TEMA 5: INTRODUCERE ÎN CRIPTOGRAFIE

Obiectivele temei:

- Familizarizarea cu obiectivele și noțiunile de bază ale criptografiei.
- Analiza cifrurilor clasice.
- Studierea mecanismelor de criptare.
- Introducere în atacurile criptografice.

Cuvinte-cheie:

cifru,
sistem de criptare,
text clar,
text criptat,
substituția,
transpunerea,
cifru monoalfabetic,
cifru polialfabetic,
mașini rotor,
criptanaliză,
atacuri criptografice,
forţă brută (metoda exhaustivă).

Există dovezi că criptografia ca metodă de protecție a textului s-a născut odată cu apariția scrisului, iar metodele de secretizare a mesajelor erau deja cunoscute de civilizațiile antice din India, Egipt și Mesopotamia. Cele mai vechi texte indiene printre cele 64 de arte menționau și metodele de modificare a textului, unele dintre care pot fi considerate metode criptografice. Pentru a ascunde ce a scris, autorul unei plăci din Mesopotamia ce conținea o rețetă pentru fabricarea glazurii pentru ceramică utiliza semne necunoscute, sărea peste unele litere, iar numele le înlocuia cu cifre. De atunci au fost găsite mai multe referiri la criptografie, cea mai mare parte a având o utilizare militară.

În zilele noastre Criptografia oferă unele dintre cele mai bune metode (dacă nu chiar cele mei bune) de protecție a informației. Metodele criptografice sunt aplicate în rețele și pe WEB, ele sunt implementate în aplicații software și pe dispozitive hardware, devenind deja un accesoriu indispensabil al tehnologiilor moderne.

5.1. Obiective și noțiuni de bază ale criptografiei

5.1.1. Noțiune de criptografie

Criptografia reprezintă o ramură a matematicii care se ocupă cu securizarea informaţiei, asigurând de asemenea autentificarea şi restricţionarea accesului într-un sistem informatic. În realizarea acestora se utilizează în mare parte metode matematice, profitând de unele probleme cu complexitate de rezolvare suficient de înaltă. Termenul „*criptografie*” este compus din cuvintele de

origine greacă *κρυπτός*, kryptós (ascuns) şi γράφειν, gráfein (a scrie). Criptografia modernă urmăreşte realizarea următoarelor obiective:

- *confidenţialitatea* (*privacy*) – proprietatea de a păstra secretul informaţiei, pentru ca aceasta să fie folosită numai de persoanele autorizate;
- *integritatea datelor* (*integrity*) – proprietatea de a evita orice modificare (inserare, ştergere, substituţie) neautorizată a informaţiei;
- *autenticitatea* (*authenticity*) – proprietatea de a identifica o entitate conform anumitor standarde, care include autenticitatea unei entităţi sau a sursei informaţiei;
- *non-repudierea* (*nonrepudiation*) – proprietatea care previne negarea unor evenimente anterioare.

Celelalte obiective legate de securitatea informaţiei (*autentificarea mesajelor, semnături, autorizare, validare, controlul accesului, certificare, timestamping, confirmarea recepţiei, anonimitate, revocare*) pot fi derivate din aceste patru.

Împreună cu Criptografia se dezvoltă *Criptanaliza* – (din greacă, *kryptós*, „ascuns", şi *analýein*, „a dezlega") studiul metodelor de obţinere a înţelesului informaţiilor criptate, fără a avea acces la informaţia secretă necesară în mod normal pentru aceasta. De regulă, aceasta implică găsirea unei chei secrete. Criptografia şi Criptanaliza împreună constituie *Criptologia* (din greacă, *kryptós*, „ascuns", şi λόγος, „cuvânt") – ştiinţa care se ocupă cu metodele de criptare şi decriptare.

În continuare sunt date unele noţiunile fundamentale cu care se operează în criptologie.

O mulţime nevidă *T* se numeşte *alfabet*.

Elementele alfabetului *T* se numesc *litere*.

O consecutivitate finită de elemente din alfabetul *T* se numeşte *cuvânt*. Una şi aceeaşi literă poate intra într-un cuvânt de mai multe ori.

Numărul de elemente ale alfabetului se numeşte *lungimea alfabetului*.

Un cuvânt ce nu conţine nici o literă se numeşte *cuvânt nul*.

Lungimea cuvântului, notată deseori cu *w*, este numărul de litere în acest cuvânt, unde fiecare literă se consideră de câte ori se întâlneşte în el.

Vom nota cu T^* mulţimea tuturor cuvintelor alfabetului *T*. Submulţimile mulţimii T^* le vom numi *limbaje* (formale) peste *T*.

Un mesaj în forma sa originară se numeşte *text clar* (uneori *text în clar* sau *text plan*, în engleză *plaintext*) şi îl vom nota cu *pt* sau cu *m* sau *M* (de la „*message*" - mesaj).

Rescrierea textului clar, folosind o metodă cunoscută numai de expeditor (eventual şi de destinatar), se numeşte *criptare* (sau *cifrare*) a mesajului. Scopul rescrierii textului clar are ca scop camuflarea *textului clar* în aşa fel încât substanţa să nu sufere modificări semantice pentru ca mesajul să poată fi restabilit de destinatar în versiunea inițială a acestuia.

Text criptat sau *text cifrat* (în engleză *ciphertext*) se numeşte textul obţinut în rezultatul operaţiei de criptare a textului plan. Textul criptat îl vom nota cu *ct*, cu *c* sau cu *C* (de la „*cipher*" - cifrul). Textul cifrat se mai numeşte *criptogramă*.

Procesul retransformării criptogramei în textul original este numit *decriptare* sau *descifrare*.

În procesul schimbului de informație criptată destinatarul primeşte printr-un canal textul criptat de la expeditor şi îl decriptează, ştiind metoda folosită pentru criptare, iar în rezultat el obţine mesajul iniţial. În literatura de specialitate expeditorul de obicei este numit *Alice* iar destinatarul este numit *Bob*. Deci, Alice şi Bob trebuie să stabilească în prealabil detaliile modalităţii de criptare şi de decriptare.

Criptarea se foloseşte pentru a fi siguri că informaţia este inaccesibilă oricărei persoane care nu deţine instrumentul necesar decriptării. Chiar dacă oricine poate vedea datele în formă criptografică, oricum nu va înţelege nimic, care să conducă spre descifrarea textului original. Persoana care interceptează criptograma şi încearcă să obţină textul clar aplicând diverse metode, însă fără a avea cheia de decriptare, este numită *criptanalist*.

5.1.2. Sisteme de criptare

Un sistem care realizează operaţiile de criptare şi decriptare se numeşte *sistem de criptare* (sau *sistem criptografic*, sau *criptosistem*). În criptografia modernă un sistem de criptare este definit ca o structură cu cinci componente ($\boldsymbol{P}$, $\boldsymbol{C}$, $\boldsymbol{K}$, $\boldsymbol{E}$, $\boldsymbol{D}$):

- $\boldsymbol{P} = \{m : m \in \boldsymbol{T}^*\}$ – spaţiul (mulţimea) textelor în clar, scrise pentru un alfabet nevid $\boldsymbol{T}$ (în mod obişnuit $\boldsymbol{T}=\{0,1\}$);
- $\boldsymbol{K}$ – spaţiul (mulţimea) cheilor de criptare k, $k \in \boldsymbol{K}$;
- Familia funcţiilor de criptare dependentă de chei şi de un algoritm de criptare $\boldsymbol{E}$

  $\boldsymbol{E}_k : \boldsymbol{P} \rightarrow \boldsymbol{C}$, $\boldsymbol{E}_k = \{e_k : e_k(m) = c$ şi e_k este injectivă$\}$;
- Familia funcţiilor de decriptare dependentă de chei şi de un algoritm de decriptare $\boldsymbol{D}$

  $\boldsymbol{D}_k : \boldsymbol{C} \rightarrow \boldsymbol{P}$, $\boldsymbol{D}_k = \{ d_k : d_k(e_k(m)) = m$ pentru orice $m \in \boldsymbol{P}\}$;
- $\boldsymbol{C}$ spaţiul (mulţimea) mesajelor cu text criptat unde:

  $\boldsymbol{C}=\{c :$ există $k \in \boldsymbol{K}, m \in \boldsymbol{P}, c = \boldsymbol{E}_k(m)\}$.

Schema aplicării unui sistem de criptare este prezentată în figura 5.1.

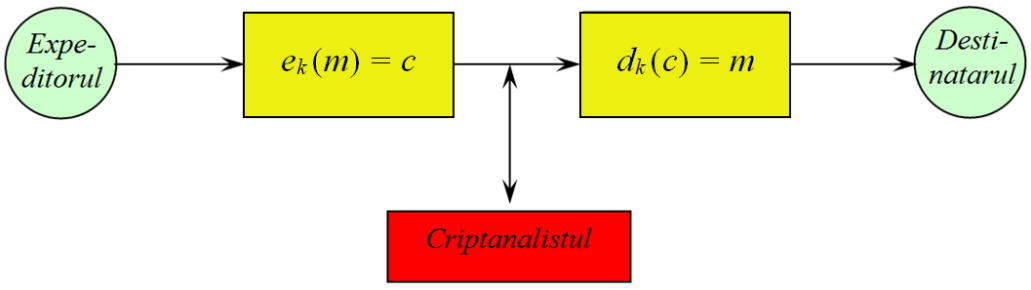

Figura 5.1. *Schema aplicării unui sistem de criptare*

Un sistem de criptare este realizat în baza a trei algoritmi: algoritmul de criptare e_k, algoritmul de decriptare d_k și algoritmul de generare a cheilor.

Pentru ca un sistem de criptare să fie considerat bun, el trebuie să îndeplinească trei criterii (enunţate de Francis Bacon în sec. XVII):

1. Fiind date e_k şi $m \in \boldsymbol{P}$ să fie uşor de calculat $e_k(m)$.
2. Fiind date d_k şi $c \in \boldsymbol{C}$ să fie uşor de determinat $d_k(c)$.
3. Să fie imposibil de determinat m din c, fără a cunoaşte k.

Criteriile 1 şi 2 implică faptul că pentru utilizatorii legali algoritmii de criptate și decriptare nu trebuie să fie prea complicați (se presupune că utilizatorii au un timp acceptabil pentru calcule). În criteriul 3 „imposibilitatea” e înlocuită în prezent cu „dificultatea de a calcula”. Se presupune că un interceptor de asemenea are acces la tehnica de calcul. Ultimul criteriu defineşte (sub o formă vagă) ideea de ”securitate” a sistemului. La aceste criterii, Bacon adăuga şi o a patra regulă:

4. Textul criptat trebuie să fie un text banal, fără suspiciuni.

Această condiţie nu mai poate fi considerată importantă şi este utiliză astăzi doar de un subdomeniu strict al criptografiei, numit *steganografie* – ştiinţa despre transmiterea secretă a informaţiei prin păstrarea secretului a însuşi faptului transmiterii acestei informaţii.

5.1.3. Tipuri de cifruri

Metodele de criptare pot fi divizate pe categorii în felul următor (figura 5.2):

a) *în funcţie de tipul operaţiilor folosite*:
   - bazate pe substituţii;
   - bazate pe transpuneri;
b) *în funcţie de tipul de chei folosite*:
   - metode simetrice (cu cheie secretă);
   - metode asimetrice (cu cheie publică);
c) *metoda prin care datele sunt procesate*:
   - cu cifruri bloc;
   - cu cifruri fluide (flux, şir).

Cifrurile clasice foloseau substituţia sau transpoziţia. Printre metodele moderne de criptare deosebim două direcţii principale: *cifruri cu cheie secretă* (sau *cifruri simetrice*) în care e_k este bijectivă şi *cifruri cu chei publice* (sau *cifruri asimetrice*) – cifruri în care e_k nu este bijectivă.

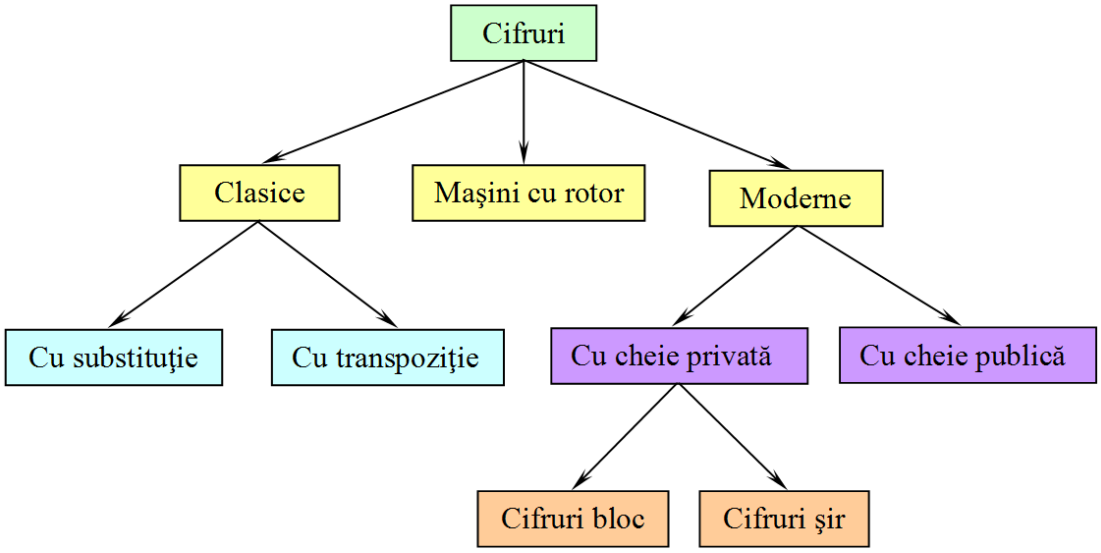

Figura 5.2. *Clasificarea cifrurilor*

Cifrurile simetrice pot fi de *tip bloc* sau de *tip şir* (sau *flux*, sau fluide, în engleză *stream cipher*). Algoritmii de tip bloc acţionează asupra blocurilor de text clar şi text cifrat. Algoritmii de tip şir se aplică şirurilor de text clar şi text cifrat, la nivel de bit sau octet.

Algoritmii moderni de tip bloc criptează mesajul în blocuri de 64 – 265 biţi (lungimea de 64 biți este deja considerată insuficientă, standardul curent de cifrare simetrică – *AES* – lucrează cu blocuri de lungime 128, 192 sau 256 biți). Pentru acesta se aplică o funcţie matematică între un bloc de biţi ai mesajului în clar şi cheie (care poate varia ca mărime), rezultând acelaşi număr de biţi pentru mesajul criptat. Funcţia de criptare este realizată astfel încât să îndeplinească următoarele cerinţe:

- cunoscând un bloc de biţi ai textului clar şi cheia de criptare, algoritmul să poată genera rapid un bloc al textului criptat;
- cunoscând un bloc de biţi ai textului criptat şi cheia de criptare/decriptare, sistemul să poată genera rapid un bloc al textului clar;
- cunoscând blocurile textului clar şi ale textului criptat, să fie dificil de generat cheia.

Cifrurile şir la fel formează o clasă importantă de algoritmi de criptare. Ceea ce le caracterizează şi le diferenţiază faţă de cifrurile bloc este faptul că cifrurile şir procesează informaţia în unităţi oricât de mici, chiar bit cu bit, aplicând funcţia XOR între biţii cheii şi biţii de cifrat, iar funcţia de criptare se poate modifica în cursul criptării. Cifrurile şir sunt algoritmi cu memorie, în sensul că procesul de criptare nu depinde doar de cheie şi de textul clar, ci şi de starea curentă. În cazul în care probabilitatea erorilor de transmisie este mare, folosirea cifrurilor şir este avantajoasă deoarece au proprietatea de a nu propaga erorile. Ele se folosesc şi în cazurile în care datele trebuie procesate una câte una, datorită lipsei de spaţiu de memorie.

Cifruri *asimetrice* utilizează o pereche de chei: o *cheie publică* și o *cheie privată*. Un utilizator care deține o astfel de pereche își publică o cheie (cheia publică) astfel încât oricine

dorește să o poată folosi pentru a-i transmite un mesaj criptat. Numai deținătorul cheii secrete (private) este cel care poate decripta mesajul astfel criptat.

Cele două chei sunt legate matematic, însă cheia privată nu poate fi obținută din cheia publică. În caz contrar, oricine ar putea decripta mesajele destinate unui alt utilizator, fiindcă oricine are acces la cheia publică a acestuia. O analogie foarte potrivită pentru proces este folosirea cutiei poștale. Oricine poate pune în cutia poștală a cuiva un plic, dar la plic nu are acces decât posesorul cheii de la cutie.

Criptografia asimetrică se mai numește criptografie cu chei publice şi e compusă din două mari ramuri:

- criptarea cu cheie publică – un mesaj criptat cu o cheie publică nu poate fi decodificat decât folosind cheia privată corespunzătoare. Metoda este folosită pentru a asigura confidențialitatea;
- semnături digitale – un mesaj semnat cu cheia privată a emițătorului poate fi verificat de către oricine, prin acces la cheia publică corespunzătoare, astfel asigurându-se autenticitatea mesajului.

5.2. Cifruri clasice

Criptografia clasică se încadrează în clasa criptografiei cu chei simetrice și este criptografia dinaintea calculatorului, de unde şi denumirea de criptografie pre-computaţională. În criptografia clasică, algoritmii erau bazaţi pe caracter şi constau dintr-o serie de transformări elementare (substituţii și transpoziţii) ale caracterelor textului clar. Unii algoritmi aplicau aceste transformări în mod repetat, îmbunătăţind în acest mod securitatea algoritmului. În criptografia modernă bazată pe calculator (criptografie computaţională), lucrurile s-au complicat, dar multe dintre ideile criptografiei clasice au rămas nemodificate.

5.2.1. Cifrul de substituţie

Cifrul de substituţie (*substitution cipher*) este cifrul la care fiecare caracter sau grup de caractere ale textului clar *m* este substituit cu un alt caracter sau grup de caractere în textul cifrat *c*, descifrarea făcându-se prin aplicarea substituţiei inverse asupra textului cifrat. În criptografia clasică există patru tipuri de cifruri de substituţie: monoalfabetică, polialfabetică, poligramică și homofonică.

Cifruri de substituţie monoalfabetică (*monoalphabetic ciphers*) sunt cifrurile în care fiecare caracter al textului în clar *m* este înlocuit cu un caracter corespunzător în textul cifrat *c*. Reprezentanți ai acestei clase sunt: cifrul Cezar, cifrul Afin, cifrul Polibios, etc.

Cifrul lui *Cesar* (sau *Cezar*). În acest cifru fiecare literă a textului clar este înlocuită cu o nouă literă obţinută printr-o deplasare alfabetică. Cheia secretă *k*, care este aceeaşi la criptare cât şi la decriptare, constă în numărul care indică deplasarea alfabetică, adică $k \in \{1, 2, 3, \ldots, n-1\}$, unde *n*

este lungimea alfabetului. Criptarea şi decriptarea mesajului cu cifrul Cezar poate fi definită de formulele

$$c = e_k(x) = x + k \pmod{n},$$
$$m = d_k(y) = y - k \pmod{n},$$

unde x și y sunt reprezentarea numerică a caracterului respectiv al textului clar. Funcţia numită *Modulo* (a mod b) returnează restul împărţirii numărului întreg a la numărul întreg b. Această metodă de criptare este numită aşa după Iulius Cezar, care o folosea pentru a comunica cu generalii săi, folosind cheia $k = 3$ (tabelul 5.1).

De exemplu, pentru $k = 3$ avem

$$e_k(\mathrm{S}) = 18 + 3 \pmod{26} = 21 = \mathrm{V}$$
$$d_k(\mathrm{V}) = 21 - 3 \pmod{26} = 18 = \mathrm{S}$$

În acest caz pentru m =„*cifrul cezar*”, obținem c =„*fliuxo fhcdu*”.

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

Tabelul 5.1. *Cifrul Cezar cu cheia k*=3

Cifrul lui Cezar este foarte uşor de spart, deci este un cifru foarte slab. Astfel, un criptanalist poate obţine textul clar prin încercarea tuturor celor 25 de chei. Nu se ştie cât de util era cifrul Cezar în timpul când era folosit de către cel de la care îi provine numele, dar este probabil ca el să fi fost destul de sigur, atât timp cât numai câţiva dintre inamicii lui Cezar erau în stare să scrie şi să citească, dar mai ales să cunoască concepte de criptanaliză.

Cifrul afin este o generalizare a cifrului Cezar, având cheia

$$k = \{(a, b) \mid a, b \in \mathrm{Z}_{26} = \{0, 1, 2, \ldots, 25\}, \mathrm{cmmdc}(a, 26) = 1\},$$

iar funcţiile de criptare şi decriptare pentru o cheie $k = (a, b)$ sunt

$$e_k(\boldsymbol{x}) = a\boldsymbol{x} + b \pmod{26}$$

și

$$d_k(\boldsymbol{y}) = a^{-1}(y+26 - b) \pmod{26},$$

unde $a \cdot a^{-1} = 1 \bmod n$.

Condiţia ca a să fie prim cu 26 asigură existenţa lui a^{-1} în Z_{26}. Pentru calcularea lui a^{-1} poate fi aplicat *algoritmul Euclid extins* (vezi compartimentul 7), sau se verifică pa rând toate numerele reciproc prime cu a în clasa modulo n până când găsim acel număr pentru care $a \cdot a^{-1} = 1 \bmod n$, acesta fiind și unicul cu o astfel de proprietate.

De exemplu, pentru $a = 7$, $b = 16$ funcţia de criptare este $e_k(x) = 7x + 16$. În acest caz pentru m =„*cifrul afin*”, obținem c =„ *euzfap qzud*”.

Deoarece $7 \cdot 15 = 1 \pmod{26}$, avem $7^{-1} = 15 \pmod{26}$, iar decriptarea se realizează matematic folosind funcţia

$$d_k(y)= 15\cdot (y+26-15)(\text{mod } 26)=15y+15\cdot 11 \text{ (mod } 26)=15y + 9 \text{ (mod } 26).$$

Condiţia cmmdc(a, 26)=1 asigură de asemenea injectivitatea funcţiei e_k, necesară pentru un proces corect de criptare-decriptare. De exemplu, pentru $e_k(x) = 10x + 1$, *A* şi *N* se transformă ambele în *B,* iar *O* nu apare ca imagine în alfabetul substituţiei. În acest caz avem:

$$\text{cmmdc}(a, 26)= \text{cmmdc}(10, 26)=2\neq 1.$$

Astfel de cazuri au loc întotdeauna atunci când a nu respectă condiția de mai sus.

Orice cheie k a cifrului afin este determinată complet de valorile întregi (a, b) pentru care cmmdc(a, 26) = 1. Sunt posibile 12 valori pentru a: 1, 3, 5, 7, 9, 11, 15, 19, 21, 23, 25 şi 26 valori pentru b, care se iau independent de a, cu o singură excepţie a = 1, b = 0 (care se exclude deoarece nu conduce la nici o criptare). Aşadar mulţimea cheilor în acest caz este alcătuită din $12\cdot 26-1=311$ chei diferite, număr suficient de mic pentru atacul prin forţa brută, adică prin verificarea tuturor cheilor posibile.

Trebuie de avut în vedere că în cazul în care este necesar de a cripta mesaje în limba română sau rusă (sau oricare altă limbă care are un alfabet) procedeul rămâne același, doar că trebuie să ținem cont de faptul că alfabetele respective au 31 și 33 de litere, adică n=31 și respectiv n=33.

Cifrul *Polibios*. Cifrul Cezar nu este cel mai vechi algoritm de criptare. Se pare că primul astfel de algoritm a fost utilizat de Polybios (istoric grec, care a decedat cu 30 ani înaintea naşterii lui Cezar). Iniţial acesta a fost doar un sistem maritim de semnalizare cu torţe iar ulterior i s-a dat o semnificaţie criptografică.

În cifrul Polibios pentru fiecare alfabet se construieşte un careu aparte de cifrare, de cele mai dese ori cu numărul de coloane şi linii egal (însă nu e o condiţie necesară). Dimensiunile careului depind de lungimea n a alfabetului. Pentru a crea careul se iau două numere întregi, produsul cărora e cel mai aproape de n. Liniile şi coloanele se numerotează. După aceasta literele alfabetului se înscriu în acest careu în ordinea lor naturală. Dacă nu sunt suficiente celule pentru literele alfabetului se pot înscrie într-o celulă 2 litere sau se pot omite câteva litere (de obicei de frecvenţa cât mai redusă). Pentru alfabetul latin, putem avea careuri Polibios 5×5, după cum este reprezentat în tabelul 5.2 (în al treilea careu a fost omisă litera Q care este una cu frecvenţă redusă):

	m	*o*	*u*	*s*	*e*
m	A	B	C	D	E
u	F	G	H	I/J	K
s	L	M	N	O	P
c	Q	R	S	T	U
a	V	W	X	Y	Z

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

	a	*b*	*c*	*d*	*e*
a	A	B	C	D	E
b	F	G	H	I	J
c	K	L	M	N	O
d	P	R	S	T	U
e	V	W	X	Y	Z

Tabelul 5.2. *Exemple de tabele ale cifrului Polibios*

În operaţia de criptare, fiecare caracter x va fi reprezentat printr-o pereche de litere (x, y), unde $x \in \{m, o, u, s, e\}$ iar $y \in \{m, u, s, c, a\}$ pentru primul exemplu din figură, sau $x, y \in \{1, 2, 3, 4, 5\}$ și $x, y \in \{a, b, c, d, e\}$ pentru exemplele doi și trei.

Astfel, pentru primul exemplu de tabel, textul clar VENI VIDI VICI este criptat în

ma em us su ma su sm su ma su um su.

Cifrul Polibius, prezentat în exemplul din mijloc al tabelului 5.2, a fost folosit de condamnații din penitenciarele ruseşti şi de către prizonierii americani din Vietnam. Cifrul este foarte simplu de învăţat şi poate fi aplicat folosind diverse semne drept coordonate-chei (cifre, puncte, figuri, etc). Cifrul Polibios a fost utilizat de asemenea în cadrul altor sisteme de criptare, cum ar fi sistemul nihilist, cifrul ADFGVX (utilizat de armata germană în primul război mondial) sau sistemul Bifid.

Punctul slab al sistemelor de criptare monoalfabetice constă în frecvenţa de apariţie a caracterelor în text. Dacă un text criptat este suficient de lung şi se cunoaşte limba în care este scris textul clar, sistemul poate fi spart printr-un atac bazat pe *frecvenţa apariţiei literelor* într-o limbă (atacul prin analiza frecvenței), această frecvență fiind o problemă studiată intens (nu neapărat în scopuri criptografice) iar în rezultat au fost construite diverse structuri de ordine relativ la frecvenţa apariţiei literelor în fiecare limbă europeană și în alte limbi.

De obicei, cu cât un text criptat este mai lung, cu atât frecvenţa literelor folosite se apropie de această ordonare generală. O comparare între cele două relaţii de ordine (cea a caracterelor din textul criptat şi cea a literelor din alfabetul limbii curente) conduce la realizarea câtorva corespondenţe (literă text clar – literă text criptat), ceea ce stabileşte în mod univoc cheia de criptare. Pentru algoritmul Cezar este suficientă stabilirea unei singure perechi, pentru cel afin trebuiesc două perechi etc. Pentru limba română frecvenţa literelor este prezentată în tabelul 5.3 şi figura 5.3.

A	Ă	Â	B	C	D	E	F	G	H	I	Î	J	K	L	M
9,95	*4,06*	*0,91*	*1,07*	*5,28*	*3,45*	*11,47*	*1,18*	*0,99*	*0,47*	*9,96*	*1,40*	*0,24*	*0,11*	*4,48*	*3,10*
N	O	P	Q	R	S	Ş	T	Ţ	U	V	W	X	Y	Z	
6,47	*4,07*	*3,18*	*0,00*	*6,82*	*4,40*	*1,55*	*6,04*	*1,00*	*6,20*	*1,23*	*0,03*	*0,11*	*0,07*	0,71	

Tabelul 5.3. *Frecvenţa literelor limbii române*

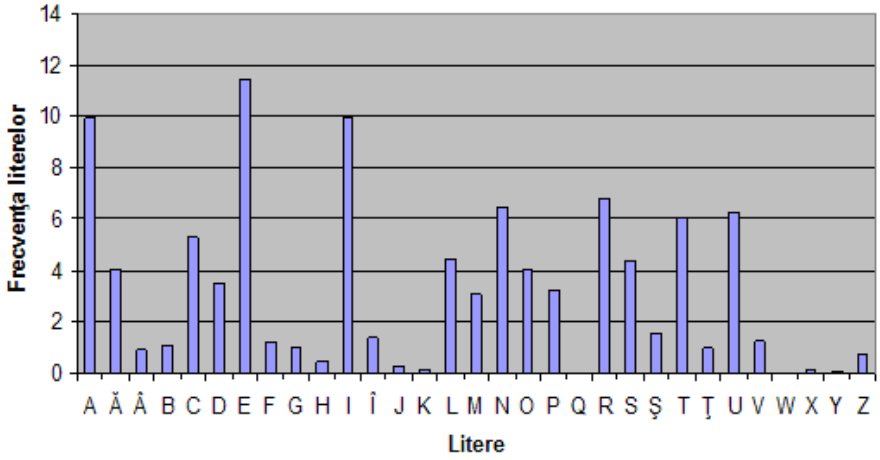

Figura 5.3. *Frecvenţa literelor limbii române*

Cifruri de substituţie polialfabetică (*polyalphabetic ciphers*). Slăbiciunea cifrurilor monoalfabetice este definită de faptul că distribuţia lor de frecvenţă reflectă distribuţia alfabetului folosit. Un cifru este mai sigur din punct de vedere criptografic dacă prezintă o distribuţie cât mai regulată, care să nu ofere informaţii criptanalistului.

O cale de a aplatiza distribuţia este combinarea distribuţiilor ridicate cu cele scăzute. Daca litera *T* este criptată câteodată ca *a* şi altă dată ca *b*, şi dacă litera *X* este de asemenea câteodată criptată ca *a* şi altă dată ca *b*, frecvenţa ridicată a lui *T* se combină cu frecvenţa scăzută a lui *X* producând o distribuţie mai moderata pentru *a* şi pentru *b*. Două distribuţii se pot combina prin folosirea a doua alfabete separate de criptare, primul pentru caracterele aflate pe poziţii pare în textul clar, al doilea pentru caracterele aflate pe poziţii impare, rezultând necesitatea de a folosi alternativ doua tabele de translatare, de exemplu permutările

$$p_1(a)=(3\cdot a) \bmod 26 \text{ şi } p_2(a)=(7\cdot a+13) \bmod 26.$$

Diferenţa dintre cifrurile polialfabetice şi cele monoalfabetice constă în faptul că substituţia unui caracter variază în text, în funcţie de diverşi parametri (poziţie, context etc.). Aceasta conduce bineînţeles la un număr mult mai mare de chei posibile. Se consideră că primul sistem de criptare polialfabetic a fost creat de Leon Battista în 1568. Unele aplicaţii actuale folosesc încă pentru anumite secţiuni astfel de sisteme de criptare.

Cifrul Vigenere. La fel ca cifrul Cezar, cifrul Vigenere deplasează literele, dar, spre deosebire de acesta nu se poate sparge uşor în 26 combinaţii. Cifrul Vigenere foloseşte o deplasare multiplă. Cheia nu este constituită de o singură deplasare, ci de mai multe, fiind generate de câţiva întregi k_i, unde $0 \le k_i \le 25$, dacă luăm ca reper alfabetul latin cu 26 de litere. Criptarea se face în felul următor:

$$c_i = m_i + k_i \pmod{26}.$$

Cheia poate fi, de exemplu, $k = (5, 20, 17, 10, 20, 13)$ şi ar provoca deplasarea primei litere cu 5, $c_1=m_1 + 5 \pmod{26}$, a celei de a doua cu 20, $c_2 = m_2 + 20 \pmod{26}$, ş.a.m.d. până la sfârşitul cheii

şi apoi de la început, din nou. Cheia este de obicei un cuvânt, pentru a fi mai uşor de memorat – cheia de mai sus corespunde cuvântului „*furtun*”. Metoda cu deplasare multiplă oferă protecţie suplimentară din două motive:

- primul motiv este că ceilalţi nu cunosc lungimea cheii;
- cel de al doilea motiv este că numărul de soluţii posibile creşte; de exemplu, pentru lungimea cheii egală cu 5, numărul de combinaţii care ar fi necesare la căutarea exhaustivă ar fi $26^5 = 11\ 881\ 376$.

Decriptarea pentru cifrul Vigenere este asemănătoare criptării. Diferenţa constă în faptul că se scade cheia din textul cifrat,

$$m_i = c_i - k_i \pmod{26}.$$

Pentru simplificarea procesului de cifrare se poate utiliza următorul tabel, numit *Tabula Recta* (tabelul 5.4). Aici toate cele 26 cifruri sunt situate pe orizontală şi fiecărui cifru îi corespunde o anumită literă din cheie, reprezentată în colana din stânga tabelului. Alfabetul corespunzător literelor textului clar se află în prima linie de sus a tabelului. Procesul de cifrare este simplu – este necesar ca având litera ***x*** din cheie şi litera ***y*** din textul clar să găsim litera textului cifrat care se află la intersecţia liniei ***x*** şi coloanei ***y***. Se poate de procedat şi în conformitate cu ecuaţiile ce definesc cifrul $c_i = m_i + k_i \pmod{26}$ şi $m_i = c_i - k_i \pmod{26}$, aşa cum este arătat în exemplul ce urmează.

	a	***b***	***c***	***d***	***e***	***f***	***g***	***h***	***i***	***j***	***k***	***l***	***m***	***n***	***o***	***p***	***q***	***r***	***s***	***t***	***u***	***v***	***w***	***x***	***y***	***z***
a	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
b	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
c	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
d	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
e	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
f	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
g	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
h	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
i	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
j	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
k	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
l	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
m	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
n	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
o	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
p	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
r	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
s	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
t	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
u	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
v	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
w	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
x	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	**V**	W
y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
z	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

Tabelul 5.4. *Tabula Recta pentru cifrul Vigenere*

Problemă.

De cifrat, utilizând cifrul Vigenere, mesajul „*Per aspera ad astra*” folosind cheia ***K***= SUPER.

Soluție. Pentru a cifra sau descifra mai întâi facem corespondenţa următoare:

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

Apoi alcătuim şi completăm tabelul:

Textul clar ***M***	P	E	R	A	S	P	E	R	A	A	D	A	S	T	R	A
Cheia ***K***	**S**	**U**	**P**	**E**	**R**	S	U	P	E	R	S	U	P	E	R	S
Textul clar ***M***	15	4	17	0	18	15	4	17	0	0	3	0	18	19	17	0
Cheia ***K***	18	20	15	4	17	18	20	15	4	17	18	20	15	4	17	18
M+***K*** (mod 26)	7	24	6	4	9	7	24	6	4	17	21	20	7	23	8	18
Textul cifrat ***C***	H	Y	G	E	J	H	Y	G	E	R	V	U	H	X	I	S

Criptograma este ***C***= *HYGEJHYGERVUHXIS*.

Pentru decriptare procedăm la fel, cu excepţia $m_i = c_i - k_i \pmod{26}$.

Pentru acesta la fel completăm următorul tabel:

Textul cifrat ***C***	H	Y	G	E	J	H	Y	G	E	R	V	U	H	X	I	S
Cheia ***K***	S	U	P	E	R	S	U	P	E	R	S	U	P	E	R	S
Textul cifrat ***C***	7	24	6	4	9	7	24	6	4	17	21	20	7	23	8	18
Cheia ***K***	18	20	15	4	17	18	20	15	4	17	18	20	15	4	17	18
M–***K*** (mod 26)	15	4	17	0	18	15	4	17	0	0	3	0	18	19	17	0
Textul clar ***M***	P	E	R	A	S	P	E	R	A	A	D	A	S	T	R	A

Textul clar este ***M*** = PERASPERAADASTRA.

Criptanaliza sistemului Vigenere constă în următoarele:

fie $c = c_0\, c_1 \ldots c_{n-1}$ textul criptat cu cheia $k = k_0\, k_1 \ldots k_{p-1}$; putem aranja acest text sub forma unei matrice cu p linii şi $\lceil n/p \rceil$[1] coloane, astfel:

$$\begin{matrix} c_0 & c_p & c_{2p} & \ldots \\ c_1 & c_{p+1} & c_{2p+1} & \ldots \\ \ldots & \ldots & \ldots & \ldots \\ c_{p-1} & c_{2p-1} & c_{3p-1} & \ldots \end{matrix}$$

Elementele de pe prima linie au fost criptate după formula

$$c_{pr} = a_{pr} + k_0 \pmod{26},\ k \geq 0,$$

adică cu un sistem Cezar (k_0 fiind o valoare fixată din $\boldsymbol{Z}_{26}$). În mod similar şi celelalte linii. Deci, dacă s-ar cunoaşte lungimea p a cheii, problema s-ar reduce la criptanaliza a p texte criptate cu Cezar – sistem de criptare monoalfabetic. Sunt cunoscute două metode pentru aflarea lungimii cheii: testul lui Kasiski şi indexul de coincidenţă.

[1] ⌈x⌉ - *funcţia plafon*, sau *partea întreagă prin exces*, este cel mai mic număr întreg mai mare decât numărul real x dat.

Prima metodă constă în studiul textului criptat şi aflarea de perechi de segmente de cel puţin 3 caractere identice (această lungime este propusă de Kasiski). Pentru fiecare astfel de pereche, se determină distanta dintre segmente. După ce s-au găsit mai multe astfel de distanţe, valoarea lui *p* va fi cel mai mare divizor comun al lor (sau – eventual un divizor al acestuia).

A doua metodă de aflare a lungimii cheii de criptare într-un sistem Vigenere se bazează pe un concept definit în 1920 de Wolfe Friedman - *indexul de coincidenţă*. Dacă $c = c_1c_2...c_n$ este o secvenţă de *n* caractere alfabetice, probabilitatea ca două caractere din *c*, alese aleator, să fie identice se numeşte "*index de coincidenţă*" $I_c(x)$ al lui *c*.

Cifrul omofonic (homophonic ciphers) este un cifru de substituţie în care un caracter al alfabetului mesajului clar (alfabet primar) poate să aibă mai multe reprezentări. Ideea utilizată în aceste cifruri este uniformizarea frecvenţelor de apariţie a caracterelor alfabetului textului cifrat (alfabet secundar), pentru a îngreuna atacurile criptanalitice. Astfel, litera *A* - cu cea mai mare frecvenţă de apariţie în alfabetul primar – poate fi înlocuită de exemplu cu *H*, # sau *m*.

Cifrul omofonic este un cifru intermediar între sistemele mono şi cele polialfabetice. Principalul lui scop este de a evita atacul prin frecvenţa de apariţie a caracterelor. Se presupune că a fost utilizat prima oară în 1401 de către ducele de Mantua. În cifrul omofonic fiecărui caracter $a \in m$ i se asociază o mulţime $H(a) \subset c$ astfel încât:

- $H(a) \cap H(b) = \varnothing \Leftrightarrow a \neq b$;
- Dacă *a* apare mai frecvent decât *b* în textele clare, atunci card ($H(a)$) ≥ card(H(b)).

Criptarea unui caracter $x \in m$ se face cu un element ales aleator din $H(a)$. Pentru decriptarea lui $y \in c$ se caută o mulţime $H(x)$ astfel încât $y \in H(x)$.

Exemplu. Pentru limba engleză poate fi utilizat cifrul definit de tabelul 5.5, în primele două linii ale căruia sunt aşezate literele alfabetului latin şi frecvenţele rotunjite ale acestora. În coloanele de sub litera *x* este situat $H(x)$. De exemplu

$$H(n) = \{18, 58, 59, 66, 71,91\}.$$

Pentru criptarea textului „*ab*" se poate folosi oricare din secvenţele
0948, 1248, 3348, 4748, 5348, 6748, 7848, 9248, 0981, 1281, 3381, 4781, 5381, 6781, 7881, 9281.

Tabelul 5.5 a fost completat cu numerele 00, 01,…, 99, ele fiind în cantitate de 100 pentru a simplifica calculele legate de frecvența literelor (adică pentru a corespunde cu 100% a frecvențelor). Însă elementele acestui tabel pot fi oricare simboluri, iar cantitatea lor va determina coeficientul de proporționalitate pentru frecvențele fiecărei litere din alfabet.

a	**b**	**c**	**d**	**e**	**f**	**g**	**h**	**i**	**j**	**k**	**l**	**m**	**n**	**o**	**p**	**q**	**r**	**s**	**t**	**u**	**v**	**w**	**x**	**y**	**z**
8	2	3	4	12	2	2	6	6	1	1	4	2	6	7	2	1	6	6	9	3	1	2	1	2	1
09	48	13	01	14	10	06	23	32	15	04	26	22	18	00	38	94	29	11	17	08	34	60	28	21	02
12	81	41	03	16	31	25	39	70			37	27	58	05	95		35	19	20	61		89		52	
33		62	45	24			50	73			51		59	07			40	36	30	63					
47			79	44			56	83			84		66	54			42	76	43						
53				46			65	88					71	72			77	86	49						
67				55			68	93					91	90			80	96	69						
78				57										99					75						
92				64															85						
				74															97						
				82																					
				87																					
				98																					

Tabelul 5.5. *Exemplu de cifru omofonic pentru limba engleză*

Deşi mai greu de spart decât cifrurile de substituţie simple (monoalfabetice), cifrul omofonic nu maschează total proprietăţile statistice ale textului clar. În cazul unui atac cu text clar cunoscut, cifrul se sparge extrem de uşor. Atacul cu text cifrat este mai dificil, dar unui calculator îi va lua doar câteva secunde pentru a-l sparge.

Substituţie poligramică realizează substituirea unor blocuri de caractere (poligrame) din textul clar, distrugând astfel semnificaţia, atât de utilă în criptanaliză, a frecvenţelor diferitelor caractere. Vom considera un mesaj $m=m_1m_2...m_dm_{d+1}...$ şi un cifru care prelucrează poligrame de lungime d. Criptograma rezultată este $c=c_1c_2...c_dc_{d+1}...c_{d+d}...$, fiecare poligramă $m_{i,d+1}...m_{i,d+d}$ fiind transformată în poligrama $c_{i,d+1}...c_{i,d+d}$ prin funcţia de substitute f_j astfel:

$$c_{i,d+j} = f_j(m_{i,d+1}\ldots m_{i,d+d})$$

În cazul cifrării literelor singulare frecvenţa de apariţie a literelor în textul cifrat este egală cu frecvenţa de apariţie a literelor corespunzătoare din textul clar. Această invarianţă a frecvenţelor furnizează o cantitate de informaţie suficientă criptanalistului pentru spargerea cifrului. Pentru minimizarea informaţiei colaterale furnizate de frecvenţa de apariţie a literelor s-a utilizat cifrarea grupurilor de d litere (d-grame). În cazul când un grup de d litere este substituit printr-un alt grup de d litere, substituţia se numeşte poligramică. Substituţia poligramică cea mai simplă se obţine pentru d =2 când digrama m_1m_2 din textul clar se substituie cu digrama c_1c_2 din textul cifrat.

Un exemplu clasic pentru substituţia diagramelor este cifrul lui *Playfair* (tabelul 5.6).

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

Tabelul 5.6. *Exemplu de cifru Playfair*

Primele litere din pătrat (în caz general – un dreptunghi) reprezintă un cuvânt cheie k (literele care se repetă se scriu o singură dată, în acest exemplu cheia fiind $\boldsymbol{k}$=PLAYFAIR), după care pătratul se completează cu literele alfabetului în ordinea lor naturală, fără repetare. Cifrarea se executa după următoarele reguli:

- Pentru separarea literelor identice alăturate se introduc nişte caractere de separare care, de regula, au frecventa de apariţie redusă, cum sunt de exemplu literele X, Q în limba româna. În cazul în care numărul de caractere în textul clar este impar se procedează la fel. La descifrare aceste litere introduse se omit.
- Dacă m_1m_2 sunt dispuse în vârfurile opuse ale unui dreptunghi, atunci c_1c_2 sunt caracterele din celelalte vârfuri ale dreptunghiului, c_1 fiind în aceeaşi linie cu m_1. De exemplu RK devine CG, deci RK $\rightarrow$ CG;
- Dacă m_1 şi m_2 se găsesc într-o linie, atunci c_1 şi c_2 se obţin printr-o deplasare ciclică spre dreapta a literelor m_1 şi m_2. De exemplu PA $\rightarrow$ LY iar NO $\rightarrow$ OQ;
- Dacă m_1 şi m_2 se află în aceeaşi coloană atunci c_1 şi c_2 se obţin printr-o deplasare ciclică a lui m_1, m_2 de sus în jos. De exemplu AH $\rightarrow$ BQ iar OV $\rightarrow$ VL;

Descifrarea se executa după reguli asemănătoare cu cele de cifrare, inversând direcția.

Exemplu. Folosind exemplul de mai sus ($\boldsymbol{k}$ = PLAYFAIR) pentru textul clar m=„*VINE IARNA*” obţinem textul cifrat c=„UR UN BP IO YW”. Aici am introdus la sfârşitul mesajului litera X iar AX $\rightarrow$ YW. La descifrare după sensul mesajului se omite această literă.

Cifrul Playfair se folosea în scopuri tactice de către forţele militare britanice în timpul celui de-al doilea război al Burilor (1899-1902) dar şi în primul război mondial. La fel a fost utilizat de către australieni şi germani în timpul celui de-al doilea război mondial. El era utilizat deoarece era suficient de rapid în aplicare şi nu necesita nici un utilaj special. Scopul principal al utilizării lui era protecţia informaţiei importante (însă nu şi secrete) pe parcursul unei lupte. La momentul când criptanaliştii inamici spărgeau cifrul, informaţia deja nu mai era utilă pentru inamic.

Utilizarea cifrului Playfair în prezent nu are sens deoarece laptop-urile moderne pot sparge cu uşurinţă cifrul în câteva secunde. Primul algoritm de spargere pentru Playfair a fost descris în anul 1914 de către locotenentul Iosif O. Moubornom într-o broşură de 19 pagini.

5.2.2. Cifrul de transpoziţii

Spre deosebire de cifrurile cu substituţie, care păstrează ordinea literelor din textul sursă, însă le transformă, cifrurile cu transpoziţie (*transposition ciphers*) reordonează literele, fără a le „deghiza”. Criptarea prin metoda transpoziţiei este o tehnică mai eficientă decât criptarea prin substituţie, dar are, la rândul ei, o mulţime de dezavantaje. Textul criptat prin metoda transpoziţiei

păstrează toate caracterele textului iniţial, dar în altă ordine obţinută prin aplicarea algoritmului ce va fi prezentat în continuare.

Criptarea prin transpoziţie constă în scrierea textului iniţial din care s-au eliminat spaţiile şi semnele de punctuaţie într-o matrice de dimensiune *M*×*N* şi interschimbarea anumitor linii (sau coloane) între ele. Textul criptat se obţine prin scrierea caracterelor din noua matrice de pe fiecare coloană în parte, începând cu colţul din stânga-sus. Dacă lungimea textului iniţial este mai mică decât numărul de elemente ce pot fi scrise în matrice, atunci textul se completează cu elemente aleatoare, până ajunge la dimensiunea *M*·*N*.

Pentru textul „*Securitatea este asigurată*”, care are lungimea de 24 de caractere, se pot alege mai multe matrice de dimensiune *M*×*N*, o posibilitate ar fi ca matricea să aibă 4 linii şi 6 coloane, dar pentru ca textul să fie mai greu de decodificat trebuie să conţină şi caractere alese aleator, sau într-un mod mai inteligent, care să îngreuneze munca celui care doreşte să afle conţinutul secret din mesajul criptat. Fie am ales o matrice care are 5 linii şi 6 coloane. Textului iniţial i se adaugă 6 caractere aleatoare şi se obţine textul *Securi tateae steasi gurată xyztwu* care se scrie în matricea din partea stângă, aşa cum e arătat în tabelul 5.7:

	1	**2**	**3**	**4**	**5**	**6**
1	*S*	*e*	*c*	*u*	*r*	*i*
2	*t*	*a*	*t*	*e*	*a*	*e*
3	*s*	*t*	*e*	*a*	*s*	*i*
4	*g*	*u*	*r*	*a*	*t*	*ă*
5	*x*	*y*	*z*	*t*	*w*	*u*

	1	**2**	**3**	**4**	**5**	**6**
5	*x*	*y*	*z*	*t*	*w*	*u*
3	*s*	*t*	*e*	*a*	*s*	*i*
4	*g*	*u*	*r*	*a*	*t*	*ă*
1	*S*	*e*	*c*	*u*	*r*	*i*
2	*t*	*a*	*t*	*e*	*a*	*e*

Tabelul 5.7. *Exemplu de cifru cu transpoziţie*

Prin scrierea liniilor 1, 2, 3, 4, 5 în ordinea 5, 3, 4, 1, 2, această ordine fiind cheia cifrului, adică *k*=5,3,4,1,2, se obţine matricea din partea dreaptă. Textul criptat care se obţine este

c= xsgSt ytuea zerct taaue wstra uiăie.

Pentru ca procesul de decriptare să fie mai simplu şi să nu mai fie nevoie de ordinea în care au fost puse liniile din matricea creată, se foloseşte o versiune a criptării prin transpoziţie care se bazează pe un cuvânt-cheie. Pentru a cripta un text folosind o cheie şi metoda transpoziţiei, se alege o cheie ale cărei litere determină ordinea în care se vor scrie coloanele din matricea aleasă. Pentru a afla ordinea în care vor fi scrise coloanele din textul iniţial, se ordonează alfabetic literele din cheie, şi fiecărei litere i se asociază numărul de ordine din şirul ordonat. Lungimea cheii trebuie să fie egală cu numărul de coloane din matrice.

Considerăm textul anterior, scris într-o matrice de dimensiuni 5×6, şi cheia „*butuc*” (tabelul 5.8.). Literele din cheie se ordonează alfabetic şi se obţine şirul: *b, c, t, u, u*. Indicele 1 este asociat cu litera *b*, indicele 2 - cu litera *c*, indicele 3 - cu litera *t*, indicele 4 - cu prima literă *u*, iar indicele 5

este asociat cu a doua literă *u*. Pentru a rescrie liniile, pentru fiecare indice *i* din şirul ordonat se caută indicele *j*, care reprezintă poziţia literei cu indicele *i* din cheie şi se scrie linia *j*, astfel:

		1	**2**	**3**	**4**	**5**	**6**
b	**1**	*S*	*e*	*c*	*u*	*r*	*i*
u	**4**	*t*	*a*	*t*	*e*	*a*	*e*
t	**3**	*s*	*t*	*e*	*a*	*s*	*i*
u	**5**	*g*	*u*	*r*	*a*	*t*	*ă*
c	**2**	*x*	*y*	*z*	*t*	*w*	*u*

	1	**2**	**3**	**4**	**5**	**6**
1	*S*	*e*	*c*	*u*	*r*	*i*
2	*x*	*y*	*z*	*t*	*w*	*u*
3	*s*	*t*	*e*	*a*	*s*	*i*
4	*t*	*a*	*t*	*e*	*a*	*e*
5	*x*	*y*	*z*	*t*	*w*	*u*

Tabelul 5.8. *Exemplu de transpoziţie cu cheie*

Textul cifrat care se obţine în final este *c= Sxstx eytay czetz utaet rwsaw iuieu*.

Pentru a decripta un mesaj criptat cu această metodă, criptograma se scrie în matrice pe coloane, începând cu colţul stânga-sus, şi apoi se realizează operaţia inversă, adică pentru fiecare indice *j* al literelor din cheie, se caută indicele *i* asociat literei din şirul sortat şi se scrie linia cu indicele *i*. Din noua matrice astfel obţinută se scriu literele de pe fiecare linie, în ordinea naturală.

Spargerea unui cifru cu transpoziţie începe cu verificarea dacă acesta este într-adevăr de acest tip prin calcularea frecventelor 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 transpoziţie. 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 spre exemplu sintagma „*săprezinte*”, 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 *m* acest număr de coloane.

Pentru a descoperi modul de ordonare a coloanelor, dacă *m* este mic, se pot considera toate posibilităţile de grupare a câte două coloane (în număr de $m \cdot (m-1)$). Se verifică dacă ele formează împreună un text corect numărând frecventele literelor şi comparându-le cu cele statistice. Perechea cu cea mai bună potrivire se consideră corect poziţionată. Apoi se încearcă, după acelaşi principiu, determinarea coloanei succesoare perechii din coloanele rămase iar apoi - a coloanei predecesoare. În urma acestor operaţii, 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).

De la apariţia cifrurilor cu substituţie şi a celor cu transpoziţie anii au trecut şi tehnicile de criptare au evoluat foarte mult. Problema construirii unui cifru imposibil de spart i-a preocupat îndelung pe criptanalişti; ei au dat o rezolvare teoretică simplă încă de acum câteva decenii dar metoda nu s-a dovedit fiabilă din punct de vedere practic, după cum se va vedea în continuare.

Tehnica propusă pentru un cifru perfect presupune alegerea unui şir aleator de biţi pe post de cheie şi aducerea textului sursă în forma unei succesiuni de biţi prin înlocuirea fiecărui caracter cu codul său ASCII. Apoi se aplică o operaţie logică - de tip SAU exclusiv (operaţia inversă echivalentei: 0 xor 0 = 0, 0 xor 1 = 1, 1 xor 0 = 1, 1 xor 1 = 0) - între cele două şiruri de biţi. Textul cifrat rezultat nu poate fi spart pentru că nu există indicii asupra textului sursă şi nici textul cifrat nu oferă criptanalistului informaţii. Pentru un eşantion de text cifrat suficient de mare, orice literă sau grup de litere (diftong, triftong) va apărea la fel de des.

Acest procedeu este cunoscut sub numele de metoda cheilor acoperitoare. Deşi este perfectă din punct de vedere teoretic, metoda are, din păcate, câteva dezavantaje practice:

- cheia nu poate fi memorată, astfel încât transmiţătorul şi receptorul să poarte câte o copie scrisă a ei fiindcă în caz că ar fi „capturaţi", adversarul ar obţine cheia;
- cantitatea totală de date care poate fi transmisă este determinată de dimensiunea cheii disponibile;
- o nesincronizare a transmiţătorului şi receptorului care generează o pierdere sau o inserare de caractere poate compromite întreaga transmisie fiindcă toate datele ulterioare incidentului vor apărea ca eronate.

5.3. Mecanisme de criptare

Sistemele de criptare pot fi aduse la un grad mai mare de securitate dacă se folosesc mijloace mecanice de criptare. Astfel de mecanisme special construite vor uşura operaţiile de criptare/decriptare şi în acelaşi timp vor fi capabile să creeze un număr mult mai mare de chei posibile.

5.3.1. Mijloace mecanice de criptare

Primele astfel de mecanisme au apărut încă în antichitate. În secolul V î.e.n. pentru criptarea datelor se folosea un baston, numit *Schitala* (figura 5.4), în jurul căruia se înfăşura spiră lângă spiră o panglică foarte îngustă de piele, papirus sau pergament pe care, pe generatoare se scriau literele mesajelor. După ce textul era scris panglica se desfăşura, mesajul devenea indescifrabil, deoarece literele erau dezasamblate. Mesajul se putea descifra numai de o persoană care dispunea de un baston de grosime şi lungime identice cu bastonul iniţial pe care se înfăşura din nou panglica primită de receptor. Astfel Schitala realiza o transpoziţie, aceasta fiind o primă formă a acestei metode de criptare. Conform istoricilor greci, acest mod de comunicare era folosit de spartani în timpul campaniilor militare. El avea avantajul de a fi rapid şi nu genera erori de transmitere. Dezavantajul însă era acela că putea fi uşor de spart.

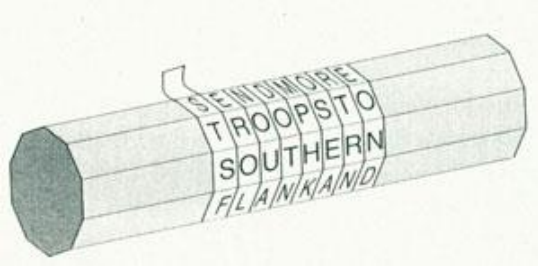

Figura 5.4. *Schitala*

Leon Battista Alberti **(**14.02.1404 – 25.04.1472) – scriitor, arhitect, pictor, sculptor, matematician, criptograf, filozof italian şi umanist al Renaşterii a inventat „*Criptograful lui Alberti*" (figura 5.5), care era alcătuit din două discuri concentrice cu diametre diferite, suprapuse. Fiecare disc era împărţit în 24 sectoare pe care erau înscrise litere şi cifre. Pe discul exterior, care rămânea static, erau scrise 20 de litere ale alfabetului italian (alfabetul italian nu avea literele *H, J, K, W, X, Y*) în ordinea lor firească, iar apoi cifrele 1, 2, 3, 4. Pe discul interior care se rotea, erau scrise 23 de litere ale alfabetului latin (fără *J, K, Y*) şi conjuncţia *ET*. Ordinea lor era arbitrară. Pentru cifrare se stabilea o cheie, de exemplu *G=a*. Aceasta însemna că pentru cifrare litera *a* de pe discul mic se aşeza în dreptul literei *G* de pe discul mare şi apoi începea cifrarea. Alberti recomanda schimbarea cheii după un număr de cuvinte.

Criptograful lui Alberti a fost perfecţionat de către Silvester, Argenti şi alţii, constituind un element de bază pentru criptografele de tip disc apărute ulterior. Silvester Porta a împărţit discurile în 26 sectoare (Figura 5.5) utilizând astfel toate cele 26 litere ale alfabetului latin (nu numai italian), criptograful său realizând astfel o substituţie simplă complet literală.

Criptograful avea două particularităţi care fac ca invenţia să fie un mare eveniment în criptografie. În primul rând acest mecanism nu era altceva decât un algoritm de criptare polialfabetică. În rândul al doilea discul respectiv permitea utilizarea aşa numitelor coduri cu recifrare, care au apărut abia la sfârşitul secolului XIX, adică peste patru secole după invenţia lui Alberti. În acest scop pe discul exterior erau scrise cifrele 1, 2, 3, 4. Alberti a compus un cod care consta din 336 grupuri de coduri numerotate de la 11 la 4444. Fiecărui cod îi corespundea o oarecare frază terminată. Când fraza se întâlnea în mesaj ea se înlocuia cu codul respectiv, iar cu ajutorul discului cifrele erau criptate ca nişte semne ordinare ale mesajului, fiind transformate în litere.

Leon Alberti poate fi considerat un criptograf ilustru şi din motivul că este autorul primei lucrări de criptologie din Europa („*De cifris*") publicată în 1946. În această lucrare erau prezentate exemple de versiuni posibile de cifrare dar şi se argumenta necesitatea aplicării criptografie în practică ca un instrument ieftin şi sigur de protecţie a informaţiei.

Criptograful lui Alberti *Criptograful lui Silvester*

Figura 5.5. *Versiuni de Criptografe*

Ideea de maşină de criptare apare clar prima dată la Thomas Jefferson, primul secretar de Stat al Statelor Unite (preşedinte era George Washington), care a inventat un aparat de criptat numit roată de criptare, folosit pentru securitatea corespondenţei cu aliaţii – în special cei francezi. Un cilindru Jefferson (figura 5.6) este format din n discuri de dimensiuni egale (iniţial $n = 26$ sau $n = 36$) aşezate pe un ax. Discurile se pot roti independent pe ax, iar pe muchia fiecăruia sunt înscrise cele 26 litere ale alfabetului, într-o ordine aleatoare (dar diferită pentru fiecare disc).

La criptare, textul clar se împarte în blocuri de n caractere. Fiecare astfel de bloc se scrie pe o linie (generatoare) a cilindrului, rotind corespunzător fiecare disc pentru a aduce pe linie caracterul căutat. Oricare din celelalte n-1 linii rămase poate constitui blocul de text criptat.

Figura 5.6. *Cilindre Jefferson*

Pentru decriptare este necesar un cilindru identic, în care se scrie pe o linie textul criptat (de n caractere) şi apoi se caută printre celelalte 25 linii un text cu semnificaţie semantică. Probabilitatea de a avea un singur astfel de text creşte cu numărul de discuri din cilindru.

O mică diferenţă apare dacă textul clar nu are nici o semnificaţie semantică (s-a folosit o dublă criptare). Atunci trebuie convenită dinainte o anumită distanţă de criptare s ($1 \leq s \leq 25$).

Ordinea discurilor poate fi de asemenea schimbată. De exemplu, un cilindru cu $n = 20$ discuri poate realiza 20! = 2 432 902 008 176 640 000 texte criptate diferite pentru acelaşi text clar. Cilindrul Jefferson realizează o substituţie polialfabetică de perioadă n. Dacă ar fi privit ca un sistem de criptare Vigenere, lungimea cheii este enormă (de multe ori n^n, în funcţie de modalităţile de aranjare a alfabetelor pe discuri). Cilindrul Jefferson a fost reinventat ulterior de mai multe ori, cea mai celebră fiind se pare maşina de criptat „*Enigma*”.

Thomas Jefferson a folosit acest aparat în perioada 1790 − 1802, după care se pare că ideea s-a pierdut. Devenit preşedinte, Jefferson a fost atras de sistemul Vigenere, pe care îl consideră mai sigur şi-l recomandă secretarului său de stat James Madison ca înlocuitor al sistemului pe care îl inventase anterior.

5.3.2. Maşini rotor

O maşină rotor (rotor machine, figura 5.7) 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 substituţie simplă. Deoarece rotoarele se mişcă cu viteze de rotaţie diferite, perioada unei maşini cu n rotoare este $n \cdot 26!$.

Aplicarea practică a acestor maşini a început numai la începutul secolului XX. Una dintre primele maşini rotor a fost maşina germană „*Enigma*", elaborată în anul 1917 de către Eduard Hebern şi perfectată mai târziu de mai multe ori. Din punct de vedere comercial ea a fost disponibilă pe piaţă încă din anul 1920, însă importanţa ei a fost dată de utilizarea maşinii de către diverse guverne, în mod special de către Germania nazistă înainte şi în timpul celui de-al doilea război mondial.

Dintre toate dispozitivele criptografice create de-a lungul timpului maşina Enigma a fost un echipament mai special din 2 puncte de vedere: criptografic și istoric.

Figura 5.7. *Modelul militar german numit Wehrmacht Enigma*

Importanţa din punct de vedere criptografic este dată de faptul că echipe de criptanalişti (matematicieni la origine) de toate naţionalităţile, în efort combinat, au încercat pe de o parte perfecţionarea maşinii, pe de altă parte spargerea cifrurilor. Printre cei care au participat la spargerea cifrului au făcut parte şi polonezul Rajewski şi britanicul Turing (inventatorul maşinilor Turing).

Importanţa istorică rezidă din rolul mare jucat de aceste maşini în timpul celui de-al doilea război mondial, mai precis faptul că descifrarea de către aliaţi a codului (nume de proiect ULTRA) a dus, după unii istorici, la scurtarea războiului cu aproximativ un an.

Construcţia maşinii. Maşina Enigma era o combinaţie de părţi mecanice şi electrice. Principalele ei componente erau, după cum urmează:

- *Tastatura*: o tastatură obişnuită similară cu cea pentru maşinile de scris.
- *Placa cu lămpi*: asemănătoare unei tastaturi cu lămpi în loc de taste. Pe lămpi erau tipărite literele alfabetului ce deveneau vizibile prin aprinderea lămpii corespunzătoare.

- *Placa cu comutatoare*: mufe (prize) câte una pentru fiecare literă, ce se conectau prin fire în 6 perechi (Această componentă fusese adăugată de germani pentru a creşte securitatea maşinii).
- *Trei roţi*: se mai numeau *rotoare* (*roţi detaşabile*) fiecare dintre ele având câte un set de 26 de contacte, câte unul pentru fiecare literă a alfabetului (figura 5.8).
- *Roata reflectoare*: roată fixă identică cu celelalte 3, având un set de 26 de contacte grupate în perechi.
- *Cabluri*: asigurau conexiunile între taste şi lămpi precum şi între lămpi şi primul rotor, între primul rotor şi al doilea, al doilea şi al treilea, al treilea si roata reflectoare.
- *Baterie*: pentru alimentarea circuitelor electrice.

Figura 5.8. *Rotor Enigma*

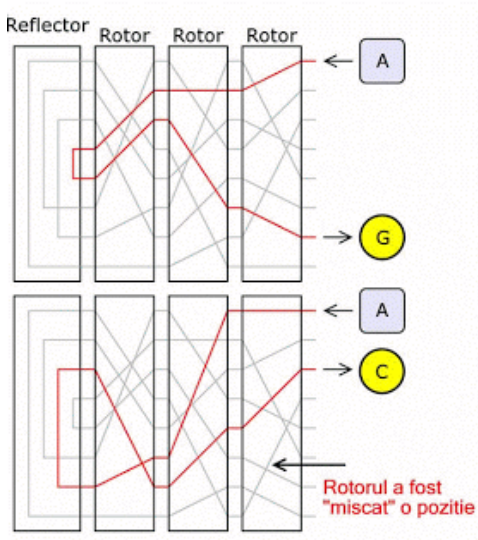

Figura 5.9. *Principiul de funcţionare al maşinii* (*Enigma*)

Principiul de funcţionare al maşinii Enigma se prezenta conform schemei din Figura 5.9. Prin apăsarea tastei ”*A*” curentul era trecut prin setul de rotoare până la reflector de unde se ”întorcea” înapoi aprinzându-se becul ”*G*”. Litera ”*A*” se criptează diferit (”*G*” si ”*C*”) doar printr-o simplă rotire a primului rotor care face ca semnalul să circule pe o rută complet diferită.

Pentru *operarea maşinii* în primul rând toţi operatorii aveau maşini identice (pentru asigurarea inter-operabilităţii). Iniţierea criptării unui mesaj se făcea în 2 paşi:

- *Pasul* 1: setarea maşinii – operaţie ce consta în fixarea ordinei şi poziţiei fiecărui rotor precum şi alegerea celor 6 perechi de conectori prin placa cu comutatoare (switch board).
- *Pasul 2*: scrierea propriu-zisă a mesajului – pentru criptarea mesajului operatorul apăsa pe tasta corespunzătoare primei litere din textul necodat (să zicem ”*N*”). În acest moment se aprindea o lampă (să zicem ”*T*”) corespunzătoare codificării. Repetând şi pentru celelalte litere, rezulta textul codat.

Trebuie de menţionat că toate setările din *Pasul* 1 erau înscrise în manuale de operare (code books), setări ce se schimbau de regulă zilnic. Fiecare operator avea câte un exemplar. De fapt, aceste setări constituiau cheia criptosistemului Enigma. Un atribut extrem de important al maşinii *Enigma* era că cheile de cifrare şi cele de decifrare erau aceleaşi. Cu alte cuvinte dacă la ”transmitere” ”*N*” se transforma în ”*T*”, la ”destinaţie ”*T*” se transforma în ”*N*” (folosind bine-nţeles aceleaşi setări ale maşinii).

Utilizarea intensivă colaborată cu posibilitatea transmiterii informaţiei folosind aceleaşi *day key* la care se adăugau intensele activităţi de contraspionaj i-au condus pe germani la teama că maşina ar putea fi compromisă. Efectul, a fost introducerea unui *protocol*. Acesta spunea: „*fiecare operator va transmite suplimentar, înaintea mesajului propriu zis, o cheie a mesajului* (*message key*)”. Aceste chei erau cuvinte (nu neapărat cu sens) formate din 3 litere alese în mod aleator de operatorul maşinii. Cu alte cuvinte, operatorul trebuia să seteze maşina conform instrucţiunilor zilnice din manualul de operare (*code book*), după care trimitea cheia din cele trei litere alese aleator. În aşa fel maşina se seta într-un mod complet aleator. Condiţiile radio proaste, lucrul sub presiune, precum şi alte condiţii de lucru nefavorabile puteau conduce la transmiterea (sau recepţionarea) greşită (alterată) a cheii, fapt ce ar fi făcut inutilă transmiterea mesajului propriu-zis (evident, datorită faptului că maşinile de la transmiţător şi cea de la receptor ar fi fost setate diferit). Pentru a minimiza astfel de incidente, operatorilor li s-a cerut să transmită cheia de 2 ori. De exemplu cheia *hot* se transmitea *hothot* și se recepţiona *dugraz*. Însă în mod ironic, ceea ce se dorea o măsură de securitate în plus, de fapt a compromis maşina.

Matematic vorbind, mulţimea cheilor posibile era atât de mare încât nici nu se punea problema ”atacării” maşinii, cel puţin nu la aceea vreme, prin metoda exhaustivă („*brute-force*”). Enigma a fost elaborată astfel încât securitatea să fie păstrată chiar dacă inamicul cunoaşte schemele rotoarelor, cu toate că în practică setările erau secrete. Cu o schemă secretă de setare cantitatea totală a configurărilor posibile era de ordinul 10^{114} (circa 380 biţi) iar dacă schema şi alte setări operaţionale erau cunoscute acest număr se reducea la 10^{23} (76 biţi). Germanii credeau că maşina Enigma este una infailibilă datorită imensităţii setărilor posibile ce i se puteau aplica. Era ireal să începi măcar să alegi o configurare posibilă.

Din punct matematic de vedere transformarea Enigmei pentru fiecare literă este rezultatul matematic a permutărilor. Pentru un aparat cu trei rotoare fie P transformarea pe tabela de prize, U - reflectorul, şi L, M, R - acţiunea rotorului din stânga, din mijloc, dreapta respectiv. Atunci criptarea E poate fi notată cu:

$$E = PRMLUL^{-1}M^{-1}R^{-1}P^{-1}$$

După fiecare apăsare de tastă rotoarele se rotesc, schimbând transformarea. De exemplu dacă rotorul de dreapta R e rotit cu i poziţii, transformarea devine: $\rho^{i}R\rho^{-i}$, unde ρ este permutarea

ciclică. Similar, rotorul din mijloc şi cel din stânga pot fi reprezentate ca *j* şi *k* rotaţii respectiv a lui *M* şi *L*. Funcţia de criptare poate fi descrisă astfel:

$$E = P(\rho^{i}R\rho^{-i})(\rho^{j}M\rho^{-j})(\rho^{k}L\rho^{-k})U(\rho^{k}L^{-1}\rho^{-k})(\rho^{j}M^{-1}\rho^{-j})(\rho^{i}R^{-1}\rho^{-i})P^{-1}$$

Pentru elucidarea funcţionării maşinii Enigma este sugestivă simularea (în flash) de la www.enigmaco.de/enigma (figura 5.10).

Primele spargeri ale maşinii Enigma au avut loc la începutul anilor 30 de către matematicienii polonezi *Alicen Rejewski, Jerzy Rozycki* şi *Henryk Zygalsk*. Cu noroc şi intuiţie Rejewski şi echipa lui au reuşit să compromită maşina, totul fiind posibil nu datorită vreunei ”scăpări” în proiectarea maşinii ci deciziei nemţilor de a transmite repetitiv (de 2 ori) cheia.

Ulterior Enigma a fost perfecţionată, spargerea ei devenind practic imposibilă pentru acele timpuri. Un aport considerabil în direcţia spargerii acestei maşini a avut Alan Turing, care proiectase o maşină electromecanică (denumită „*Bombe*” după modelul original polonez) ce putea ajuta la spargerea maşinii Enigma mai rapid decât „bomba” din 1932 a lui *Rejewski,* din care s-a şi inspirat. „Bombe” (figura 5.11), cu o îmbunătăţire sugerată de matematicianul Gordon Welchman, a devenit una din principalele unelte automate utilizate pentru a ataca traficul de mesaje protejat de Enigma.

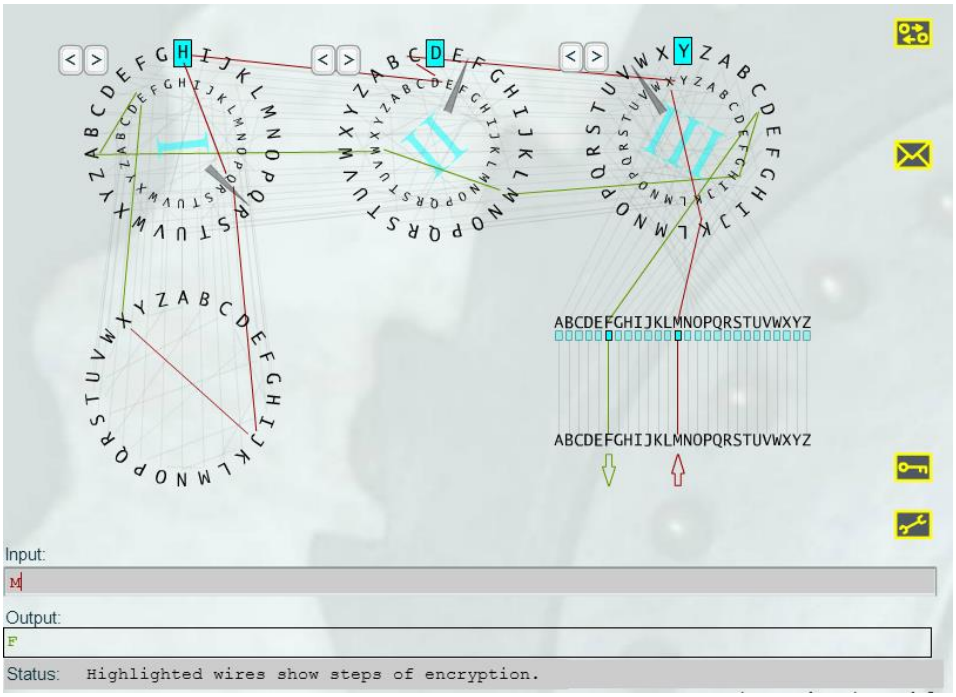

Figura 5.10. *Simulator maşina Enigma*

Maşina „Bombe” căuta setări potenţial corecte pentru un mesaj Enigma (adică, ordinea rotoarelor, setările rotoarelor, etc.), folosind un fragment de text clar probabil. Pentru fiecare setare posibilă a rotoarelor (numărul maxim posibil fiind de ordinul a 1019 stări, sau 1022 pentru maşinile Enigma de la U-boat, care aveau patru rotoare, faţă de maşina Enigma standard care avea doar trei). Aceasta efectua un lanţ de deducţii logice pe baza fragmentului probabil, deducţii implementate electric. „Bombe” detecta când avea loc o contradicţie, şi elimina setarea, trecând la următoarea.

Peste două sute de astfel de maşini create de Alan Turing au fost în funcţiune până la sfârşitul războiului.

Maşinile cu rotor au fost folosite activ pe parcursul războiului II mondial. Pe lângă maşina germană Enigma au fost folosite şi Sigaba (SUA), Typex (Marea Britanie), Red, Orange, Purple (Japonia). Maşinile cu rotor au fost vârful criptografiei formale deoarece realizau cifruri suficient de rezistente într-un mod relativ simplu.

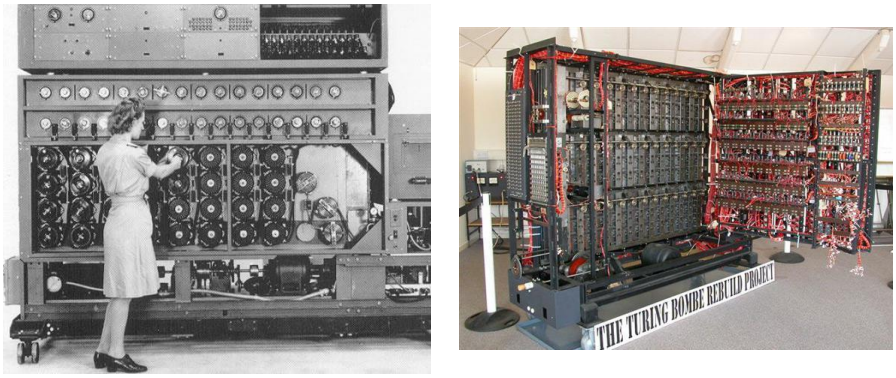

Figura 5.11. *Maşina BOMBE* (*Alan Turing*)

Atacurile încununate de succes asupra maşinilor cu rotor au fost posibile numai la începutul anilor 40 odată cu apariţia maşinilor electronice de calcul. Tot în această perioadă criptografia devine ştiinţific ramură aparte a matematicii odată cu publicarea (anul 1949) articolului lui Claude Elwood Shannon[2] „Communication Theory of Secrecy Systems", care a pus bazele ştiinţifice ale sistemelor de criptare cu cheie secretă (sistemelor simetrice).

5.4. Atacuri criptografice

5.4.1. Noţiune de criptanaliză şi atac criptografic

Atacul asupra securităţii unui sistem criptografic defineşte orice acţiune ce compromite securitatea acelui sistem. Atacurile criptografice pot fi îndreptate împotriva:

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

După modelul de atacare al unui atacator, aceste atacuri se pot clasifica după cum urmează:

a) atacuri pasive (de intercepţie):
   - de înregistrare a conţinutului mesajelor;
   - de analiză de trafic;

b) atacuri active:
   - de întrerupere (atac la disponibilitate);
   - de modificare (atac la integritate);

[2] *Claude Elwood Shannon* (30 aprilie 1916 - 24 februarie 2001) a fost un matematician american, inginer electrotehnist și criptograf cunoscut sub numele de „tatăl teoriei informației".

- de fabricare(atac la autenticitate).

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

1. de înregistrare a conţinutului mesajelor;
2. de analiză a traficului.

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 detecţia lor.

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

Dintre atacurile active se regăsesc:

- întreruperea serviciului;
- modificarea;
- modificări în program astfel încât acesta va lucra diferit;
- modificarea conţinutului mesajelor transmise în reţea;
- fabricarea: un neavizat inserează informaţii false în sistem.

Atacurilor active pot fi caracterizate prin faptul că deşi pot fi detectate, prevenirea lor este foarte grea, deoarece ar însemna protecţie fizică permanentă a întregului sistem.

Atacurile amintite mai sus sunt, în principiu, atacuri mai generale, nu neapărat specifice exclusiv algoritmilor criptografici, iar o analiză mai detaliată a unora dintre ele va fi făcută în compartimentul 9.

Atacurile specifice algoritmilor criptografice sunt atacurile de criptanaliză.

Criptanaliza este ştiinţa spargerii cifrurilor şi se ocupă de obţinerea valorii iniţiale a informaţiei criptate fără a avea acces la informaţia secretă, adică la cheia necesară pentru acest lucru. Persoana care se ocupă cu criptanaliza se numeşte *criptanalist*. Atacul criptografic bazat pe metode de criptanaliză se mai numește atac criptanalitic.

Analiza criptografică studiază metode de atac pornind de la informaţii minimale despre cheile de criptare, algoritmii utilizați, protocoalele de autentificare, segmente de text clar şi segmentele corespondente din textul criptat, sau doar pe baza unuia sau a unui set de texte criptate utilizând același algoritm.

Scopul metodelor de criptanaliză este descoperirea mesajelor în clar şi/sau a cheii din mesajul criptat. Orice cifru este creat în scopul de a asigura confidențialitatea informației protejate și, reieșind din acest concept, întotdeauna se vor găsi oameni care doresc să obţină accesul

la informaţia dată. În esență, se încearcă determinarea unui punct vulnerabil al algoritmului, care să poată fi exploatat folosind metode pentru care timpul de căutare să fie considerabil mai mic decât timpul necesar verificării tuturor combinațiilor de chei posibile (atac cu forţa brută).

Nu există încă un sistem criptografic despre care să se poată afirma că este pe deplin sigur, dar pot fi considerate sigure acele criptosisteme pentru care atacurile cunoscute necesită un timp mult prea îndelungat pentru a putea fi considerate practice.

5.4.2. Tipuri de atacuri criptografice

Se cunosc mai multe tipuri de atacuri criptografice. O categorie aparte sunt atacurile ce se realizează prin forţă brută (*brute force*), adică aplică o metodă exhaustivă de căutare prin încercarea tuturor combinaţiilor posibile fie de chei de criptare, fie de simboluri din text pentru deducerea textului în clar (de exemplu, la metodele de criptare prin substituţia sau transpoziţia literelor din mesaje de tip text). Complexitatea acestui atac este în funcţie de cantitatea tuturor variantelor posibile.

Acest tip de atac este unul general, adică poate fi aplicat la orice algoritm de criptare. Din acest motiv în elaborarea sistemelor de criptare autorii încearcă să obţină ca acest atac să fie cel mai eficient, comparativ cu celelalte metode de spargere. Sistemul se proiectează astfel încât forţa brută să aibă un spaţiu al soluţiilor suficient de voluminos pentru ca rezultatul aplicării forţei brute să nu fie obţinut pe parcursul a câtorva ani, sau uneori şi secole.

În baza complexităţii aplicării forţei brute se face evaluarea securităţii sistemului. În particular, cifrul se consideră sigur dacă nu există o metodă de spargere semnificativ mai rapidă decât forţa brută.

Atacurile criptografice bazate pe forţa brută sunt cele mai universale, dar şi cele mai îndelungate. În legătură cu aceasta există deja şi se mai elaborează încontinuu posibilităţi de optimizare a metodei forţei brute. Printre aceste metode optimizate se numără următoarele:

- metoda ramifică şi mărgineşte (*branch and bound method*);
- metoda calculelor paralele (*parallel calculation method*).

Metoda *ramifică şi mărgineşte* reprezintă un algoritm de căutare a soluţiilor optime pentru diverse probleme. Esenţa lui constă în separarea submulţimii de soluţii admisibile care nu conţine soluţii optime. Metoda a fost propusă pentru prima dată de către A. H. Land şi A. G. Doig în 1960 pentru programarea discretă.

Calculul paralel este execuția în paralel pe mai multe procesoare a aceloraşi instrucțiuni, sau și a unor instrucțiuni diferite, cu scopul rezolvării mai rapide a unei probleme, de obicei special adaptată sau subdivizată. Ideea de bază aici constă în divizarea mulţimii soluţiilor în N submulţimi, fiecare din ele fiind mult mai „mică” decât originalul, astfel realizarea „forţei brute” va necesita de n ori mai puţin timp, în funcţie de numărul de submulţimi ale partiţiei făcute. Problema

fundamentală aici constă în determinarea şi divizarea mulţimii soluţiilor. „Forţa brută” se aplică până când un procesor nu a găsit soluţia (cheia) necesară.

De rând cu forţa brută se mai aplică şi *metodele statistice* de atac. Aceste metode se divizează în două subcategorii:

- metode de criptanaliză a proprietăţilor statistice ale gamei de criptare;
- metode de criptanaliza a complexităţii şirului.

Prima subcategorie studiază şirurile la ieşirea algoritmilor de criptare. În acest caz criptanalistul cu ajutorul diverselor teste statistice încearcă să găsească valoarea următorului bit al şirului cu o probabilitate mai mare decât probabilitatea alegerii aleatoare.

În cazul al doilea criptanalistul încearcă să genereze şiruri analogice cu gama însă aplicând metode mult mai simple.

Din diversitatea de tipuri şi metode de atac criptografic, demonstrate, verificate şi aplicate de matematicieni, informaticieni și criptanaliști trebuie menţionate atacurile cu text clar sau text cifrat:

- *Atac cu text cifrat* (*ciphertext-only attack*) interceptat, prin analiza căruia se încearcă găsirea textului original sau a cheii de criptare. Acest atac se bazează pe informaţii referitoare la secvenţe de text cifrat și este una dintre cele mai dificile metode criptografice din cauza informaţiei sumare pe baza căreia trebuie să se deducă informaţii referitoare la textul clar sau la chei.
- *Atac cu text clar cunoscut* (*known plaintext attack*), este un atac în care criptanalistul are acces nu doar la textul cifrat, ci şi la textul clar corespunzător. El încearcă să descopere o corelație între cele două pentru a găsi cheia de criptare sau pentru a crea un algoritm care îi va permite să descifreze orice mesaje criptate cu această cheie. Textele în clar necesare pentru acest atac pot fi obţinute prin diverse metode, de exemplu dacă se cunoaşte că se trimite un fişier cifrat cu un nume ştiut, atunci din extensia fişierului se pot face concluzii despre conţinutul anumitor fragmente ale fişierelor, de exemplu a heeder-ului. Acest atac este mai puternic decât atacul cu text cifrat.
- *Atac cu text cifrat ales* (*chosen ciphertext attack*), este un atac în care criptanalistul alege un text cifrat și încearcă să găsească textul clar potrivit. Acest lucru se poate face cu o decriptare oracul (o mașină care decriptează fără a demasca cheia). Atacul este aplicat la criptarea cu cheie publică – se începe cu un text cifrat și căutări de potrivire de date ale textului clar postate public.
- *Atac cu text clar ales* (*chosen plaintext attack*), este un atac în care criptanalistul poate cripta un text clar la alegerea sa și studia textul cifrat rezultat. Scopul criptanalistului este acelaşi ca la atacul cu text clar cunoscut: de a afla cheia de criptare sau de a găsi o altă metodă pentru descifrarea mesajelor cifrate cu aceeaşi cheie. Însă criptanalistul trebuie să mai aibă

posibilitatea de a alege câteva texte în clar şi să obţină rezultatul cifrării lor. Obţinerea textului cifrat respectiv pentru textul clar dat uneori se poate face prin crearea şi transmiterea unui mesaj necifrat în numele unuia din utilizatorii care folosesc criptarea. În cazul coincidenţii unor factori acest mesaj poate fi cifrat şi retransmis înapoi. Acest atac este cel mai des utilizat în criptografia asimetrică, în cazul în care criptanalistul are acces la o cheie publică.

- *Atac cu text clar ales adaptiv* (a*daptive chosen plaintext attack*), este un caz particular, mai confortabil, al atacului cu text clar ales. Confortul atacului constă în faptul că pe lângă posibilitatea alegerii textului clar, criptanalistul poate lua decizia de a cifra un oarecare text clar în baza operaţiilor de cifrare deja efectuate. Cu alte cuvinte la realizarea atacului cu text clar ales criptanalistul alege doar un bloc mare al textului clar pentru a fi cifrat şi apoi, în baza datelor obţinute începe spargerea sistemului. În cazul organizării atacului cu text clar ales adaptiv criptanalistul poate obţine rezultatul cifrării oricărui bloc al textului clar pentru a acumula datele care îl interesează şi care vor fi luate în seamă la alegerea ulterioară a blocurilor textului clar etc. Anume adaptivitatea (posibilitatea feedback-ului) îi dă un avantaj atacului cu text clar ales adaptiv.
- *Atac cu text cifrat ales adaptiv* (*Adaptive Chosen Ciphertext Attack*), este un atac analogic atacului cu text clar ales adaptiv.

  În continuare vom menţiona alte tipuri şi metode de tac utilizate în prezent.

Atacul de tip dicţionar (*dictionary attack*) este o metodă criptanalitică în care atacatorul pregăteşte și memorează un tabel cu corespondenţe text clar - text criptat de tipul perechilor ($P_iC_i=E_{Ki}(P)$, K_i) sortate după C_i. Ulterior, atacatorul monitorizează comunicaţia şi în momentul în care va găsi un text criptat C_j care se regăseşte în tabelul său va găsi imediat cheia de criptare K_j.

Atacul zilei de naştere (*Birthday attack*), se bazează pe cunoscutul paradox al „zilei de naştere” şi a variantelor sale (într-un grup de 23 de persoane, probabilitatea să existe două dintre ele născute în aceeaşi zi din an este peste 0,5; în general, într-un grup de $\sqrt{k}$ numere aleatoare având k valori posibile, probabilitatea ca cel puţin două să fie egale este în jur de 0,5). Problema poate fi astfel generalizată: dacă o funcţie f: $A \rightarrow B$ poate lua oricare din cele n valori din mulţimea B cu probabilităţi egale, atunci după calculul funcţiei pentru $\sqrt{n}$ valori diferite este foarte posibil să găsim o pereche de valori x_1 și x_2 astfel încât $f(x_1) = f(x_2)$. Evenimentul reprezintă o coliziune, iar pentru funcţii cu distribuţie impară, coliziunea poate apărea şi mai devreme. Semnătura digitală este susceptibilă de a fi supusă unui astfel de atac.

Atac cu întâlnire la mijloc (*Meet-in-the-middle attack*) este similar cu atacul zilei de naştere, cu excepţia faptului că în acest caz analistul are o flexibilitate mai mare. În loc să aştepte

coincidenţa a două valori într-o singură mulţime de date, analistul poate căuta o intersecţie a două mulţimi.

Presupunem că atacatorul cunoaşte o mulţime de texte în clar P și texte criptate C cu cheile k_1 și k_2. Atunci el poate calcula $E_K(P)$ pentru toate cheile posibile K și să memoreze rezultatele, apoi poate calcula $D_K(C)$ pentru fiecare K şi să compare cu rezultatele memorate - dacă va găsi o coincidenţă este ca și cum ar fi găsit cele două chei şi poate verifica direct pe textul în clar şi cel criptat. Dacă dimensiunea cheii este n, atacul va folosi doar 2^{n+1} criptări în contrast cu un atac clasic, care ar avea nevoie de 2^{2n} criptări.

Atacul omului din mijloc (*Man-in-the-middle attack*) descrie situaţia când un atacator are posibilitatea să citească şi să modifice mesajele schimbate între doi corespondenţi fără ca cele două părţi să sesizeze faptul că metoda de comunicare între ei a fost compromisă.

Atacul începe de obicei cu ascultarea canalului şi se termină cu încercarea criptanalistului de a înlocui mesajul interceptat, extragerea informaţiilor utile din el, redirecţionarea mesajului la unele resurse externe. Fie că subiectul *A* planifică transmiterea spre subiectul *B* a unei informaţii oarecare. Subiectul *C* posedă cunoştinţe despre structura şi proprietăţile metodei de transmitere a datelor, precum şi a însuşi faptului transmiterii acelei informaţii pe care intenţionează să o intercepteze (de exemplu cheia privată). Pentru săvârşirea atacului *C* se „prezintă” lui *A* drept *B*, iar lui *B* drept *A*. Subiectul *A* consideră eronat că transmite informaţia lui *B* şi o trimite lui *C*, care la rândul său efectuează unele operaţii cu ea (o copiază sau o modifică în scopuri personale) şi o transmite lui *B*. Ultimul consideră că informaţia a fost primită direct de la *A*.

Posibilitatea unui astfel de atac rămâne o problemă serioasă pentru sistemele bazate pe chei publice.

Atacul în reluare (*replay attack*) este un atac în care atacatorul memorează o sesiune de comunicare în ambele sensuri (mesajele schimbate de ambii corespondenţi) sau bucăţi din sesiune. Ideea atacului nu este de a decripta o sesiune de comunicare, ci de a crea confuzii şi mesaje false.

Atacul cu chei relaţionate (*related keys attack*). În acest caz atacatorul descoperă o relaţie între un set de chei și are acces la funcţiile de criptare cu astfel de chei relaţionate. Scopul declarat este de a găsi chiar cheile de criptare. Algoritmi ca IDEA, GOST, RC2 și TEA au prezentat slăbiciuni când au fost supuse atacului.

Atacul prin alunecare (*slide attack*) poate fi văzut ca o variantă a atacului cu chei relaţionate în care relaţiile sunt definite pe aceeaşi cheie. Atacul este eficient în cazul unor procese iterative sau recursive (algoritmi simetrici de tip şir sau bloc) care prezintă grade de similitudine între cicluri succesive ale procesului iterativ. Complexitatea atacului este independentă de numărul de cicluri ai algoritmului. Slăbiciuni în cazul acestui atac au fost relevate în algoritmul Feistel și chiar în cazul SHA-1.

Atacul de corelaţie (*correlation attack*) se efectuează asupra generatorului de filtrare din cifrurile şir bazate pe generatoare de tip LFSR (Linear Feedback Shift Register, mai detaliat despre LSFR vezi în compartimentul 6), în două faze: întâi se determină o funcţie între şirul de biţi cheie generat şi biţii registrului de deplasare, după care șirul de chei este interpretat ca o versiune afectată de zgomot a şirului generat de LFSR.

Atacul de corelaţie rapidă (*fast correlation attack*) se aplică generatoarelor de chei bazate pe LFSR, ca şi atacul de corelaţie, dar sunt mai rapide și exploatează existenţa unei corelaţii între şirul de chei şi ieşirea unui LFSR, numit LFSR ţintă, a cărui stare iniţială depinde de anumiţi biţi ai cheii secrete.

Atacurile de corelaţie rapidă evită examinarea tuturor iniţializărilor posibile ale LFSR-ului ţintă folosind anumite tehnici eficiente de corectare a erorii. Astfel, descoperirea stării iniţiale a LFSR-ului constă în decodarea subşirului de chei relativ la codul FSR-ului.

Atacul prin interpolare (*interpolation attack*) este o tehnică de atac asupra cifrurilor simetrice bloc construite din funcţii algebrice simple. Dacă textul criptat este scris ca un polinom, funcţie de elementele textului clar şi gradul polinomului este suficient de mic, atunci un număr limitat de perechi de text clar/criptat sunt suficiente pentru determinarea funcţiei de criptare. Acest lucru permite atacatorului să cripteze sau să decripteze blocuri de date fără a recupera propriu zis cheia de criptare. Atacul a fost introdus în 1997 şi aplicat prima dată pe o variantă a algoritmului SHARK, un predecesor al algoritmului Rijndael, care stă la baza standardului AES. Acest tip de atac poate fi generalizat, astfel că ideea interpolării poate fi aplicată şi în cazul unor polinoame probabilistice.

Atacul „divide şi cucereşte” (*divide and conquer attack*). Atacurile din această categorie încearcă diviziunea cheilor în bucăţi mai mici, pentru a face posibilă căutarea exhaustivă. Acest tip de atac este eficient în măsura în care este posibil să determinăm bucăţi separate din chei. Problema constă în validarea sau invalidarea unui segment de cheie, fără a avea informaţii despre restul cheii.

Atacul temporal (*timing attack*). Durata de execuţie a unui echipament hardware de criptare poate furniza informaţii despre parametrii implicaţi astfel încât analiza atentă şi măsurarea timpului de execuţie poate duce, în anumite condiţii, la recuperarea cheilor secrete. Pentru a putea realiza un astfel de atac, atacatorul are nevoie de un set de mesaje împreună cu durata lor de procesare pe echipamentul criptografic. Metodele de măsurare a timpului sunt diverse: monitorizarea activităţii procesorului, măsurarea timpului într-o secvenţă de interogare/răspuns etc. Atacul a fost aplicat algoritmilor RSA și RC5, dar şi unor protocoale de Internet de tipul SSL.

5.4.3. Criptanaliza liniară și diferențială

În continuare sunt expuse două tehnici de criptanaliză - *criptanaliza liniară* şi *criptanaliza diferenţială* - care la momentul actual sunt unele dintre cele mai răspândite metode de spargere a cifrurilor bloc.

Criptanaliza liniară (*linear cryptanalisis*) este o tehnică introdusă de Matsui si Yamagishi în 1991 (*A new Method for known plain text attack of FEAL cipher*) care încearcă să exploateze apariţiile cu probabilitate mare ale expresiilor liniare ce implică biţi de text clar, biţi de text criptat şi biţi ai subcheilor. În acest caz se presupune că atacatorul cunoaşte un set aleator de texte clare, precum și textele criptate corespunzătoare. Se aplică algoritmilor simetrici, de tip bloc și a fost utilizată cu succes în criptanaliza algoritmului DES – precedentul standard decriptare simetrică în SUA. Sunt elaborate atacuri pentru cifrurile bloc şi cele de tip flux. Descoperirea criptanalizei liniare a constituit un imbold pentru elaborarea noilor scheme de criptare.

Criptanaliza liniară este o tehnică prin care se urmăreşte construcţia unui sistem de ecuaţii liniare între biţii textului clar, ai textului cifrat şi ai cheii. Rezolvarea acestui sistem de ecuaţii duce la aflarea cheii de cifrare. Sistemul de ecuaţii ce se construieşte poate fi chiar un sistem probabilist, în sensul că o ecuaţie este verificată cu o anumită probabilitate.

Criptanaliza se face în două etape. Prima - construirea relațiilor dintre textul clar, textul cifrat și cheie, care sunt adevărate cu o probabilitate mare. A doua - utilizarea acestor relații, împreună cu perechile cunoscute de text clar şi text cifrat pentru obţinerea biților cheii.

La momentul actual pentru orice cifru nou este necesar de demonstrat că el este rezistent la criptanaliza liniară.

Criptanaliza diferenţială (*diferential cryptanalysis*) este o tehnică introdusa de Biham şi Schamir prima dată în 1991 și concretizată în „Diferential Cryptanalysis of the Data Encryption Standard” publicat în 1993. Tehnica face parte din categoria atacurilor cu text clar ales. În general fiind dată perechea (D, D'), numită caracteristică, tehnica constă în generarea de texte clare (P_1, P_2) cu $D = P_1 - P_2$ astfel încât $D' = C_1 - C_2$ şi aflarea unei părţi a cheii, restul fiind căutată exhaustiv. Operația „–” este operaţia de grup care, spre exemplu în cazul cifrurilor bloc, constă în adunarea cheii de runda.

Criptanaliza diferenţială exploatează apariţiile cu mare probabilitate a diferenţelor din textele clare, precum și a diferenţelor apărute în ultimul ciclu al criptorului în care atacatorul poate selecta intrările și examina ieşirile în încercarea de a deduce cheia.

În final trebuie de subliniat că la elaborarea sistemelor de criptare trebuie să fie luate în considerare performanţele criptanalizei pentru a diminua riscurile posibile.

Întrebări și subiecte pentru aprofundarea cunoștințelor și lucrul individual

1. Care este locul criptografiei în procesul de asigurare a securității informației?

2. Numiți și comentați cele cinci componente ale unui sistem de criptare a informației.
3. Faceți o clasificare a tipurilor de algoritmi de criptare. Care sunt avantajele și dezavantajele fiecărui tip de cifru?
4. Fie mesajul *M=NumePrenume*. Utilizând algoritmii de criptare Cezar, Afin, Polibios, Playfair, Vigenere și de transpoziție realizați criptarea și decriptarea mesajului *M*, alegând cheia de criptare potrivită (înlocuiți *NumePrenume* cu cele personale).
5. Realizați un studiu pentru a comenta argumentat efectele utilizării mașinii Enigma în cel de-al Doilea Război Mondial.
6. Descrieți etapele realizării unui atac criptografic bazat pe analiza frecvenței.