Pentru autentificare se aplică lui M transformarea secretă D_A. Către B se va transmite: $C=D_A(M)$. La recepție, B va aplica transformarea publică E_A corespunzătoare lui A: $E_A(C)=E_A(D_A(M)=M$.

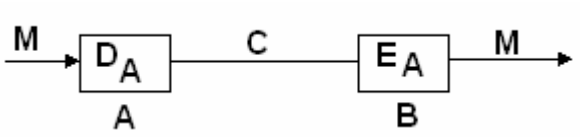


Fig. 3.3 Autentificarea în criptosisteme cu chei publice

Autentificarea este realizată deoarece către B nu pot fi transmise mesaje false $C'=D_A(M')$, deoarece numai A cunoaşte D_A (cheia secretă). În acest caz nu se realizează însă şi protecția, pentru că M poate fi obținut de orice apelant E_A a lui C, E_A fiind publică.

Pentru a realiza simultan protecția şi autentificarea (Fig. 5.4), spațiul M trebuie să fie echivalent spațiului C, astfel încât orice pereche (E_A,D_A), (E_B,D_B) să fie mutual inverse:
$E_A(D_A(M))=D_A(E_A(M))=M$
$E_B(D_B(M))=D_B(E_B(M))=M$

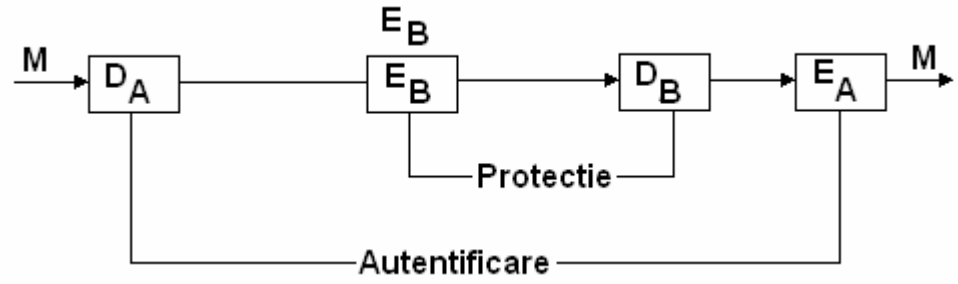


Fig. 3.4 Protecție şi autentificare în criptosistemele cu chei publice

Utilizatorul A va aplica mai întâi transformarea secretă D_A asupra mesajului M, după care va transmite lui B criptograma : $C=E_B(D_A(M))$. Receptorul B obține mesajul în clar aplicând criptogramei propria-i funcție de descifrare D_B şi apoi transformare publică a lui A,
E_A: $E_A(D_B(C)=E_A(D_B(E_B(D_A(M))))=E_A(D_A(M))=M$.

Semnătura digitală în criptosistemele cu chei publice

Fie mesajul semnat de A, transmis către receptorul B. Semnătura lui A trebuie să aibă următoarele proprietăți:

- B trebuie să fie capabil să valideze semnătura lui A;
- să fie imposibil pentru oricine, inclusiv B, să falsifice semnătura lui A;
- în cazul în care A nu recunoaşte semnătura unui mesaj M, să existe un „judecător” care să rezolve disputa dintre A şi B.

Implementarea semnăturii digitale este extrem de simplă în cazul sistemelor cu chei publice. În acest caz, D_A poate servi ca semnătura digitală pentru A. Receptorul B al mesajului M, semnat de A, este sigur atât de autenticitatea emiţătorului cât şi a datelor.

Protocolul semnăturii digitale se desfăşoară astfel:

- A semnează pe M : $S=D_A(M)$;
- A trimite lui B criptograma : $C=E_B(S)$;
- B validează semnătura lui A, verificând dacă $E_A(S)=M$;

$D_B(C)=D_B(E_B(S))=S$;

$E_A(S)=E_A(D_A(M))=M$;

Un „judecător” rezolvă eventualele dispute dintre A şi B, controlând dacă $E_A(S)$ conduce la M, în acelaşi mod ca şi B.

5.2 Sisteme de cifrare cu chei publice exponențiale

Sistemele criptografice cu chei publice sunt sisteme de tip asimetric.Ideea care stă la baza acestui concept constă în faptul că, cheia de cifrare este făcută publică de fiecare utilizator şi poate fi folosită de către toți ceilalți utilizatori pentru cifrarea mesajelor ce îi sunt adresate. În schimb, procedura (cheia) de descifrare este ținută secretă. În Fig. 3.5 se prezintă sistemul cu chei publice, în paralel cu un sistem clasic.

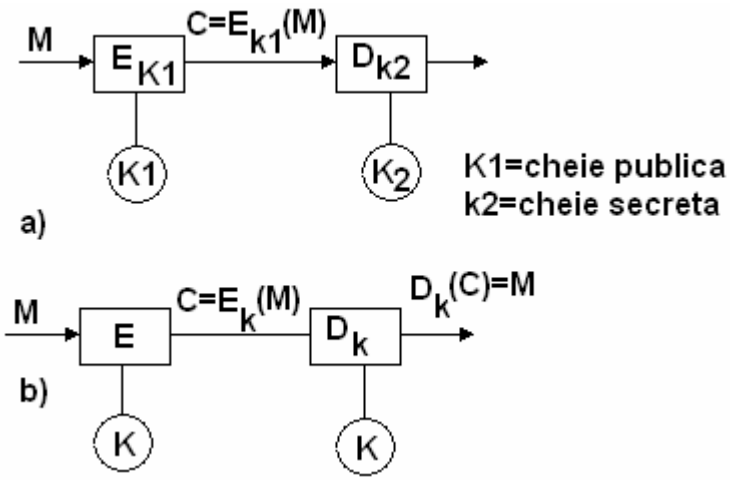


Fig. 5.5 Criptosistem cu chei publice a) şi convențional b)

Dacă notăm cu {m} –mulțimea mesajelor, {c} –mulțimea receptoarelor,
E-procedura de cifrare şi D-procedura de descifrare, un criptosistem cu chei publice trebuie să satisfacă următoarele:
(1) dacă C=E(M), atunci M=D(C) sau D(E(M))=M, $\forall M \in \{M\}$;
(2) E şi D sunt uşor şi rapid aplicabile
(3) dezvăluirea publică a lui E nu trebuie să compromită pe D, ceea ce înseamnă că obținerea lui D din E este matematic imposibil sau presupune un consum enorm de resurse.

Metoda propusă permite comunicații sigure între utilizatorii care nu au stabilit contacte prealabile. De exemplu, dacă utilizatorul A doreşte să transmită un mesaj confidențial utilizatorului B, A va căuta în fişierul public E_B şi se va transmite la B: $C=E_B(M)$. Conform proprietății a treia, B este singurul utilizator care ştie să descifreze criptograma C aplicând cheia D_B ținută secretă.

În plus, pentru ridicarea gradului de securitate al transmisiei s-a propus ca E şi D să îndeplinească şi următoarea proprietate adițională:

(4) dacă S=D(M), atunci M=E(S) sau E(D(M))=M pentru $\forall M \in \{M\}$.

Varianta S este numită semnătură digitală şi reprezintă o metodă de autentificare reciprocă. În timp ce B poate fi sigur că mesajul recepționat a venit de la adevăratul A, A poate fi sigur că nimeni nu va putea să-i atribuie un mesaj fals. Utilizatorul A poate semna mesajul către B astfel: $S=D_A(M)$, apoi, trimițând criptograma: $C=E_B(S)$.

În aceste condiții numai B poate recunoaşte pe S din C calculând:

$D_B(C)=D_B(E_B(S))=S$. Apoi mesajul se poate obține calculând :

$E_A(S)=E_A(D_A(M))=M$.

Diffie şi Hellman sugerează o metodă de implementare practică a conceptului propus. Se indică utilizarea unor funcții greu inversabile. O funcție este greu inversabilă dacă este inversabilă şi uşor de calculat, dar pentru aproape toate valorile y din codomeniu este imposibil computațional să se calculeze

$$x=f^{-1}(y).$$

Cu alte cuvinte este imposibil să se calculeze f^{-1} dacă se dispune de o descriere completă a lui f. În concluzie, o funcție este greu inversabilă dacă:

- este uşor să se calculeze y din x, y=f(x);
- există inversa funcției;
- este computațional imposibilă determinarea inversei funcției.

O funcție greu inversabilă se spune că este cu trapă atunci când f^{-1} este uşor de calculat numai dacă se dispune de o informație numită **trapă**; necunoaşterea acestei informații face ca funcția să fie greu inversabilă. O astfel de pereche de funcții (f, f^{-1}) poate constitui perechea (E, D) a unui criptosistem cu chei publice. În general pentru procedurile E şi D se indică scheme bazate pe operații modulo n cu elemente din inelul claselor de resturi modulo n. Securitatea algoritmilor implementați prin metoda cifrării asimetrice au la bază dificultatea calculului logaritmului modulo număr prim.

Fie q un număr prim şi un întreg X, $X \in [1, q^{-1}]$. Se pot calcula:

$Y=a^x \pmod q$, unde a este un element primitiv al câmpului Galois GF(q).

După cum se ştie, clasele de resturi modulo q formează un inel; dacă q este un număr prim acesta formează un câmp Galois GF(q). Într-un câmp Galois GF(q) există (q^{-1}) numere care se

numesc elemente primitive ale câmpului. Dacă $a,a^2,...,a^{\Phi(q)}$ sunt puterile lui a, acestea au ca resturi modulo q pe 1,2,...,Φ(q), unde Φ(q) este indicatorul lui Euler, $\Phi(q)=q^{-1}$.

Fiecare utilizator A alege în mod aleator un număr X_A, $X_A=\{1,2,...,q^{-1}\}$ şi se calculează $Y_A=a^{XA}(\text{mod } q)$. Numărul X_A este ţinut secret în timp ce Y_A se face public. Dacă utilizatorii A şi B doresc să comunice, ei utilizează următoarea cheie de comunicaţie:

$K_{AB}=Y_A^{XB}=Y_B^{XA}=Y_B^{XAXB} \pmod q$.

În timp ce utilizatorii A şi B pot calcula cheia K_{AB} pornind de la x propriu(secret) şi y public al partenerului, un criptanalist trebuie să calculeze K_{AB} pornind de la Y_A şi Y_B, singurele făcute publice; se procedează astfel: $K_{AB}=Y_A^{\log YB}(\text{mod } q)$. Acest lucru face ca sistemul să fie deosebit de greu de spart datorită imposibilităţii calculului logaritmului modulo q.

5.2.1. Cifrul Rivest-Shamir Adleman (RSA)

Cel mai larg utilizat şi verificat criptosistem cu chei publice a fost cel introdus de Rivest, Shamir şi Adleman, care este acum cunoscut ca RSA.

Acest cifru reprezintă standardul în domeniul semnăturilor digitale şi al confidenţialităţii cu chei publice. Sub diferite forme de implementare, prin programe sau dispozitive hardware speciale, RSA este astăzi recunoscută ca cea mai sigură metodă de cifrare şi autentificare disponibilă comercial.

Acest cifru se bazează pe o idee uimitor de simplă din teoria numerelor şi totuşi a rezistat la toate atacurile criptanaliştilor de până acum.Ideea se bazează pe faptul că, deşi este uşor să înmulţeşti două numere prime mari, este însă extrem de greu să se factorizeze produsul lor.Astfel, produsul poate fi făcut cunoscut şi utilizat ca o cheie de criptare.Numerele prime însele nu pot fi reconstituite din produsul lor.Pe de altă parte, numerele prime sunt necesare pentru decriptare.

În continuare voi prezenta detalii despre sistemul RSA.Fie p şi q două numere prime mari distincte şi aleatoare(având aproximativ 100 cifre zecimale fiecare).Se notează :

$n = pq$ şi $\Phi(n) = (p-1)(q-1)$

unde Φ este funcţia Euler.Se alege un număr aleator mare $D>1$ astfel încât $(D,\Phi(n)) = 1$ şi se calculează numărul $E, 1<E<\Phi(n)$ care să satisfacă congruenţa:

$ED = 1(\text{mod}\Phi(n))$.

Numărul n este modulul, E este exponentul de criptare şi D exponentul de decriptare.Numerele n şi E formează cheia publică de criptare, iar p, q, Φ(n) şi D formează trapa secretă.Evident că informaţia despre trapa secretă nu constă în patru parametri independenţi.De exemplu, cunoaşterea lui p conduce imediat la aflarea şi a celorlalţi trei parametri.

Metoda de criptare implică calcul exponențial într-un câmp finit(modulo n). Mesajul cifrat se obține din mesajul în clar printr-o transformare(codare) bloc. Fie unul din aceste blocuri din mesaj M, bloc ce are proprietatea $M \in (0, n-1)$; proprietate care se obține prin modul de împărțire a mesajului în blocuri. Blocul cifrat, C, corespunzător blocului în clar, se obține calculând exponențiala: $C=M^E(\text{mod } n)$, E şi n reprezentând astfel cheia secretă de decriptare.

Decriptarea se face prin operația : $M=C^D(\text{mod } n)$, unde D este cheia secretă de decriptare.

Cele două chei E şi D trebuie să satisfacă relația:

$M=C^D(\text{mod } n)=M^{ED}(\text{mod } n)$, pentru ca algoritmul să poată fi într-adevăr folosit. Pentru aceasta plecăm de la teorema Euler-Fermat: P este un număr prim dacă $a^{p-1}=1(\text{mod } p)$ oricare ar fi a, $a \in [1,p]$. Astfel, dacă am ales un număr prim n, pentru orice bloc $n \in (0,1)$ avem proprietatea de la care pornim $M^{\Phi(n)}(\text{mod } n)=1$,

Dacă E şi D satisfac relația $ED(\text{mod } \Phi(n))=1$, putem scrie:

$ED=K\Phi(n)+1=\Phi(n)+\Phi(n)+\Phi(n)+\ldots+\Phi(n)+1$, $M^{ED}=M^{\Phi(n)+\Phi(n)+\Phi(n)+\ldots+\Phi(n)+1}=$

$M^{\Phi(n)}M^{\Phi(n)}M^{\Phi(n)}...M(\text{mod } n)$. Deci $ED(\text{mod } \Phi(n))=1 \Rightarrow M^{ED}=M(\text{mod } n)$.

Astfel am asigurat o transformare reversibilă de criptare pe baza unei exponențiale într-un câmp finit. Mai rămâne să asigurăm securitatea cheii de criptare, iar în cazul de mai sus este uşor de determinat având la dispoziție E şi n, ştiind că $ED(\text{mod } \Phi(n))=1$ şi $\Phi(n)=n-1$.

Securitatea se bazează pe ideea de factorizare a unui număr mare. Pornind de la această idee numărul n se poate obține prin produsul a două numere prime mari p şi q: $n=pq$, astfel încât indicatorul lui Euler, $\Phi(n)=(p-1)(q-1)$, devine mult mai greu de determinat având la dispoziție n.

Folosind această schemă se poate obține un sistem performant de criptare cu chei publice. Un sistem care asigură confidențialitatea va avea ca elemente următoarele perechi:

- (E, n) cheia publică;
- (D, n) cheia secretă;

Un criptanalist care are la dispoziție perechea (E, n) va trebui să determine D ținând cont că $ED(\text{mod } \Phi(n))=1$. Pentru aceasta trebuie determinat $\Phi(n)=(p-1)(q-1)$, deci implicit p şi q, problemă care se reduce la a factoriza numărul n, problemă practic imposibilă pentru un număr n mare.

Exemplu

Vom utiliza un cifru cu p=47 şi q=79

$n=pq=47\cdot 79=3713$.

Alegând D=47, E va fi 37 pentru a satisface relația:

$E\cdot D \bmod((p-1)(q-1))=1$, $E=[(p-1)(q-1)+1]/D$

Astfel, pentru a coda mesajul „A sosit timpul” vom coda mai întâi fiecare literă a alfabetului. De exemplu: A=00, B=01,...,Y=25, blanc=26.

Mesajul va deveni: M=0018 1418 0819 1908 1215 2011. În continuare vom coda fiecare număr de 4 :

$0018^E \bmod(n) = 0018^{37} \bmod (3713) = 3091$

$1418^E \bmod(n) = 1418^{37} \bmod (3711) = 0943$

$0819^E \bmod(n) = 0819^{37} \bmod (3713) = 3366$

$1908^E \bmod(n) = 1908^{37} \bmod (3713) = 2545$

$1215^E \bmod(n) = 1215^{37} \bmod (3713) = 0107$

$2011^E \bmod(n) = 2011^{37} \bmod (3713) = 2965$

Astfel mesajul cifrat devine: 3091 0943 3366 2545 0107 2965.

La decriptare se vor calcula pe rând:

$3091^E \bmod(n) = 3091^{37} \bmod(3713) = 0018$

$0943^E \bmod(n) = 0819^{37} \bmod(3713) = 1418$,

… , obținând mesajul inițial.

O problemă care apare în dezvoltarea unui astfel de algoritm este cea a calculului valorilor sistemului: a numărului n şi a celor două chei E şi D, calcul care se va face la nivel de zeci de digiți pentru a asigura un nivel de securizare mare. Se poate spune ca lucrând cu operanzi pe 512 biți sistemul este deocamdată imposibil de spart.

5.2.2. Cifrul EL GAMAL (EG)

El Gamal propune în o nouă metodă de semnătură bazată pe schema de distribuție a cheilor a lui Diffie şi Hellman. Schema presupune că A şi B doreau stabilirea unei chei secrete k_{AB}, A folosind informația secretă x_A iar B pe x_B. De asemenea există un întreg prim mare p şi un element a primitiv modulo p. A va calcula $y_A=a^{xA} \bmod p$ şi va emite la B pe y_A. Similar B va emite la A: $y_B=a^{xB} \bmod p$.

Ca urmare fiecare poate acum calcula cheia secretă comună:

$k_{AB}=(a^{xA})^{xB} \bmod p=(y_A)^{xB} \bmod p=(a^{xB})^{xA} \bmod p=(y_B)^{xA} \bmod p$.

Pentru ca A să transmită un mesaj m cifrat la B, $0 \le m \le p-1$, A va putea alege un $k \in [0, p-1]$ care va servi drept cheie secretă x_A. El va calcula apoi cheia: $k_{AB}=y_B^k \bmod p$, unde y_B a fost obținut direct de la B sau dintr-un fişier cu chei publice. A va pute emite la B perechea (c_1, c_2), unde: $c_1=a^k \bmod p$, şi

$c_2=k_{AB}\, m \bmod p$.

Descifrarea se face şi ea în două etape. Mai întâi se obține k_{AB}:

$k_{AB}=(a^k)^{xB}=(c_1)^{xB} \bmod p$.

Apoi se va obține mesajul clar prin împărțirea lui c_2 la k_{AB}. El Gamal propune pe baza acestei scheme o nouă metodă de semnătură. Fie m un document semnat, $0 \le m \le p-1$. Fişierul public va conține cheia $y=a^x \bmod p$ pentru fiecare utilizator. Pentru a semna un document, A va folosi cheia

sa secretă x_A într-o astfel de manieră încât semnătura să poate fi verificată de orice alt utilizator pe baza cheii publice y_A.

1) Procedura de semnare

a) Se alege aleator un întreg k, $k \in [0, p-1]$, astfel ca c.m.m.d.c.(k, p-1) = 1;

b) Se calculează: $r=a^k \bmod p$.

Semnătura lui m este perechea (r,s), $0 \leq r, s \leq p-1$, care satisface condiția: $m=y^r r^s \pmod p$ (*);

c) acum condiția (*) poate fi scrisă: $a^m=a^{xr}a^{ks} \bmod p$, prin rezolvarea ecuației $m=xr+ks \bmod p$, care are soluție dacă se respectă procedura a).

2) Verificarea semnăturii

Fiind dați m,r,s, este uşor să se verifice autenticitatea semnăturii calculând $a^m \pmod p$, $y^r r^s \pmod p$ şi verificând apoi dacă sunt egale. Este indicat ca valoarea lui k, aleasă aleator, să fie folosită o singură dată. Se poate folosi în acest scop un generator de k bazat pe un model **DES** (generator cu contor).

Generalizare a schemei El Gamal:

- cheia publică: $E_A=a^{D_A} \pmod n$, unde a = este o constantă a sistemului cunoscută de către toți parametrii; n = este un număr prim;
- cheia secretă: D_A, un număr natural.

(1) Semnarea unui mesaj M se face după următorul algoritm:

pentru i=1,p-1 *execută*

generez aleator k_i în [0, n-1] astfel încât

c.m.m.d.c. $(k_i, n-1)=1$

calculez $a_i=a^{ki} \pmod n$ din ecuația:

$M=D_A a_1+k_1 a_2+...+k_{p-1}a_p \pmod{n-1}$

sfârşit

Ca urmare semnătura lui M realizată cu nivelul de securitate p este reprezentată de: $S=(a_1,a_2,...,a_p)$.

(2) Verificarea semnăturii se face calculând separat:

Valoarea 1 = $a^M \bmod n$

şi

Valoarea 2 = $e_A{}^{a1}a_1{}^{a2}...a_{p-1}{}^{ap} \bmod n$

şi comparându-le dacă sunt egale.

Această schemă introduce un factor de complexitate suplimentară prin nivelul p de securitate, ceea ce sporeşte rezistența schemei la atac criptanalitic. Criptanalistul este confruntat cu necesitatea inversării tuturor funcțiilor exponențiale $a_i=a^{ki} \pmod n$ în câmp finit, ceea ce reprezintă o problemă grea computațională dacă n este mare (128,256,512 biți).

5.2.3. Sisteme de cifrare cu chei publice de tip rucsac

Metoda de cifrare cu cheie publică **MH** este bazată pe cunoscuta problemă a rucsacului, care constă în a determina într-o mulțime de numere întregi, o submulțime de o sumă dată. Merkle şi Hellman propun o metodă a cărei securitate depinde de dificultatea rezolvării următoarei probleme: fiind dat un întreg pozitiv C şi un vector $A=(a_1,a_2,...,a_n)$ de întregi pozitivi, să se găsească o submulțime a lui A a cărei sumă să fie C. Cu alte cuvinte este necesar să se determine un vector binar $M=(m_1,...,m_n)$, astfel că C=AM sau: $C = \sum a_i m_i$.

Intuitiv problema este următoarea: cunoscând greutatea unui rucsac închis şi greutățile mai multor obiecte care s-ar putea afla în interiorul său, să se determine setul de obiecte aflate în rucsac, fără a se face deschiderea lui.

Cel mai bun algoritm cunoscut pentru rezolvarea ei – în cazul în care dimensiunea rucsacului este n – cere $(2^{n/2})$ timp şi o memorie de $(2^{n/2})$.

Există totuşi o clasă specială de probleme de un asemenea tip, numită „rucsac simplu”, ce pot fi rezolvate într-un timp ce depinde liniar de n.

Într-un rucsac simplu elementele a_i (i=1,...,n) sunt în relație de dominanță, adică $a_i > \sum a_j$, j=1,...,i.

Relația de dominanță simplifică soluția rucsacului, ea putându-se face după următoarea procedură:

Procedura RUCSAC-SIMPLU (C,A,M)

Pentru j=N,1 *execută*

Dacă $c \geq a_j$ *atunci* $m_j=1$

altfel $m_j=0$

atribuie $c=c-a_j \cdot m_j$

dacă c=0 *atunci* „Soluția este M”

altfel „Nu există soluție”

sfârşit.

În [MERK 78] se prezintă 2 variante ale metodei, ambele utilizabile doar în secretizare, nu şi în autentificare.

1) Algoritmul MH cu trapă aditivă

În proiectarea unui astfel de criptosistem trebuie convertit mesajul simplu într-un mesaj cu trapă (trapdoor knapsack), care este greu de rezolvat. Mai întâi se selectează un vector rucsac simplu $A'=(a_1',a_2',...,a_n')$. Acesta permite o soluție simplă a problemei C′=A′M.

Apoi se alege un întreg n astfel ca $n>2a_n' > \sum a_i'$, i=1,n.

În continuare se alege un alt întreg w, astfel ca c.m.m.d.c (n,w)=1 şi se calculează inversa lui w mod n. În final vectorul A´ este transformat într-un vector „rucsac greu”:

$A=w\cdot A' \bmod n$

$a_i=w\cdot a_i' \bmod n$

Acum rezolvarea problemei C=A·M este dificilă, dacă nu se dispune de o informație „trapă” (inversa lui w şi n), cu ajutorul căreia calculul se simplifică:

$C'=W^{-1}\cdot C \bmod n=W^{-1}\cdot A\cdot M \bmod n=W^{-1}\cdot(W\cdot A')\cdot M \bmod n=A'\cdot M \bmod n=A'\cdot M$

Transformarea de cifrare E_A (publică) foloseşte cheia publică care este vectorul „rucsac greu”, A. Se obține criptograma:

$C=E_A(M)=A\cdot M.$

Transformarea de descifrare D_A foloseşte cheia secretă (A',n,W^{-1}) calculând pe baza funcției „rucsac simplu”:

$D_A(C)=(\text{rucsac\_simplu})\ (W^{-1}C \bmod n, A')\ (W^{-1}C \bmod n, A')=M.$

Exemplu. Fie A′=(1,3,5,10) vectorul rucsac simplu, n=20, w=7. Inversul multiplicativ al lui w este w′=3.

Acest rucsac simplu este transformat într-unul cu trapă A:

A=(7·1 mod 20, 7·3 mod 20, 7·5 mod 20, 7·10 mod 20)=(7,1,15,20).

Considerăm următorul mesaj M care va fi cifrat:

$(M)_{10}=13$ deci $(M)_2=(1101)$.

Criptograma C este obținută cu ajutorul vectorului trapă A:

$C=E_A(M)=7+1+10=18$

C=(10010).

La descifrare se obține mesajul clar cu vectorul simplu A′ care este secret:

$M=D_A(C)=D_A(18)=RUCSAC\_SIMPLU(3\cdot 18 \bmod 20, A')=$

$= RUCSAC\_SIMPLU(14,A')=(1101)=13.$

Cifrul **Merkle Hellman** este folosit în sistemul de operare UNIX pentru protecția confidențialității scrisorilor în comenzile enroll, xget şi xsend.

2) Algoritmul MH cu trapă multiplicativă

Un rucsac cu trapă multiplicativă este uşor de rezolvat dacă elementele vectorului A sunt relativ prime. Fiind date A=(6,11,35,43,169) şi M=2838, este uşor de determinat că M=6·11·43, deoarece 6,11 şi 43 divid pe M, în timp ce 35 şi 169 nu. Un rucsac multiplicativ este transformat într-unul aditiv prin folosirea logaritmilor. Pentru ca ambii membri să aibă valori rezonabile, logaritmii sunt luați peste câmpul Galois GF(n), unde n este un număr prim.

Exemplu. Fie m=4 (numărul de componente ale rucsacului), n=257, A'=(2,3,5,7) şi b=131 – baza logaritmului. Informațiile secrete sunt n,A' şi b. Rezolvând ecuațiile:

$131^{a1}=2 (\text{mod } 257)$

131^{a2}=3(mod 257)

131^{a3}=5(mod 257)

131^{a4}=7(mod 257)

rezultă a_1=80, a_2=183, a_3=81, a_4=195 şi deci A=(80,183,81,195) este vectorul făcut public.

Găsirea logaritmilor peste GF(n) este relativ uşoară dacă (n-1) are numai factori primi mici (aceasta înseamnă valori de ordinul a 10^6 până la 10^{12}).

Presupunem că avem dată criptograma C=183+81=264 şi vrem să găsim soluția C=A·X. Cunoscând informația trapă n, A' şi b putem calcula c'=b^c(mod n).În exemplul C'=131^{264} (mod 257)=15=$(2^0)(3^1)(5^1)(7^0)$. Rezultă că X=(0,1,1,0), deoarece $b^c = b(\sum a_i x_i) = \prod a_i^{`xi}$ (mod n). În acest caz este necesar ca $\prod a_i$ <n pentru a se asigura că $\prod a_i^{`xi}$ (mod n) este egală cu $\prod a_i^{`xi}$ în mulțimea numerelor întregi.

O variantă este să se utilizeze primele n numere prime ca a_i', caz în care la m=100, n are lungimea de 730 de biți. Orice variantă aleasă trebuie să facă un compromis între securitatea şi extensia datelor.

5. 2.4.Avantajele evidente ale cheilor publice

Avantajele criptografiei cu chei publice sunt extraordinare, cu condiția ca aceasta să poată fi realizată practic fără nici un fel de urmări secundare negative. O inovație adusă de cheile publice priveşte *gestiunea cheilor*: cum se pot mânui şi transmite chei. Să considerăm un criptosistem clasic (simetric). Cheia de criptare o furnizează şi pe cea de decriptare şi, prin urmare, prima nu poate fi făcută publică. Aceasta înseamnă că cele două părți legale (emițătorul şi receptorul) trebuie să cadă de acord *în avans* asupra metodei de criptare. Aceasta poate avea loc fie prin întâlnirea dintre cele două părți, fie prin transmiterea cheii de criptare pe un anumit canal sigur.

Dacă se foloseşte un criptosistem cu chei publice, nu este obligatoriu ca cele două părți să se fi întâlnit – ele pot chiar să nu se cunoască una cu alta sau chiar să nu fi comunicat între ele vreodată. Aceasta ester un avantaj uriaş, de exemplu, în cazul unei mari bănci de date, unde există numeroşi utilizatori şi unul dintre ei vrea să comunice numai cu anumit utilizator. Atunci el poate realiza acest lucru introducând informații în însăşi baza de date.

Criptosistemele cu chei publice şi cele clasice se pot compara şi în ce priveşte *lungimea* cheii. Deoarece fiecare cheie trebuie descrisă într-un fel sau altul, descrierea fiind o secvență de litere ale unui anumit alfabet (adică un cuvânt), pare foarte natural să vorbim despre lungimea unei chei. Există o diferență considerabilă între criptosistemele cu chei publice şi cele clasice.

Să considerăm la început un criptosistem clasic. Dacă cheia este mai lungă decât textul clar, realmente nu s-a obținut nimic. Deoarece cheia trebuie transmisă pe un canal sigur am putea transmite textul clar în locul cheii pe acelaşi canal sigur. Evident, în anumite situații cheia se transmite în avans în aşteptarea unui moment crucial.

Să considerăm acum şi criptosistemul cu chei publice. Lungimea cheii de criptare este în mare măsură irelevantă. Cheia este oricum publică. Aceasta înseamnă că şi lungimea cheii de decriptare este irelevantă, receptorul trebuind să o stocheze într-un singur loc.

Uşurința gestiunii cheilor poate fi privită cu îndreptățire ca avantajul esențial al criptografiei cu chei publice.