

David NACCACHE

Emil SIMION

Adela MIHĂIȚĂ

Ruxandra-Florentina OLIMID

Andrei-George OPRINA

# Criptografie și securitatea informației



## Aplicații

David NACCACHE, Emil SIMION, Adela MIHĂIȚĂ, Ruxandra-Florentina OLIMID, Andrei-George OPRINA - Criptografie și securitatea informației. Aplicații



9789737556752

**MATRIX**  
ROM  
BUCUREȘTI

# Criptografie și Securitatea Informației. Aplicații.

Colectivul de coordonare  
David Naccache    Emil Simion

Colectivul de autori  
Adela Georgescu [cap.]  
David Naccache [cap.]  
Ruxandra-Florentina Olimid[cap.]  
Andrei-George Oprina [cap.]  
Steluța Pricopie [cap.]  
Emil Simion [cap.]



## Prefață

Intrând progresiv în era informației, societățile industrializate se găsesc în fața unui paradox: pe de o parte, puterea și influența Europei și a Americii de Nord au crescut semnificativ, în principal datorită măiestriei modalităților prin care se controlează fluxurile de informații, precum și valorii crescute a datelor procesate. Pe de altă parte, după cum au demonstrat-o deja criza Wikileaks sau viermele Stuxnet, apar noi amenințări și vulnerabilități care fac ca dependența noastră de sistemele informaționale să fie crucială.

De aceea, dezvoltarea atacurilor cibernetice, precum și disponibilitatea online a instrumentelor utilizate în activitatea de piraterie conduce la obiective strategice importante și cultivă necesitatea de a pregăti experți pentru acest domeniu.

Criptografia este peste tot în jurul tău. În timp ce tu citești aceste rânduri, în vecinătatea ta se transmit informații cifrate prin telefoane mobile, relee de pay-TV, precum și routere wireless. Mediul în care trăim se schimbă într-un ritm alert. Această evoluție este rezultatul progresului în domeniul tehnologiilor hardware și al matematicii.

Criptografia aplicată s-a dezvoltat considerabil în ultimii ani, pentru a putea satisface cerințele crescute de securitate ale diverselor domenii legate de tehnologia informației, cum ar fi telecomunicațiile, rețelistica, bazele de date, precum și aplicațiile de telefonie mobilă. Sistemele criptografice sunt din ce în ce mai complexe și mai tehnice și necesită din ce în ce mai multă putere de calcul (de exemplu schema de cifrare perfect homomorfă a lui Gentry). În plus, algoritmi criptografici trebuie utilizați împreună cu protocoale adecvate, a căror proiectare și înțelegere necesită o analiză delicată.

Această carte vă oferă instrumentele necesare pentru a începe să vă dezvoltați aptitudinile în domeniul criptografiei. În timp ce citiți aceste rânduri în limba română, străinul care sunt vă îndeamnă să realizați că unele dintre cele mai luminate minți care au adus contribuții acestui domeniu își aveau originile în spațiul lingvistic și cultural românesc. De exemplu, cel care a spart mașina de cifrat "Purple" a japonezilor, faptă care a dus la divulgarea secretelor diplomatice japoneze înainte de intrarea Americii în cel de-al doilea război mondial, provenea din orașul Chișinău, Republica Moldova, oraș în care familia lui se mutase după plecarea din București la sfârșitul anilor 1890. Știința secretelor are o lungă tradiție în România, țară care a fost nevoită constant să se bazeze pe propriile talente pentru a-și păstra independența. Experții au prezis că următoarele războaie vor începe în spațiul cibernetic. Autorii acestei cărți, care sunt pedagogi și cercetători, au importanta datorie morală de a lăsa moștenire României astfel de talente vitale.

În trecut, am avut onoarea de a cunoaște sau a fi mentorul unor cercetători și studenți români foarte talentați. Întotdeauna am fost uimit de creativitatea acestora, de dorința lor de a-și atinge scopurile, precum și de dăruirea pentru muncă. Sper că această carte va contribui la dezvoltarea continuă de asemenea talente, astfel încât domeniul științific căruia i-am dedicat o bună parte a vieții mele să beneficieze de acest formidabil rezervor de talente.

Dacă sunteți un student talentat și interesat de studii doctorale în domeniu, nu ezitați să mă contactați pentru sfaturi.

Prof. David Naccache

Université Paris II, Pantheon-Assas, PRES Sorbonne Universités

Membru al laboratorului informatic al Ecole normale supérieure. Paris, Franța



## Cuvânt înainte

Lucrarea de față conține aplicații practice abordate de autori în cadrul seminariilor ce se desfășoară la disciplina *Criptografie și Securitate*, la Facultatea de Matematică Informatică din cadrul Universității din București, la masterul de *Securitatea Tehnologiei Informației*, organizat de Academia Tehnică Militară, precum și la masterul de *Teoria Codării și Stocării Informației*, organizat de Facultatea de Științe Aplicate din cadrul Universității Politehnica București.

Această culegere de probleme este un prim pas în dezvoltarea colaborării dintre școala românească de criptologie și școala franceză reprezentată în cazul de față de David Naccache, profesor la universitatea Pantheon-Assas Paris II. Din acest motiv se regăsesc, în culegerea de față, capitolele dedicate principiilor criptologice și atacurilor în mediul de implementare, ce acoperă un gol din curricula sistemului de învățământ din România, capitole elaborate în colaborare cu profesorul David Naccache.

Materialul este structurat în capitole independente, fiecare capitol fiind constituit din trei părți: prezentarea metodei (breviar teoretic), exemple de aplicare și probleme propuse spre rezolvare, pentru fiecare dintre acestea indicându-se rezultatul ce trebuie obținut.

Întrucât criptografia este o disciplină computațională, autorii au considerat utilă introducerea unui capitol special dedicat aplicațiilor software care pot constitui logistica necesară desfășurării în bune condiții a laboratoarelor la această disciplină.

În continuare considerăm util să definim unele dintre principalele noțiuni utilizate în cadrul acestei culegeri de probleme.

*Criptologia* este știința scrierilor secrete, având drept obiect apărarea secretului datelor și informațiilor confidențiale, cu ajutorul sistemelor criptografice.

*Criptografia* este latura defensivă a *criptologiei*, având drept obiect de activitate elaborarea (conceperea) sistemelor criptografice și a regulilor folosite.

*Criptanaliza* este latura ofensivă a *criptologiei*, având drept obiect de activitate studierea sistemelor criptografice proprii pentru a le oferi caracteristicile necesare, astfel încât acestea să-și îndeplinească funcția pentru care au fost concepute. Totodată criptanaliza poate analiza sistemele criptografice ale terțelor părți (prin intermediul criptogramelor realizate cu ele) astfel încât prin spargerea acestora să obțină informații utile instituției pe care o deservește.

Prin *algoritm criptografic* înțelegem o mulțime de transformări uniinversabile prin care mulțimea mesajelor (textelor) clare dintr-o limbă se transformă în mulțimea  $\mathcal{M}$  a criptogramelor.

*Cheia de cifrare* constituie o convenție particulară, materializată, printr-un cuvânt, frază, număr, șir numeric etc. și care dirijează (reglementează) *operația de cifrare*.

Un *protocol criptografic* este un set de reguli, între doi sau mai mulți parteneri, prin intermediul căruia are loc o operație de autentificare și/sau transfer de cheie sau mesaje.

Un *sistem criptografic* este compus din trei elemente: algoritm de cifrare, sistem de generare al cheilor și protocol de distribuție al cheilor de cifrare.

*Descifrarea* este operația inversă cifrării și ea constă în aplicarea sistemului de cifrare cunoscut (în prezența cheii corecte) asupra criptogramelor pentru aflarea mesajului clar.

*Decriptarea* este operația prin care, numai pe baza analizei criptogramelor realizate cu un sistem de cifru necunoscut, se pune în evidență mesajul clar care a fost criptografiat și se determină caracteristicile sistemului criptografic folosit pentru cifrare.

Dr. mat. Emil Simion



# Cuprins

<b>1</b>	<b>Sistemul de cifrare Cezar</b>	<b>1</b>
1.1	Breviar teoretic . . . . .	1
1.2	Exerciții rezolvate . . . . .	1
1.3	Exerciții propuse . . . . .	2
<b>2</b>	<b>Metoda substituției</b>	<b>5</b>
2.1	Breviar teoretic . . . . .	5
2.2	Exerciții rezolvate . . . . .	6
2.3	Exerciții propuse . . . . .	8
<b>3</b>	<b>Sistemul de cifrare Playfair</b>	<b>11</b>
3.1	Breviar teoretic . . . . .	11
3.2	Exerciții rezolvate . . . . .	12
3.3	Exerciții propuse . . . . .	13
<b>4</b>	<b>Sistemul de cifrare Hill</b>	<b>17</b>
4.1	Breviar teoretic . . . . .	17
4.2	Exerciții rezolvate . . . . .	17
4.3	Exerciții propuse . . . . .	19
<b>5</b>	<b>Sisteme de cifrare polialfabetice</b>	<b>23</b>
5.1	Breviar teoretic . . . . .	23
5.2	Exerciții rezolvate . . . . .	24
5.3	Exerciții propuse . . . . .	25
<b>6</b>	<b>Metoda transpoziției</b>	<b>27</b>
6.1	Breviar teoretic . . . . .	27
6.2	Exerciții rezolvate . . . . .	27
6.3	Exerciții propuse . . . . .	28
<b>7</b>	<b>Sisteme mixte</b>	<b>31</b>
7.1	Breviar teoretic . . . . .	31
7.2	Exerciții rezolvate . . . . .	31



7.3	Exerciții propuse . . . . .	33
<b>8</b>	<b>Generatoare pseudoaleatoare</b>	<b>35</b>
8.1	Breviar teoretic . . . . .	35
8.2	Exerciții rezolvate . . . . .	37
8.3	Exerciții propuse . . . . .	37
<b>9</b>	<b>Calcul în corpuri Galois</b>	<b>39</b>
9.1	Breviar teoretic . . . . .	39
9.2	Exerciții rezolvate . . . . .	39
9.3	Exerciții propuse . . . . .	40
<b>10</b>	<b>Algoritmul RIJNDAEL - Standardul AES</b>	<b>43</b>
10.1	Breviar teoretic . . . . .	43
10.2	Exerciții rezolvate . . . . .	43
10.3	Exerciții propuse . . . . .	46
<b>11</b>	<b>Criptanaliza cifrurilor bloc</b>	<b>51</b>
11.1	Breviar teoretic . . . . .	51
11.2	Exerciții rezolvate . . . . .	51
11.3	Exerciții propuse . . . . .	53
<b>12</b>	<b>Lema chinezească a resturilor</b>	<b>55</b>
12.1	Breviar teoretic . . . . .	55
12.2	Exerciții rezolvate . . . . .	56
12.3	Exerciții propuse . . . . .	57
<b>13</b>	<b>Sistemul de cifrare Merkle-Hellman</b>	<b>59</b>
13.1	Breviar teoretic . . . . .	59
13.2	Exerciții rezolvate . . . . .	60
13.3	Exerciții propuse . . . . .	61
<b>14</b>	<b>Sistemul de cifrare RSA</b>	<b>63</b>
14.1	Breviar teoretic . . . . .	63
14.2	Exerciții rezolvate . . . . .	64
14.3	Exerciții propuse . . . . .	65
<b>15</b>	<b>Sistemul de cifrare ElGamal</b>	<b>67</b>
15.1	Breviar teoretic . . . . .	67
15.2	Exerciții rezolvate . . . . .	67
15.3	Exerciții propuse . . . . .	67

<b>16 Aritmetica pe curbe eliptice</b>	<b>69</b>
16.1 Breviar teoretic . . . . .	69
16.2 Exerciții rezolvate . . . . .	70
16.3 Exerciții propuse . . . . .	71
<b>17 Sistemul de cifrare ElGamal bazat pe curbe eliptice</b>	<b>73</b>
17.1 Breviar teoretic . . . . .	73
17.2 Exerciții rezolvate . . . . .	73
17.3 Exerciții propuse . . . . .	74
<b>18 Sistemul de cifrare Menezes-Vanstone</b>	<b>77</b>
18.1 Breviar teoretic . . . . .	77
18.2 Exerciții rezolvate . . . . .	77
18.3 Exerciții propuse . . . . .	78
<b>19 Funcții de dispersie</b>	<b>81</b>
19.1 Breviar teoretic . . . . .	81
19.2 Exerciții propuse . . . . .	83
<b>20 Semnătura ElGamal</b>	<b>85</b>
20.1 Breviar teoretic . . . . .	85
20.2 Exerciții rezolvate . . . . .	85
20.3 Exerciții propuse . . . . .	85
<b>21 Semnătura DSA/ECDSA</b>	<b>87</b>
21.1 Breviar teoretic . . . . .	87
21.2 Exerciții rezolvate . . . . .	88
21.3 Exerciții propuse . . . . .	88
<b>22 Protocolul Diffie-Hellman de stabilire a cheilor</b>	<b>91</b>
22.1 Breviar teoretic . . . . .	91
22.2 Exerciții rezolvate . . . . .	91
22.3 Exerciții propuse . . . . .	91
<b>23 Protocolul Blom</b>	<b>93</b>
23.1 Breviar teoretic . . . . .	93
23.2 Exerciții rezolvate . . . . .	94
23.3 Exerciții propuse . . . . .	94
<b>24 Protocolul Shamir de partajare a secretelor</b>	<b>95</b>
24.1 Breviar teoretic . . . . .	95
24.2 Exerciții rezolvate . . . . .	95
24.3 Exerciții propuse . . . . .	96

<b>25 Scheme de partajare a secretelor bazate pe CRT</b>	<b>97</b>
25.1 Breviar teoretic . . . . .	97
25.2 Exerciții rezolvate . . . . .	97
<b>26 Canale subliminale</b>	<b>99</b>
26.1 Breviar teoretic . . . . .	99
26.2 Exerciții rezolvate . . . . .	99
26.3 Exerciții propuse . . . . .	99
<b>27 Principii criptografice</b>	<b>101</b>
<b>28 Atacuri în mediul de implementare</b>	<b>105</b>
28.1 Breviar teoretic . . . . .	105
28.2 Exerciții propuse . . . . .	106
<b>29 Resurse software</b>	<b>107</b>
29.1 CrypTool . . . . .	107
29.2 OpenSSL . . . . .	109
29.3 Studiu de caz: Implementarea funcțiilor criptografice în MAPLE . . . . .	111
29.4 PARI/GP . . . . .	116
<b>30 Concursuri publice</b>	<b>119</b>
<b>31 Probleme de sinteză</b>	<b>127</b>
31.1 Enunțuri . . . . .	127
31.2 Răspunsuri . . . . .	135
<b>Bibliografie</b>	<b>141</b>

# Capitolul 1

## Sistemul de cifrare Cezar

### 1.1 Breviar teoretic

Algoritmul de cifrare al lui Cezar este un sistem de cifrare monoalfabetic pentru care textul clar este construit din literele alfabetului latin  $A-Z$  și cheia de cifrare este reprezentată de un număr întreg  $k \in \{0, \dots, 25\}$ .

În faza de preprocesare, delimitatorul de spațiu este ignorat sau înlocuit cu caracterul cel mai puțin frecvent din limba în care este textul clar (în limba română Q).

Fiecarei litere din textul sursă  $i$  se asociază ordinea lexicografică  $x$ . Pentru cifrare, aceasta se înlocuiește prin caracterul cod  $(x + k) \bmod 26$ . Pentru descifrare se utilizează regula inversă:  $(x - k) \bmod 26$ .

### 1.2 Exerciții rezolvate

**Exercițiul 1.2.1** *Să se cifreze mesajul:*

*CRIPTOGRAFIE*

*algoritmul utilizat fiind cifrul lui Cezar cu cheia de cifrare  $k = 7$ .*

*Rezolvare:* Se cifrează literă cu literă, ținând cont de poziția ocupată de litere în alfabet:

- Literei C îi corespunde  $x = 2$ , deci se va cifra în  $(2 + 7) \bmod 26 = 9$  adică J;

- Literei R îi corespunde  $x = 16$ , deci se va cifra în  $(16 + 7) \bmod 26 = 23$ , adică X;

Se continuă în mod analog pentru fiecare literă și în final se obține JYPWA VNYHM PL.

**Exercițiul 1.2.2** *Să se decripteze mesajul:*

*JAJSN SHWDU YTQTL DXNQJ SHJNX LTQIJ SXXXX*

*algoritmul utilizat fiind cifrul lui Cezar. Indicați cheia de cifrare.*

*Rezolvare:* Se verifică, pe rând, toate cheile posibile, până când se obține un text cu sens.

În funcție de lungimea cheii, corespondența dintre literele textului clar și cele ale textului cifrat devine:

$x$	0	1	2	3	4	5	6	...	25
<i>textul clar</i>	<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>	<i>E</i>	<i>F</i>	<i>G</i>	...	<i>Z</i>
$k = 1$	<i>B</i>	<i>C</i>	<i>D</i>	<i>E</i>	<i>F</i>	<i>G</i>	<i>H</i>	...	<i>A</i>
$k = 2$	<i>C</i>	<i>D</i>	<i>E</i>	<i>F</i>	<i>G</i>	<i>H</i>	<i>I</i>	...	<i>B</i>
$k = 3$	<i>D</i>	<i>E</i>	<i>F</i>	<i>G</i>	<i>H</i>	<i>I</i>	<i>J</i>	...	<i>C</i>
$k = 4$	<i>E</i>	<i>F</i>	<i>G</i>	<i>H</i>	<i>I</i>	<i>J</i>	<i>K</i>	...	<i>D</i>
$k = 5$	<i>F</i>	<i>G</i>	<i>H</i>	<i>I</i>	<i>J</i>	<i>K</i>	<i>L</i>	...	<i>E</i>
...	..	..	..	..	..	..	..	..	..

Se observă că sistemul presupune înlocuirea fiecărei litere cu litera corespunzătoare în alfabetul rotit cu  $k$  poziții.

Decriptând fiecare caracter în corespondentul său clar se obține, pe rând:

- pentru  $k = 1$  : IZIRM RGVCT XSPSK CWMPI RGIMW KSPHI RWWWW
- pentru  $k = 2$  : HYHQL QFUBS WRORJ BVLOH QFHLV JROGH QVVVV
- pentru  $k = 3$  : GXGPK PETAR VQNQI AUKNG PEGKU IQNFG PUUUU
- pentru  $k = 4$  : FWFOJ ODSZQ UPMPH ZTJMF ODFJT HPMEF OTTTT
- pentru  $k = 5$  : EVENI NCRYP TOLOG YSILE NCEIS GOLDE NSSSS

După o regrupare a literelor, pentru  $k = 5$  se obține: EVEN IN CRYPTOLOGY SILENCE IS GOLDEN.

## 1.3 Exerciții propuse

**Exercițiul 1.3.1** Scrieți o aplicație care să implementeze următoarele funcții:

- cifrarea unui text cu ajutorul algoritmului de cifrare Cezar;
- descifrarea unui text cifrat cu algoritmul lui Cezar;
- decriptarea unui text, despre care se știe că a fost cifrat prin metoda Cezar, prin generarea tuturor soluțiilor posibile.

Verificați rezultatul pe datele de intrare din exercițiile următoare.

**Exercițiul 1.3.2** Să se cifreze mesajul:

*MIRACLE*

algoritmul utilizat fiind cifrul lui Cezar, cheia de cifrare  $k = 3$ .

Răspuns: PLUDFOH.

**Exercițiul 1.3.3** Să se cifreze mesajul:

*CALCULATOR*

algoritmul utilizat fiind cifrul lui Cezar, cheia de cifrare  $k = 11$ .

Răspuns: NLWNF WLEZC.

**Exercițiul 1.3.4** *Să se cifreze mesajul:*

*ELECTRONIC MAIL*

*algoritmul utilizat fiind cifrul lui Cezar, cheia de cifrare  $k = 5$ .*

*Răspuns:* JQJHY WTSNH RFNQ.

**Exercițiul 1.3.5** *Să se cifreze mesajul:*

*DIGITAL SIGNATURE*

*algoritmul utilizat fiind cifrul lui Cezar, cheia de cifrare  $k = 2$ .*

*Răspuns:* FKIKV CNUKI PCVWT G.

**Exercițiul 1.3.6** *Să se decripteze mesajul:*

*IGQTI GYCUJ KPIVQ PXXXX*

*algoritmul utilizat fiind cifrul lui Cezar. Indicați cheia de cifrare.*

*Răspuns:* GEORGE WASHINGTON,  $k = 2$ .

**Exercițiul 1.3.7** *Să se decripteze mesajul:*

*UIPNB TKFGG FSTPO*

*algoritmul utilizat fiind cifrul lui Cezar. Indicați cheia de cifrare.*

*Răspuns:* THOMAS JEFFERSON,  $k = 1$ .

**Exercițiul 1.3.8** *Să se decripteze mesajul:*

*AREYY KYEOS VYUTM XGTZ*

*algoritmul utilizat fiind cifrul lui Cezar. Indicați cheia de cifrare.*

*Răspuns:* ULYSSES SIMPSON GRANT,  $k = 6$ .

**Exercițiul 1.3.9** *Să se decripteze mesajul:*

*CDTC JCON KPEQ NP*

*algoritmul utilizat fiind cifrul lui Cezar. Indicați cheia de cifrare.*

*Răspuns:* ABRAHAM LINCOLN,  $k = 2$ .

**Exercițiul 1.3.10** *Să se decripteze mesajul:*

*ECFDEPO ALCEJ*

*algoritmul utilizat fiind cifrul lui Cezar. Indicați cheia de cifrare.*

*Răspuns:* TRUSTED PARTY,  $k = 11$ .

**Exercițiul 1.3.11** *Să se cifreze mesajul:*

*EXAMEN CRIPTOGRAFIE*

*algoritmul utilizat fiind cifrul lui Cezar, cheia de cifrare  $k = 3$ .*

*Răspuns:* HADPH QFULS WRJUD ILH.

**Exercițiul 1.3.12** *Să se decripteze mesajul:*

*HADPH QFULS WRJUD ILH*

*algoritmul utilizat fiind cifrul lui Cezar. Indicați cheia de cifrare.*

*Răspuns:* EXAMEN CRIPTOGRAFIE,  $k = 3$ .

**Exercițiul 1.3.13** *Să se cifreze mesajul:*

*KANSAS CITY*

*algoritmul utilizat fiind cifrul lui Cezar, cheia de cifrare  $k = 4$ .*

*Răspuns:* OERWE WGMXC.

**Exercițiul 1.3.14** *Să se decripteze mesajul:*

*OERWE WGMXC*

*algoritmul utilizat fiind cifrul lui Cezar. Indicați cheia de cifrare.*

*Răspuns:* KANSAS CITY,  $k = 4$ .

# Capitolul 2

## Metoda substituției

### 2.1 Breviar teoretic

Operația de cifrare se bazează pe o *corespondență* biunivocă între *alfabetul clar* și *alfabetul cifrat*. Se presupune că alfabetul clar este format din cele 26 de litere (în limba română fără diacritice) plus *delimitatorul de cuvânt* spațiul. Alfabetul cifrat poate fi format din aceleași caractere sau doar din cele 26 de litere (ale limbii române) caz în care spațiul se va înlocui cu cea mai puțin frecventă literă ( $Q$ ) sau se va ignora pur și simplu. În continuare, delimitatorul de cuvânt este înlocuit cu litera  $Q$ .

Corespondența dintre cele două alfabete poate fi:

- aleatoare;
- pseudoaleatoare: plecând de la o parolă se construiește alfabetul cifrat.

Întrucât în cazul corespondenței aleatoare lucrurile sunt cât se poate de clare, vom prezenta pe scurt o metodă de construcție a corespondenței în cel de-al doilea caz. Pornind de la o parolă, alfabetul cifrat este construit după următorul algoritm:

- se scriu, o singură dată, în ordinea apariției, literele din parolă;
- se scriu literele alfabetului care nu apar în parolă.

*Corespondența* între cele două alfabete se realizează după regula alfabet în alfabet după o permutare fixă  $\sigma$  (aceasta poate fi chiar permutarea identică iar la descifrare se aplică același procedeu dar cu inversa permutării  $\sigma$ ).

În funcție de forma permutării substituția se numește:

- *directă* (alfabetul cifrat are același sens lexicografic cu alfabetul clar, sunt în total 26 astfel de substituții). Exemplu de substituție directă:

A	B	C	D	E	F	G	H	I	J	K	L	M
G	H	I	J	K	L	M	N	O	P	Q	R	S

N	O	P	Q	R	S	T	U	V	W	X	Y	Z
T	U	V	W	X	Y	Z	A	B	C	D	E	F



- *inversă* (alfabetul cifrat are sens invers lexicografic cu alfabetul clar, sunt în total 26 de astfel de substituții). Exemplu de substituție inversă:

A	B	C	D	E	F	G	H	I	J	K	L	M
U	T	S	R	Q	P	O	N	M	L	K	J	I

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

Reamintim aici trei exemple celebre (vechile coduri ebraice) de substituții reciproce (dacă litera  $\mathcal{X}$  se substituie cu litera  $\mathcal{Y}$  atunci  $\mathcal{Y}$  se va substitui cu  $\mathcal{X}$ ) și anume:

- *atbash* (prima jumătate a literelor alfabetului se mapează în cea de-a două jumătate în ordine invers lexicografică):

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

- *albam* (prima jumătate a literelor alfabetului se mapează în cea de-a două jumătate în ordine lexicografică):

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

- *atbah*:

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

În cele ce urmează vom presupune faptul că substituția este directă dacă nu este specificat altfel.

**Definiția 2.1** *Un cifru de substituție liniar de la  $\mathbf{Z}_m$  la  $\mathbf{Z}_m$  ( $m$  fiind numărul de caractere al alfabetului sursă) poate fi descris prin funcția  $f: \mathbf{Z}_m \rightarrow \mathbf{Z}_m$  definită prin  $f(x) = \alpha x + \beta$  cu  $\gcd(\alpha, m) = 1$ , funcția de descifrare fiind  $f^{-1}(x) = \alpha^{-1}(x - \beta)$ . Cheia de cifrare o formează numerele  $\alpha$  și  $\beta$ .*

**Observația 2.1** *Cifrul de substituție are proprietatea de confuzie (ascunderea legăturii dintre textul clar și textul cifrat).*

## 2.2 Exerciții rezolvate

**Exercițiul 2.2.1** *Să se construiască alfabetul de cifrare cu ajutorul parolei*

*TESTARE SISTEM*

iar apoi să se cifreze mesajul *IN CRIPTOGRAFIE NICI O REGULA NU ESTE ABSOLUTA*. Permutarea care realizează corespondența este:

0	1	2	3	4	5	6	7	8	9	10	11	12
25	24	23	22	21	20	19	18	17	16	15	14	13

13	14	15	16	17	18	19	20	21	22	23	24	25
12	11	10	9	8	7	6	5	4	3	2	1	0

*Rezolvare:*

Corepondența dintre alfabetul clar și alfabetul de cifrare (înainte de realizarea permutării) este:

A	B	C	D	E	F	G	H	I	J	K	L	M
T	E	S	A	R	I	M	B	C	D	F	G	H

N	O	P	Q	R	S	T	U	V	W	X	Y	Z
J	K	L	N	O	P	Q	U	V	W	X	Y	Z

Corepondența dintre alfabetul clar și alfabetul de cifrare după realizarea permutării este:

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

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

*Mesajul clar* se procesează astfel încât spațiul este înlocuit cu cea mai puțin frecventă literă:

*INQCRIPTOGRAFIEQNICIQOQREGULAQNUQESTEQAABSOLUTA.*

*Mesajul cifrat* va fi:

OHDXC OFMGQ CZUOV DHOXO DGDCV QIKZD HIDVB MVDZY BGKIM Z.

**Exercițiul 2.2.2** Să se descifreze mesajul:

*DOJMD OVPGF OMATN BXXXX*

algoritmul utilizat fiind o substituție simplă determinată de cuvântul cheie *PASSWORD*.

*Rezolvare:*

Corepondența dintre alfabetul clar și alfabetul de cifrare este:

A	B	C	D	E	F	G	H	I	J	K	L	M
P	A	S	W	O	R	D	B	C	E	F	G	H

N	O	P	Q	R	S	T	U	V	W	X	Y	Z
I	J	K	L	M	N	Q	T	U	V	X	Y	Z

Mesajul clar devine (dupa o regrupare a literelor) GEORGE WALKER BUSH. Se observă că de această dată nu s-a mai folosit Q pe post de delimitator de cuvânt.

## 2.3 Exerciții propuse

**Exercițiul 2.3.1** Dezvoltați o aplicație care să simuleze execuția funcțiilor de cifrare/descifrare corespunzătoare metodei substituției.

**Exercițiul 2.3.2** Dezvoltați o aplicație care să decripteze, prin metoda frecvenței, mesajele cifrate prin metoda substituției.

**Exercițiul 2.3.3** Dezvoltați o aplicație care să decripteze, prin metoda atacului cu text clar cunoscut, mesajele cifrate prin metoda substituției.

**Exercițiul 2.3.4** Să se cifreze mesajul:

WEB DESIGN

algoritmul utilizat fiind o substituție simplă determinată de cuvântul cheie BROWSER.

Răspuns: VSRWS PDAJ.

**Exercițiul 2.3.5** Să se cifreze mesajul:

PUBLIC KEY

algoritmul utilizat fiind o substituție simplă determinată de cuvântul cheie ASYMMETRIC.

Răspuns: KQSFC YDEX.

**Exercițiul 2.3.6** Să se descifreze mesajul:

ONCJB DFJPT DCJKN KKQTV TDSXXX

algoritmul utilizat fiind o substituție simplă determinată de cuvântul cheie CRIPTOGRAFIE.

Răspuns: FRANKLIN DELANO ROOSEVELT.

**Exercițiul 2.3.7** Să se descifreze mesajul:

EKBJO DSZAT NCGPF TJJTP YXXXX

algoritmul utilizat fiind o substituție simplă determinată de cuvântul cheie CRIPTO.

Răspuns: JOHN FITZGERALD KENNEDY.

**Exercițiul 2.3.8** Demonstrați că metoda de cifrare prin substituție este un sistem închis.

**Exercițiul 2.3.9** *Să se cifreze mesajul:*

*PRIVATE KEY*

*algoritmul utilizat fiind o substituție simplă determinată de cuvântul cheie BUCURESTI.*

*Răspuns: LNAVB PEF EY.*

**Exercițiul 2.3.10** *Să se descifreze mesajul:*

*LNAVB PEF EY*

*algoritmul utilizat fiind o substituție simplă determinată de cuvântul cheie BUCURESTI.*

*Răspuns: PRIVATE KEY.*

**Exercițiul 2.3.11** *Să se cifreze mesajul:*

*ASYMMETRIC ENCRYPTION*

*algoritmul utilizat fiind o substituție simplă determinată de cuvântul cheie BRASOV.*

**Exercițiul 2.3.12** *Să se descifreze mesajul:*

*BPPYI OQNEA OJANY LQEKJ*

*algoritmul utilizat fiind o substituție simplă determinată de cuvântul cheie BRASOV.*

*Răspuns: ASSYMETRIC ENCRYPTION.*



# Capitolul 3

## Sistemul de cifrare Playfair

### 3.1 Breviar teoretic

Sistemul *Playfair*, propus în anul 1854 de Charles Wheatstone dar promovat pentru utilizare de Lordul Playfair, este unul dintre cele mai cunoscute sisteme de cifrare digrafice (transformă un grup de 2 litere într-un grup de alte două litere). Acest sistem de cifrare este foarte simplu de folosit și mult mai sigur decât sistemele de substituție monoalfabetice.

Descriem în continuare modul de utilizare în cazul alfabetului latin compus din 26 litere. Literele alfabetului  $A - Z$  sunt trecute într-un careu de  $5 \times 5$  (litera  $I$  fiind asimilată literei  $J$ ). Textul clar este preprocesat astfel încât acesta să fie compatibil cu matricea de cifrare: delimitatorul de cuvânt este ignorat sau este înlocuit cu cea mai puțin frecventă literă, litera  $I$  este asimilată cu litera  $J$ , și dacă este cazul, se adaugă o literă la text pentru a avea un număr par de digrame.

Regula de cifrare este următoarea:

i) Dacă digrama care se dorește cifrată nu are literele pe aceeași linie sau coloană, atunci regula de cifrare este *regula dreptunghiului*, traseul fiind pe verticală de la cea de-a doua literă a digramei către prima literă. Sau, altfel spus, prima literă a perechii cifrate este aceea care se găsește pe aceeași linie cu prima literă a perechii în clar.

ii) Dacă digrama ce se dorește cifrată are literele pe aceeași linie, atunci se aplică regula: *cifrează la dreapta, descifrează la stânga*.

iii) Dacă digrama ce se dorește cifrată are literele pe aceeași coloană, atunci se aplică regula: *cifrează în jos, descifrează în sus*.

**Observația 3.1** Dacă o digramă apare în textul clar în ordine inversă atunci același lucru se va întâmpla și în textul cifrat.

**Observația 3.2** Algoritmul Playfair nu are regulă pentru cifrarea literelor duble: digramele ce conțin două litere identice sunt sparte prin introducerea artificială a unei alte litere.

**Observația 3.3** Algoritmul Playfair apare ca o extindere, în sensul reducerii numărului de tabele rectangulare folosite (de la două la unul), al cifrului cu 2 tabele.

Metoda cea mai frecventă de atac a acestui tip de cifru constă în analiza frecvenței digramelor de text clar combinată cu metoda comparației patternurilor din textul cifrat cu patternuri din dicționar.

## 3.2 Exerciții rezolvate

**Exercițiul 3.2.1** *Să se construiască matricea de cifrare Playfair cu ajutorul parolei*

*CRIPTOGRAFIE*

*iar apoi să se cifreze mesajul SI IN CRIPTOGRAFIE TACEREA ESTE AUR.*

*Rezolvare:* Matricea Playfair se obține trecând literele din parolă o singură dată în careul de  $5 \times 5$  iar apoi celelalte litere ale alfabetului în ordine lexicografică:

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

Mesajul este preprocesat, prin introducerea literei *Q* ca delimitator de cuvânt și la finalul mesajului (pentru ca acesta să aibă lungime pară):

*SIQINQCRIPTOGRAFIEQTACEREAQESTEQAURQ.*

Exemplificăm pentru fiecare caz câte o digramă:

- SI - conform regulii de cifrare se formează dreptunghiul cu colțurile I și S parcurs în sensul IQSP. Textul cifrat îl constituie digrama formată din colțurile care nu apar în textul clar, luate conform ordinii de parcurgere: QP.
- QI - întrucât literele sunt pe aceeași coloană se aplică regula cifrează în jos, descifrează în sus, obținându-se digrama XA ( X este litera situată sub Q și A este litera situată sub I).
- NQ - întrucât literele sunt situate pe aceeași linie se aplică regula cifrează la dreapta, descifrează la stânga, obținându-se digrama QS (Q este în dreapta lui N și S este în dreapta lui Q).

În continuare, respectând regulile de cifrare Playfair mesajul cifrat devine:  
QPXAQ SRIPT CEDGF ETAUI OIGTO FUAUP AUEQI NXXXX.

**Exercițiul 3.2.2** *Să se descifreze mesajul:*

*UFRIL ERGPC RQAW*

*Algoritmul utilizat este cifrul lui Playfair, parola utilizată fiind CRIPTOGRAFIE.*

*Rezolvare:* Matricea Playfair este aceeași din exercițiul anterior, fiind formată pornind de la aceeași parolă.

Exemplificăm pentru fiecare caz operația de descifrare pe câte o digramă:

- UF - conform regulii de descifrare, se formează dreptunghiul cu colțurile U și F. Textul clar îl constituie celelalte 2 colțuri, primul caracter al textului clar fiind cel care se găsește pe aceeași linie cu primul caracter în clar din digramă. Se obține SE.
- RI - întrucât literele sunt situate pe aceeași linie se aplică regula cifrează la dreapta, descifrează la stânga, obținându-se digrama CR (R este în stânga lui R și R este în stânga lui I).
- LE - întrucât literele sunt pe aceeași coloană se aplică regula cifrează în jos, descifrează în sus, obținându-se digrama ET (E este litera situată deasupra lui L și T este litera situată deasupra lui E).

În continuare, respectând regulile de descifrare Playfair mesajul cifrat devine:

SECRET WRITING.

### 3.3 Exerciții propuse

**Exercițiul 3.3.1** *Scrieți o aplicație care să implementeze următoarele funcții:*

- cifrarea unui text cu ajutorul algoritmului Playfair;
  - descifrarea unui text cifrat cu algoritmul Playfair;
- Verificați rezultatul pe datele de intrare din exercițiile următoare.*

**Exercițiul 3.3.2** *Să se cifreze mesajul:*

*SECURITY IS CHANGING FIELD*

*Algoritmul utilizat este cifrul lui Playfair, parola utilizată fiind CHANNEL.*

*Răspuns:* UAEQQ KYNMQ HANEL PEFLO CGMA.

**Exercițiul 3.3.3** *Să se cifreze mesajul:*

*AUTONOMOUS ATTACK AGENTS*

*Algoritmul utilizat este cifrul lui Playfair, parola utilizată fiind MALICIOUS.*

*Răspuns:* UFNDV EOESB CPZQL MFCHF PNGL.

**Exercițiul 3.3.4** *Să se cifreze mesajul:*

*VALUABLE SOURCE OF REFERENCE*

*Algoritmul utilizat este cifrul lui Playfair, parola utilizată fiind INSTITUTE.*

*Răspuns:* WERDB CFDNP DZDAM GMDMF MDTABV.



**Exercițiul 3.3.5** *Să se cifreze mesajul:*

*THE CIRCLE*

*Algoritmul utilizat este cifrul lui Playfair, parola utilizată fiind ALBUM.*

*Răspuns:* POFDKQDAKB.

**Exercițiul 3.3.6** *Să se descifreze mesajul:*

*KDDPM RUBVR PTSFU HPEBV*

*Algoritmul utilizat este cifrul lui Playfair, parola utilizată fiind PASSWORD.*

*Răspuns:* GERALD RUDOLPH FORD.

**Exercițiul 3.3.7** *Să se descifreze mesajul:*

*KDPEK DOSTF RDRXB NBBBB*

*Algoritmul utilizat este cifrul lui Playfair, parola utilizată fiind PASSWORD.*

*Răspuns:* GEORGE WALKER BUSH.

**Exercițiul 3.3.8** *Să se descifreze mesajul:*

*KDPEK DKBDC RDQOP MTKDC XPNS*

*Algoritmul utilizat este cifrul lui Playfair, parola utilizată fiind PASSWORD.*

*Răspuns:* GEORGE HERBERT WALKER BUSH.

**Exercițiul 3.3.9** *Să se descifreze mesajul:*

*GBQY YAAO RNBM*

*Algoritmul utilizat este cifrul lui Playfair, parola utilizată fiind TEST.*

*Răspuns:* HARRY TRUMAN.

**Exercițiul 3.3.10** *Să se descifreze mesajul:*

*PIGOY CLETY AEYLQ VSFWN*

*Algoritmul utilizat este cifrul lui Playfair, parola utilizată fiind CRYPTOOL.*

*Răspuns:* THE ART OF PROGRAMMING.

**Exercițiul 3.3.11** *Să se cifreze mesajul:*

*SINAIA*

*Algoritmul utilizat este cifrul lui Playfair, parola utilizată fiind SECRET KEY.*

*Răspuns:* RFOYHB.

**Exercițiul 3.3.12** *Să se descifreze mesajul:*

*RFOYHB*

*Algoritmul utilizat este cifrul lui Playfair, parola utilizată fiind SECRET KEY.*

*Răspuns:* SINAIA.

**Exercițiul 3.3.13** *Să se cifreze mesajul:*

*PREDEAL*

*Algoritmul utilizat este cifrul lui Playfair, parola utilizată fiind PASSWORD.*

*Răspuns:* RFRBD ONU.

**Exercițiul 3.3.14** *Să se descifreze mesajul:*

*RFRBD ONU*

*Algoritmul utilizat este cifrul lui Playfair, parola utilizată fiind PASSWORD.*

*Răspuns:* PREDEAL.



# Capitolul 4

## Sistemul de cifrare Hill

### 4.1 Breviar teoretic

Sistemul de cifrare Hill este o metodă de substituție poligrafică bazată pe calcule efectuate în algebra mod  $p$ .

În faza de preprocesare delimitatorul de spațiu este ignorat sau înlocuit cu caracterul cel mai puțin frecvent din limba în care este textul clar (în limba română Q).

Algoritmul procesează un bloc de date  $\mathbf{M}$  de  $n$  caractere (litere), cheia de cifrare fiind reprezentată de o matrice  $\mathbf{K}$  de dimensiune  $n \times n$ , inversabilă mod  $p$ .

Există două subclase ale algoritmului Hill pentru care regulile de cifrare diferă prin ordinea în care se efectuează înmulțirile: o prima subclasa are ca regulă de cifrare operația de înmulțire  $\mathbf{C} = \mathbf{MK}$  cu descifrarea  $\mathbf{M} = \mathbf{CK}^{-1}$  iar a doua subclasa folosește ca regulă de cifrare înmulțirea  $\mathbf{C} = \mathbf{KM}$  având descifrarea corespunzătoare  $\mathbf{M} = \mathbf{K}^{-1}\mathbf{C}$ .

**Observația 4.1** Dacă matricea  $\mathbf{K}$  este simetrică (matricea  $\mathbf{K}$  și transpusa ei sunt egale) atunci regulile de cifrare pentru cele două subclase sunt echivalente.

**Observația 4.2** În cazul alfabetului latin  $p = 26$ , cheia de cifrare  $\mathbf{K}$  trebuie să fie o matrice inversabilă mod 26.

### 4.2 Exerciții rezolvate

**Exercițiul 4.2.1** Să se cifreze mesajul:

*BLAZE OF GLORY.*

Algoritmul utilizat este cifrul lui Hill ( $2 \times 2$ ), cheia de cifrare fiind matricea:

$$\begin{pmatrix} J & B \\ V & I \end{pmatrix}.$$

*Rezolvare:* Prin înlocuirea literelor din cheie cu pozițiile corespunzătoare din alfabet (A - 0, B - 1, etc.) se obține:

$$\mathbf{K} = \begin{pmatrix} 9 & 1 \\ 21 & 8 \end{pmatrix}.$$

Textul clar se sparge în blocuri de 2 caractere, care se cifrează pe rând. De exemplu, BL corespunde matricii

$$\mathbf{M} = \begin{pmatrix} 1 & 11 \end{pmatrix}.$$

Digrama se cifrează în:

$$\mathbf{C} = \begin{pmatrix} 1 & 11 \end{pmatrix} \begin{pmatrix} 9 & 1 \\ 21 & 8 \end{pmatrix} \bmod 26 = \begin{pmatrix} 6 & 11 \end{pmatrix} = \begin{pmatrix} G & L \end{pmatrix}.$$

Deci, BL se cifrează în GL. Se continuă în mod analog. În final se obține: GLFSS MPBDT HB.

**Exercițiul 4.2.2** *Să se descifreze mesajul:*

*JESHB JJAZM TANCF VBJXX.*

*Algoritmul utilizat este cifrul lui Hill ( $2 \times 2$ ), cheia de cifrare fiind matricea:*

$$\begin{pmatrix} H & U \\ D & F \end{pmatrix}.$$

*Rezolvare:* Prin înlocuirea literelor din cheie cu pozițiile corespunzătoare din alfabet (A - 0, B - 1, etc.) se obține:

$$\mathbf{K} = \begin{pmatrix} 7 & 20 \\ 3 & 5 \end{pmatrix}.$$

Se determină inversa matricii  $K \bmod 26$  :

$$\mathbf{K}^{-1} = \det(\mathbf{K})^{-1} \mathbf{K}^* \bmod 26, unde$$

$$\det(\mathbf{K})^{-1} \bmod 26 = (7 \cdot 5 - 3 \cdot 20)^{-1} \bmod 26 = (-25)^{-1} \bmod 26 = 1$$

și

$$\mathbf{K}^* = \begin{pmatrix} 5 & -20 \\ -3 & 7 \end{pmatrix} \bmod 26 = \begin{pmatrix} 5 & 6 \\ 23 & 7 \end{pmatrix}.$$

S-a obținut:

$$\mathbf{K}^{-1} = \begin{pmatrix} 5 & 6 \\ 23 & 7 \end{pmatrix}.$$

Pentru descifrarea perechii JE, se determină matricea linie care conține valorile corespunzătoare din alfabet:

$$\mathbf{C} = \begin{pmatrix} J & E \end{pmatrix} = \begin{pmatrix} 9 & 4 \end{pmatrix}.$$

Prin înmulțire cu cheia de descifrare se obține:

$$\mathbf{M} = \begin{pmatrix} 9 & 4 \end{pmatrix} \begin{pmatrix} 5 & 6 \\ 23 & 7 \end{pmatrix} \bmod 26 = \begin{pmatrix} 7 & 4 \end{pmatrix} = \begin{pmatrix} H & E \end{pmatrix}.$$

Deci, JE se descifrează în HE.

Se procedează în mod analog pentru toate perechile de câte 2 caractere cifrate: SH se descifrează în RB, BJ în ER, etc.

În final, după efectuarea tuturor calculelor și regruparea literelor, se obține: HERBERT CLARK HOOVER.

### 4.3 Exerciții propuse

**Exercițiul 4.3.1** *Scrieți o aplicație care să implementeze funcțiile de cifrare și descifrare, specifice algoritmului Hill cu  $p = 26$ .*

*Verificați rezultatul pe datele de intrare din exercițiile următoare.*

**Exercițiul 4.3.2** *Să se cifreze mesajul:*

*COMPLETE AND PROPER PACKAGE.*

*Algoritmul utilizat este cifrul lui Hill, cheia de cifrare fiind matricea:*

$$\begin{pmatrix} N & T \\ C & R \end{pmatrix}.$$

*Răspuns:* GIZTL MLCNN MBTML UMDMI AUYS.

**Exercițiul 4.3.3** *Să se cifreze mesajul:*

*ESOTERIC TOPIC OF RESEARCH.*

*Algoritmul utilizat este cifrul lui Hill, cheia de cifrare fiind matricea:*

$$\begin{pmatrix} B & Y \\ G & P \end{pmatrix}.$$

*Răspuns:* ICYXC NUOZQ LMIYD LICES DWHM.

**Exercițiul 4.3.4** *Să se cifreze mesajul:*

*BENJAMIN HARRISON.*

*Algoritmul utilizat este cifrul lui Hill ( $3 \times 3$ ), cheia de cifrare fiind matricea:*

$$\begin{pmatrix} A & B & C \\ B & C & A \\ C & A & B \end{pmatrix}.$$

*Răpuns:* EJPYJ EBIXZ IRUSE ANA.

**Exercițiul 4.3.5** *Să se descifreze mesajul:*

*ZKNAW NIOZO BRXSW QNNXX.*

*Algoritmul utilizat este cifrul lui Hill ( $2 \times 2$ ), cheia de cifrare fiind matricea:*

$$\begin{pmatrix} B & E \\ V & H \end{pmatrix}.$$

*Răpuns:* RONALD WILSON REAGAN.

**Exercițiul 4.3.6** *Să se descifreze mesajul:*

*ZPXUB IRHNU VXWSP DJTNN.*

*Algoritmul utilizat este cifrul lui Hill ( $2 \times 2$ ), cheia de cifrare fiind matricea:*

$$\begin{pmatrix} J & D \\ X & C \end{pmatrix}.$$

*Răpuns:* RICHARD MILHOUS NIXON.

**Exercițiul 4.3.7** *Să se descifreze mesajul:*

*EJPYJ EBIXZ IRUSE ANA.*

*Algoritmul utilizat la cifrare este cifrul lui Hill ( $3 \times 3$ ), cheia de cifrare fiind matricea:*

$$\begin{pmatrix} A & B & C \\ B & C & A \\ C & A & B \end{pmatrix}.$$

*Răpuns:* BENJAMIN HARRISON.

**Exercițiul 4.3.8** *Să se descifreze mesajul:*

*NYNAF JUWBL ZXANM NGLEI JQWF*

*Algoritmul utilizat este cifrul lui Hill, cheia de cifrare fiind matricea:*

$$\begin{pmatrix} J & S \\ W & V \end{pmatrix}.$$

*Răspuns:* FINAL ROUND TRANSFORMATION.

**Exercițiul 4.3.9** *Să se descifreze mesajul:*

*NKTNM QZQEY WVDIA CIGMG.*

*Algoritmul utilizat este cifrul lui Hill, cheia de cifrare fiind matricea:*

$$\begin{pmatrix} D & I \\ K & B \end{pmatrix}.$$

*Răspuns:* RETRIEVE YOUR BAGGAGE.

**Exercițiul 4.3.10** *Demonstrați că algoritmul lui Hill este un algoritm de cifrare închis.*

**Exercițiul 4.3.11** *Să se cifreze mesajul:*

*OPERATIONAL RESEARCH.*

*Algoritmul utilizat este cifrul lui Hill, cheia de cifrare fiind matricea:*

$$\begin{pmatrix} F & H \\ H & I \end{pmatrix}.$$

*Răspuns:* TKJID WIMNN SFQQU CVFLD.

**Exercițiul 4.3.12** *Să se descifreze mesajul:*

*TKJID WIMNN SFQQU CVFLD.*

*Algoritmul utilizat este cifrul lui Hill, cheia de cifrare fiind matricea:*

$$\begin{pmatrix} F & H \\ H & I \end{pmatrix}.$$

*Răspuns:* OPERATIONAL RESEARCH.

**Exercițiul 4.3.13** *Să se cifreze mesajul:*

*CRYPTOLOGY.*

*Algoritmul utilizat este cifrul lui Hill, cheia de cifrare fiind matricea:*

$$\begin{pmatrix} T & E \\ S & T \end{pmatrix}.$$

*Răspuns:* CVWPB KFWCS.

**Exercițiul 4.3.14** *Să se cifreze mesajul:*

*NAVAJO CODE.*

*Algoritmul utilizat este cifrul lui Hill, cheia de cifrare fiind matricea:*

$$\begin{pmatrix} L & Q \\ L & J \end{pmatrix}.$$

*Răspuns:* NNXXL RMSTR.



**Exercițiul 4.3.15** *Să se descifreze mesajul:*

*CVWPB KFWCS.*

*Algoritmul utilizat este cifrul lui Hill, cheia de cifrare fiind matricea:*

$$\begin{pmatrix} T & E \\ S & T \end{pmatrix}.$$

*Răspuns:* CRYPTOLOGY.

**Exercițiul 4.3.16** *Să se descifreze mesajul:*

*NNXXL RMSTR.*

*Algoritmul utilizat este cifrul lui Hill, cheia de cifrare fiind matricea:*

$$\begin{pmatrix} L & Q \\ L & J \end{pmatrix}.$$

*Răspuns:* NAVAJO CODE.

# Capitolul 5

## Sisteme de cifrare polialfabetice

### 5.1 Breviar teoretic

Un sistem de cifrare de tip substituție polialfabetică este generalizarea sistemului de cifrare de substituție monoalfabetică, fiind compus dintr-un număr  $N$  de alfabete. Fiecare alfabet reprezintă o permutare (stabilită în funcție de parolă) a alfabetului de intrare. Algoritmul de cifrare constă în substituirea celei de a  $i$ -a litere  $m$  din textul clar cu litera corespunzătoare din cel de al  $i \bmod N$  alfabet.

Sistemele polialfabetice sunt ușor de identificat prin aplicarea analizei frecvențelor de apariție a literelor în secvențe decimate din textul cifrat.

Un exemplu de sistem polialfabetic este algoritmul lui Vigenère în care parola  $k_1, \dots, k_n$  este folosită periodic pentru a transforma caracterul  $m_j \in \{A, \dots, Z\}$  din textul clar după formula:  $c_j = (m_j + k_{j \bmod n}) \bmod 26$ . Pentru descifrare se folosește formula:  $m_j = (c_j - k_{j \bmod n}) \bmod 26$ .

Atacul sistemelor polialfabetice este similar cu atacul a  $N$  sisteme de substituție monoalfabetică. Deci, o procedură de *tip divide et impera* are o complexitate de  $O(N)$ . Procedura este descrisă în continuare:

**Intrare:** Textul cifrat de lungime  $M$  suficient de mare.

**Ieșire:** Textul clar corespunzător sistemului de cifrare polialfabetic.

**PASUL 1.** Determină numărul de alfabete  $N$ .

**PASUL 2.** Pentru  $j = 0$  to 4 execută:

pentru  $i = 1$  to  $N - j$  execută:

aplică procedura de reconstrucție parțială (pe baza frecvențelor  $(j + 1)$ -gramelor) a alfabetelor  $i, \dots, i + j$ .

**PASUL 3.** Conform celor  $N$  alfabete reconstruiește textul clar.

**Observația 5.1** *Procedura descrisă mai sus are ca parametru implicit de analiză numărul maxim de legături 4 : astfel, 1-gramele sunt caracterele, 2-gramele sunt dubletii, etc.*

## 5.2 Exerciții rezolvate

**Exercițiul 5.2.1** *Să se cifreze mesajul WINDS OF CHANGE cu ajutorul algoritmului Vigenère, parola fiind FUTURE.*

*Rezolvare:* Aplicând cifrarea pentru fiecare caracter al textului clar, ținând cont de poziția acestora în alfabet, se obține:

$j$	$m_j$	$k_{j(\bmod 6)}$	$c_j = (m_j + k_{j(\bmod 6)})(\bmod 26)$
1	$W - 22$	$F - 5$	$(22 + 5)(\bmod 26) = 1 - B$
2	$I - 8$	$U - 20$	$(8 + 20)(\bmod 26) = 2 - C$
3	$N - 13$	$T - 19$	$(13 + 19)(\bmod 26) = 6 - G$
4	$D - 3$	$U - 20$	$(3 + 20)(\bmod 26) = 23 - X$
5	$S - 18$	$R - 17$	$(18 + 17)(\bmod 26) = 9 - J$
6	$O - 14$	$E - 4$	$(14 + 4)(\bmod 26) = 18 - S$
7	$F - 5$	$F - 5$	$(5 + 5)(\bmod 26) = 10 - K$
8	$C - 2$	$U - 20$	$(2 + 20)(\bmod 26) = 22 - W$
9	$H - 7$	$T - 19$	$(7 + 19)(\bmod 26) = 0 - A$
10	$A - 0$	$U - 20$	$(0 + 20)(\bmod 26) = 20 - U$
11	$N - 13$	$R - 17$	$(13 + 17)(\bmod 26) = 4 - E$
12	$G - 6$	$E - 4$	$(6 + 4)(\bmod 26) = 10 - K$
13	$E - 4$	$F - 5$	$(4 + 5)(\bmod 26) = 9 - J$

Rezultă textul cifrat: BCGXJ SKWAU EKJ.

**Exercițiul 5.2.2** *Să se descifreze mesajul IHWGZ CIHGO GKAJV OI știind că a fost cifrat cu ajutorul algoritmului Vigenère, parola fiind PASSWORD.*

*Rezolvare:* Aplicând descifrarea pentru fiecare caracter al textului cifrat, ținând cont de poziția acestora în alfabet, se obține:

$j$	$c_j$	$k_{j(\bmod 8)}$	$m_j = (c_j - k_{j(\bmod 8)})(\bmod 26)$
1	$I - 8$	$P - 15$	$(8 - 15)(\bmod 26) = 19 - T$
2	$H - 7$	$A - 0$	$(7 - 0)(\bmod 26) = 7 - H$
3	$W - 22$	$S - 18$	$(22 - 18)(\bmod 26) = 4 - E$
4	$G - 6$	$S - 18$	$(6 - 18)(\bmod 26) = 14 - O$
5	$Z - 25$	$W - 22$	$(25 - 22)(\bmod 26) = 3 - D$
6	$C - 2$	$0 - 14$	$(2 - 14)(\bmod 26) = 14 - O$
7	$I - 8$	$R - 17$	$(8 - 17)(\bmod 26) = 17 - R$
8	$H - 7$	$D - 3$	$(7 - 3)(\bmod 26) = 4 - E$
9	$G - 6$	$P - 15$	$(6 - 15)(\bmod 26) = 17 - R$
10	$O - 14$	$A - 0$	$(14 - 0)(\bmod 26) = 14 - O$
11	$G - 6$	$S - 18$	$(6 - 18)(\bmod 26) = 14 - O$
12	$K - 10$	$S - 18$	$(10 - 18)(\bmod 26) = 18 - S$
13	$A - 0$	$W - 22$	$(0 - 22)(\bmod 26) = 4 - E$
14	$J - 9$	$0 - 14$	$(9 - 14)(\bmod 26) = 21 - V$
15	$V - 21$	$R - 17$	$(21 - 17)(\bmod 26) = 4 - E$
16	$O - 14$	$D - 3$	$(14 - 3)(\bmod 26) = 11 - L$
17	$I - 8$	$P - 15$	$(8 - 15)(\bmod 26) = 19 - T$

Dupa gruparea literelor rezultă: THEODORE ROOSEVELT.

### 5.3 Exerciții propuse

**Exercițiul 5.3.1** Să se cifreze mesajul *OPTIMISTIC* cu ajutorul algoritmului Vigenère, folosind parola *GOODDAYS*.

*Răspuns:* UDHLPIQLOQ.

**Exercițiul 5.3.2** Să se cifreze mesajul *THANK YOU* cu ajutorul algoritmului Vigenère, folosind parola *POLITE*.

*Răspuns:* IVLVD CDI.

**Exercițiul 5.3.3** Să se cifreze mesajul *GOING BACK IN TIME* cu ajutorul algoritmului Vigenère, folosind parola *MEMORY*.

*Răspuns:* SSUBX ZMGW WE RUQQ.

**Exercițiul 5.3.4** Să se cifreze mesajul *FAST CARS* cu ajutorul algoritmului Vigenère, folosind parola *RADAR*.

*Răspuns:* WAVT TRRV.

**Exercițiul 5.3.5** Să se cifreze mesajul *SUITCASE* cu ajutorul algoritmului Vigenère, folosind parola *TRIP*.

*Răspuns:* LLQIVRAT.

**Exercițiul 5.3.6** Să se descifreze mesajul *WIUXGHG WXGALFYK* știind că a fost cifrat cu ajutorul algoritmului Vigenère, parola fiind *TEST*.

*Răspuns:* DECENDO DECISMUS.

**Exercițiul 5.3.7** Să se descifreze mesajul *UAEGQD OOGAT* știind că a fost cifrat cu ajutorul algoritmului Vigenère, parola fiind *TANGO*.

*Răspuns:* BARACK OBAMA.

**Exercițiul 5.3.8** Să se descifreze mesajul *XVLGM OXLDC* știind că a fost cifrat cu ajutorul algoritmului Vigenère, parola fiind *BRIDE*.

*Răspuns:* WEDDING DAY.

**Exercițiul 5.3.9** Să se descifreze mesajul *IHZSV SKIEE CHWPU ACSH* știind că a fost cifrat cu ajutorul algoritmului Vigenère, parola fiind *PARADOX*.

*Răspuns:* THIS SENTENCE IS FALSE.

**Exercițiul 5.3.10** Să se descifreze mesajul *MYEYS VOJFQ ZAVLL N* știind că a fost cifrat cu ajutorul algoritmului Vigenère, parola fiind *TRANSILVANIA*.

*Răspuns:* THE LAND OF DRACULA.

**Exercițiul 5.3.11** Să se cifreze mesajul *OPERATIONAL RESEARCH* cu ajutorul algoritmului Vigenère, folosind parola *PASSWORD*.

*Răspuns:* DPWJW HZRCA DJAGV DGCZ.

**Exercițiul 5.3.12** Să se descifreze mesajul *DPWJW HZRCA DJAGV DGCZ* știind că a fost cifrat cu ajutorul algoritmului Vigenère, parola fiind *PASSWORD*.

*Răspuns:* OPERATIONAL RESEARCH.

**Exercițiul 5.3.13** Să se cifreze mesajul *CRIPTOGRAFIE* cu ajutorul algoritmului Vigenère, folosind parola *TEST*.

*Răspuns:* VVAIM SYKTJ AX.

**Exercițiul 5.3.14** Să se descifreze mesajul *VVAI MSYK TJAX* știind că a fost cifrat cu ajutorul algoritmului Vigenère, parola fiind *TEST*.

*Răspuns:* CRIPTOGRAFIE.

# Capitolul 6

## Metoda transpoziției

### 6.1 Breviar teoretic

Metoda transpoziției asigură, în cadrul sistemelor criptografice, realizarea difuziei: împrăștierea proprietăților statistice ale textului clar în textul cifrat. Metoda transpoziției îmbracă mai multe forme: textul *este citit* într-o formă matriceală linie cu linie sau coloană cu coloană, *se permută* liniile și/sau coloanele, rezultatul fiind apoi *scris* linie cu linie sau coloană cu coloană. Spre exemplu, în cazul transpoziției coloanelor, textul clar se citește, linie cu linie, într-o formă tabelară cu  $n$  coloane, acesta fiind scris pe coloane în funcție de cheia de cifrare reprezentată de o permutare din  $\sigma_n$ .

Dacă dimensiunea textului clar nu este un multiplu de  $n$  atunci acesta se poate completa sau nu cu un caracter bine precizat. În faza de preprocesare delimitatorul de spațiu este ignorat sau înlocuit cu caracterul cel mai puțin frecvent din limba în care este textul clar (în limba română Q).

### 6.2 Exerciții rezolvate

**Exercițiul 6.2.1** Să se cifreze prin metoda transpoziției ( $N = 12$ ), pornind de la parola

*CRIPTOGRAFIE*

*mesajul SI IN CRIPTOGRAFIE TACEREA ESTE AUR.*

*Rezolvare:* Vom construi secvența numerică de cifrare asociind fiecărei litere din parolă indicele din ordinea lexicografică: astfel literele din parolă, scrise în ordine lexicografică sunt:

1	2	3	4	5	6	7	8	9	10	11	12
A	C	E	F	G	I	I	O	P	R	R	T

deci parola *CRIPTOGRAFIE* produce permutarea: 2 10 6 9 12 8 5 11 1 4 7 3.

Textul clar este scris într-o tabelă cu 12 coloane:

2	10	6	9	12	8	5	11	1	4	7	3
S	I	Q	I	N	Q	C	R	I	P	T	O
G	R	A	F	I	E	Q	T	A	C	E	R
E	A	Q	E	S	T	E	Q	A	U	R	Q

Deoarece lungimea textului nu este divizibilă cu 12 vom completa ultimul rând cu o secvență cunoscută (în acest caz caracterul  $Q$ ). Textul cifrat se obține citind coloanele tabelului de cifrare în ordinea indicată de parola numerică: IAASG EORRQ PCUCQ EQAQT ERQET IFEIR ARTQN IS.

Descifrarea se va realiza în mod similar folosind permutarea inversă  $\sigma^{-1}$ .

Dacă dimensiunea transpoziției  $N$  este mai mică decât lungimea parolei atunci se vor reține  $N$  caractere din parolă.

## 6.3 Exerciții propuse

**Exercițiul 6.3.1** Scrieți un program care să implementeze funcțiile de cifrare/descifrare specifice metodei transpoziției coloanelor.

**Exercițiul 6.3.2** Să se cifreze mesajul:

*ELECTRIC HOTPLATE*

printr-o transformare de tip transpoziție cu ajutorul permutării  $\sigma = (2, 1, 3)$ .

Răspuns: LTCOL EECIH PTERQ TAQ.

**Exercițiul 6.3.3** Să se cifreze mesajul:

*CERCETARI OPERATIONALE*

printr-o transformare de tip transpoziție cu ajutorul permutării  $\sigma = (3, 1, 2)$ .

Răspuns: EEROR IAQRT IPAOL QCCAQ ETNE.

**Exercițiul 6.3.4** Să se cifreze mesajul *CRIPTOGRAFIE* prin metoda transpoziției utilizând permutarea  $\sigma = (4, 2, 1, 3)$ . Verificați rezultatul obținut.

**Exercițiul 6.3.5** Să se descifreze mesajul:

*EORSE TOROE LHDEO VT*

cifrat printr-o transformare de tip transpoziție cu ajutorul permutării  $\sigma = (2, 3, 1)$ .

Răspuns: THEODORE ROOSEVELT.

**Exercițiul 6.3.6** Să se descifreze mesajul:

*SFCME TAEAE NLR*

cifrat printr-o transformare de tip transpoziție cu ajutorul permutării  $\sigma = (1, 2, 3)$ .

*Răspuns:* STEFAN CEL MARE.

**Exercițiul 6.3.7** *Să se descifreze mesajul:*

*HTZMA VEUI IAL*

*cifrat printr-o transformare de tip transpoziție cu ajutorul permutării  $\sigma = (2, 3, 1)$ .*

*Răspuns:* MIHAI VITEAZUL.

**Exercițiul 6.3.8** *Să se descifreze mesajul:*

*NMTMA STEDI NEINO NT*

*cifrat printr-o transformare de tip transpoziție cu ajutorul permutării  $\sigma = (2, 3, 1)$ .*

*Răspuns:* SENTIMENT DOMINANT.

**Exercițiul 6.3.9** *Să se descifreze mesajul:*

*TDDDR TEAAU EIASN RLCPR*

*cifrat printr-o transformare de tip transpoziție cu ajutorul permutării  $\sigma = (3, 1, 2)$ .*

*Răspuns:* STANDARDUL DE CRIPTARE.

**Exercițiul 6.3.10** *Demonstrați că algoritmul de cifrare ce utilizează transpoziția este un sistem închis.*

**Exercițiul 6.3.11** *Să se cifreze mesajul:*

*CERCETARI OPERATIONALE*

*printr-o transformare de tip transpoziție cu ajutorul permutării  $\sigma = (2, 1, 3)$ .*

*Răspuns:* EERPAOLCC AORIARTIETNE.

**Exercițiul 6.3.12** *Să se descifreze mesajul:*

*EERPAOLCC AORIARTIETNE*

*cifrat printr-o transformare de tip transpoziție cu ajutorul permutării  $\sigma = (2, 1, 3)$ .*

*Răspuns:* CERCETARI OPERATIONALE.

**Exercițiul 6.3.13** *Să se cifreze mesajul:*

*OPERATIONAL RESEARCH*

*printr-o transformare de tip transpoziție cu ajutorul permutării  $\sigma = (2, 1, 4, 3)$ .*

*Răspuns:* PTASC OANER RORAE ILEH.

**Exercițiul 6.3.14** *Să se descifreze mesajul:*

*PTASC OANER RORAE ILEH*

*cifrat printr-o transformare de tip transpoziție cu ajutorul permutării  $\sigma = (2, 1, 4, 3)$ .*

*Răspuns:* OPERATIONAL RESEARCH.





# Capitolul 7

## Sisteme mixte

### 7.1 Breviar teoretic

Sistemele mixte au la bază o cifrare succesivă a mesajului prin metoda substituției și apoi prin metoda transpoziției sau invers.

Atacarea sistemul de cifrare se realizează de la ultima sa componentă către prima. Remarcăm faptul că substituția simplă este comutativă cu operația de transpoziție deci se poate oricând aborda mai întâi substituția și apoi transpoziția. În cazul utilizării unui sistem polialfabetic, cu număr necunoscut de alfabete, recomandarea este ca după stabilirea, prin metode statistice, a numărului de alfabete, să se abordeze concomitent identificarea efectivă a alfabetelor și al transpoziției utilizate. În cazul utilizării unui sistem poligrafic (tabele de cifrare) și o transpoziție este recomandabilă o tehnică de tip backtracking.

### 7.2 Exerciții rezolvate

**Exercițiul 7.2.1** *Să se cifreze mesajul GEOMETRIC FIGURE cu ajutorul algoritmului lui Cezar ( $k = 5$ ) și al transpoziției  $\sigma = (2, 1, 3)$ .*

*Rezolvare:* Mai întâi textul este cifrat cu sistemul Cezar folosind cheia  $k = 5$ , deci corespondența dintre cele 2 alfabete devine:

text clar	A	B	C	D	E	F	G	H	I	...
text cifrat	F	G	H	I	J	K	L	M	N	...

Astfel se obține: LJT RJY WNH KNL ZWJ. Apoi, textul obținut se așează într-o tabelă cu 3 coloane:

2	1	3
L	J	T
R	J	Y
W	N	H
K	N	L
Z	W	J

Textul cifrat se determină citind pe coloane în ordinea indicată de permutare (coloana din mijloc, apoi cea din stânga și în final cea din dreapta): JJNNWLRW KZTYHLJ .

**Exercițiul 7.2.2** Să se decripteze mesajul următor:

*DKVUR UTUBK WFCVG ETGOC XWVWC*

*OCVPQ VUVWG FGHTQ VKUUV KKNKC*

*RKCPQ OQFKC EWVG*

știind că a fost cifrat cu ajutorul algoritmului lui Cezar ( $k = 2$ ) și supracifrat prin metoda transpoziției utilizând permutarea  $(3, 2, 1)$ .

*Rezolvare:* Cum substituția și transpoziția sunt comutative, putem mai întâi decripta mesajul folosind Cezar cu cheia  $k = 2$  și apoi decripta prin metoda transpoziției.

Pentru decriptarea mesajului folosind metoda Cezar cu  $k = 2$ , fiecare caracter se înlocuiește cu caracterul situat cu 2 poziții mai înainte în alfabet:

text cifrat	A	B	C	D	E	F	G	H	I	...
text clar	Y	Z	A	B	C	D	E	F	G	...

După decriptare, textul devine: BITSP SRSZI UDATE CREMA VUTUA MATNO TSTUE DEFRO TISST IILIA PIANO MODIA CUTE .

Acesta reprezintă un text cifrat prin metoda transpoziției. Cum textul are 64 de caractere și permutarea este de lungime 3, atunci numărul de litere pe coloane este: 21, 21 și 22. Coloanele cu numai 21 de caractere sunt cele care corespund valorilor luate în ordine descrescătoare din permutarea inversă  $\sigma^{-1} = (3, 2, 1)$ :

3	2	1	1	2	3
B	U	S	S	U	B
I	T	S	S	T	I
T	U	T	T	U	T
S	A	I	I	A	S
P	M	I	I	M	P
S	A	L	L	A	S
R	T	I	I	T	R
S	N	A	A	N	S
Z	O	P	P	O	Z
I	T	I	I	T	I
U	S	A	A	S	U
D	T	N	N	T	D
A	U	O	O	U	A
T	E	M	M	E	T
E	D	O	O	D	E
C	E	D	D	E	C
R	F	I	I	F	R
E	R	A	A	R	E
M	O	C	C	O	M
A	T	U	U	T	A
V	I	T	T	I	V
		E			E

După rearanjarea coloanelor conform permutării inverse  $\sigma^{-1}$  se obține tabela din dreapta. Citind pe linii se descoperă textul clar: SUBSTITUTIA SIMPLA SI TRANSPOZITIA SUNT DOUA METODE DE CIFRARE COMUTATIVE .

## 7.3 Exerciții propuse

**Exercițiul 7.3.1** Dezvoltați o aplicație care să implementeze rutine specifice decriptării sistemelor mixte compuse din transpoziții și substituții simple.

**Exercițiul 7.3.2** Se dau criptogramele:

*Criptograma 1:*

VXEVW LWXWL DVLPS ODVLW UDQVS  
RCLWL DVXQW GRXDP HWRGH GHFLI  
UDUHF RPXWD WLYHX

*Criptograma 2:*

YAHYZ OZAZO GYOSV RGYOZ XGTYV  
UFOZO GYATZ JUAGS KZUJK JKIOL  
XGXKI USAZG ZOBKX

Care din afirmațiile de mai jos sunt adevărate:

- a) metoda de cifrare utilizată este o substituția simplă;
- b) metoda de cifrare utilizată este o transpoziție;
- c) metoda de cifrare este reprezentată de algoritmul lui Cezar;
- d) nu se poate preciza sistemul criptografic utilizat.

Justificați răspunsul. Deciptați mesajul.

Răspuns: a) și c). Textul clar: SUBSTITUTIA SIMPLA SI TRANSPOZITIA SUNT DOUA METODE DE CIFRARE COMUTATIVE.

**Exercițiul 7.3.3** Se dau criptogramele:

Criptograma 1:

BITSP SRSZI UDATE CREMA VUTUA  
MATNO TSTUE DEFRO TISST IILIA  
PIANO MODIA CUTE

Criptograma 2:

UTUAM ATNOT STUED EFROT IBITS  
PSRSZ IUDAT ECREM AVSST IILIA  
PIANO MODIA CUTE

Care din afirmațiile de mai jos sunt adevărate:

- a) metoda de cifrare utilizată este o substituția simplă;
- b) metoda de cifrare utilizată este o transpoziție;
- c) metoda de cifrare este reprezentată de algoritmul lui Cezar;
- d) nu se poate preciza sistemul criptografic utilizat.

Justificați răspunsul. Deciptați mesajul.

Răspuns: b). Textul clar: SUBSTITUTIA SIMPLA SI TRANSPOZITIA SUNT DOUA METODE DE CIFRARE COMUTATIVE.

**Exercițiul 7.3.4** Cifrați mesajul *SPECIAL PROPERTY* folosind algoritmului lui Cezar ( $k = 13$ ) și transpoziția dată de  $\sigma = (2, 4, 3, 1)$ .

Răspuns: PCRFVEE RYCLCNBG.

**Exercițiul 7.3.5** Deciptați mesajul *CPKQCG ZGTVTKGOERIH* știind că a fost cifrat cu ajutorul algoritmului lui Cezar și al unei transpoziții.

Răspuns: EXAMEN CRIPTOGRAFIE.

**Exercițiul 7.3.6** Deciptați mesajul *ZGTVTK GOERIHCPKQCG* știind că a fost cifrat cu ajutorul algoritmului lui Cezar și al unei transpoziții.

Răspuns: EXAMEN CRIPTOGRAFIE.

# Capitolul 8

## Generatoare pseudoaleatoare

### 8.1 Breviar teoretic

Un registru de deplasare cu feedback constă în  $n$  locații de memorie de câte un bit care se "deplasează" spre dreapta și o funcție de feedback care exprimă orice element nou  $a(t)$ , cu  $t \geq n$ , al secvenței în funcție de elementele generate anterior  $a(t-n), a(t-n+1), \dots, a(t-1)$ .

Funcția de feedback trebuie să fie nesaringulară, adică de forma:

$a(t) = g(a(t-1), \dots, a(t-n+1)) \oplus a(t-n)$ , unde  $\oplus$  desemnează operația SAU exclusiv (XOR). Dacă funcția de feedback este liniară (se poate implementa doar folosind operația SAU exclusiv) spunem că generatorul este un registru de deplasare cu feedback liniar (**LFSR**). Altfel, spunem că generatorul este un registru de deplasare cu feedback neliniar (**NLFSR**).

O locație de memorie a registrului se numește nivel, iar semnalele binare  $a(0), a(1), \dots, a(n-1)$  sunt încărcate ca date inițiale. Perioada secvenței produse depinde atât de numărul de niveluri, cât și de detaliile conexiunilor de feedback. Mai exact, perioada maximă a secvenței care poate fi generată de un registru de deplasare cu feedback, având  $n$  niveluri și o funcție de feedback nesaringulară este  $2^n - 1$ , adică numărul maxim de stări în care se poate afla un registru cu  $n$  niveluri (se exclude starea nulă). **LFSR**-urile sunt folosite de mult timp pentru teste **VSLI**, comunicații cu spectru distribuit etc. Funcția de feedback a unui **LFSR** are forma:

$$a(t) = c_1 a(t-1) \oplus c_2 a(t-2) \oplus \dots \oplus c_{n-1} a(t-n+1) \oplus a(t-n), \quad (8.1)$$

unde  $c_i \in \{0, 1\}$ . Conexiunea de feedback a unui **LFSR** poate fi exprimată printr-un polinom de feedback:

$$f(X) = 1 + c_1 X + c_2 X^2 + \dots + c_{n-1} X^{n-1} + X^n,$$

cu nedeterminata  $X$ . Acest polinom decide perioada și comportarea statistică a secvenței de ieșire. Pentru a preveni o secvență de ieșire trivială, trebuie ca starea „zero peste tot” să nu

fie stare inițială. De exemplu, dacă un **LFSR** cu patru niveluri are polinomul de feedback:

$$f(X) = 1 + X + X^2 + X^3 + X^4,$$

dependent de starea inițială, atunci el va genera una din secvențele de perioadă 5.

- a) 1111011110...
- b) 1000110001...
- c) 0100101001...

Sau, alt exemplu, dacă **LFSR** are polinomul de feedback dat de  $f(X) = 1 + X + X^4$ , atunci el generează o singură secvență netrivială de perioadă 15, cu cea mai bună statistică pe care o astfel de secvență o poate avea:

101100100011110...

Pentru a garanta cea mai mare perioadă posibilă  $2^n - 1$ , polinomul de feedback  $f(X)$  al **LFSR**-ului trebuie să fie *primitiv*. Aceasta înseamnă că  $f(X)$  trebuie ales astfel încât cel mai mic număr întreg pozitiv  $T$  pentru care  $X^T - 1$  este divizibil cu  $f(X)$  să fie  $T = 2^n - 1$ . Există algoritmi care testează primitivismul unui polinom. Numărul de polinoame primitive de grad  $n$  este:

$$N_p(n) = \frac{\Phi(2^n - 1)}{n},$$

unde  $\Phi(x)$ , cunoscută ca *funcția lui Euler*, desemnează cardinalul de numere naturale mai mici ca  $x$  și relativ prime cu  $x$ . Observăm că dacă un polinom  $f(X)$  este primitiv atunci și polinomul *reciproc* lui adică  $X^n f(\frac{1}{X})$  este primitiv. Se știe că orice polinom primitiv este ireductibil. Reciproca nu este adevărată. Numărul de polinoame ireductibile de grad  $n$  în algebra mod  $p$  ( $p = 2$ ) este dat de formula următoare:

$$N_I(n) = \frac{1}{n} \sum_{d|n} p^d \mu\left(\frac{n}{d}\right),$$

unde  $\mu$  este *funcția lui Möbius* definită în felul următor pentru  $n = \prod_1^k p_i^{\alpha_i}$ :  $\mu(n) = 0$  dacă

$\prod_i^k \alpha_i > 1$ ,  $\mu(n) = (-1)^k$  dacă  $n$  este produsul a  $k$  numere prime distincte și  $\mu(1) = 1$ .

Legătura între funcția lui Moebius și funcția lui Euler este dată de:

$$\frac{\phi(n)}{n} = \sum_{d|n} \frac{\mu(d)}{d}.$$

Dacă  $k$  este un *număr prim Mersenne*, adică  $k$  este număr prim de forma  $2^n - 1$  unde  $n$  este număr prim, atunci orice polinom ireductibil de grad  $k$  (în algebra mod 2) este primitiv:

$$\begin{aligned} N_I(k) &= \frac{1}{2^n - 1} \sum_{d|2^n - 1} 2^d \mu\left(\frac{2^n - 1}{d}\right) = \frac{1}{2^n - 1} [-2 + 2^{2^n - 1}] \\ &= \frac{\Phi(2^{2^n - 1} - 1)}{2^n - 1} = N_P(k). \end{aligned}$$

## 8.2 Exerciții rezolvate

**Exercițiul 8.2.1** *O secvență determinată de polinomul de feedback  $1 + X^3 + X^4$  are perioadă maximă?*

*Rezolvare:* Notăm cu  $\alpha = X \bmod f(X)$  o rădăcină a polinomului de feedback:  $1 + \alpha^3 + \alpha^4 = 0$ . Succesiv obținem puterile lui  $\alpha$ :

$$\alpha^1 = \alpha;$$

$$\alpha^2 = \alpha^2;$$

$$\alpha^3 = \alpha^3;$$

$$\alpha^4 = 1 + \alpha^3;$$

$$\alpha^5 = \alpha\alpha^4 = \alpha(1 + \alpha^3) = 1 + \alpha + \alpha^3;$$

$$\alpha^6 = \alpha\alpha^5 = \alpha(1 + \alpha + \alpha^3) = 1 + \alpha + \alpha^2 + \alpha^3;$$

$$\alpha^7 = \alpha\alpha^6 = \alpha(1 + \alpha + \alpha^2 + \alpha^3) = 1 + \alpha + \alpha^2;$$

$$\alpha^8 = \alpha\alpha^7 = \alpha(1 + \alpha + \alpha^2) = \alpha + \alpha^2 + \alpha^3;$$

$$\alpha^9 = \alpha\alpha^8 = \alpha(\alpha + \alpha^2 + \alpha^3) = 1 + \alpha^2;$$

$$\alpha^{10} = \alpha\alpha^9 = \alpha(1 + \alpha^2) = \alpha + \alpha^3;$$

$$\alpha^{11} = \alpha\alpha^{10} = \alpha(\alpha + \alpha^3) = 1 + \alpha^2 + \alpha^3;$$

$$\alpha^{12} = \alpha\alpha^{11} = \alpha(1 + \alpha^2 + \alpha^3) = 1 + \alpha;$$

$$\alpha^{13} = \alpha\alpha^{12} = \alpha(1 + \alpha) = \alpha + \alpha^2;$$

$$\alpha^{14} = \alpha\alpha^{13} = \alpha(\alpha + \alpha^2) = \alpha^2 + \alpha^3;$$

$$\alpha^{15} = \alpha\alpha^{14} = \alpha(\alpha^2 + \alpha^3) = 1.$$

Ordinul lui  $\alpha$  este  $2^4 - 1$ , în concluzie, polinomul de feedback este primitiv.

## 8.3 Exerciții propuse

**Exercițiul 8.3.1** *Implementați o rutină de testat primitivismul unui polinom din  $\mathbf{Z}_2[X]$ .*

**Exercițiul 8.3.2** *O secvență determinată de polinomul de feedback  $1 + X^2 + X^4$  are perioadă maximă?*

*Răspuns:* Nu. Polinomul nu este ireductibil, deci nu este primitiv.

**Exercițiul 8.3.3** *O secvență determinată de polinomul de feedback  $1 + X + X^4$  are perioadă maximă?*

*Răspuns:* Da. Polinomul de feedback este primitiv.

**Exercițiul 8.3.4** *O secvență determinată de polinomul de feedback  $1 + X + X^3$  are perioadă maximă?*

*Răspuns:* Da. Polinomul de feedback este primitiv.



**Exercițiul 8.3.5** *O secvență determinată de polinomul de feedback  $1 + X + X^2 + X^3$  are perioadă maximă?*

*Răspuns:* Nu. Polinomul nu este primitiv.

**Exercițiul 8.3.6** *O secvență determinată de polinomul de feedback  $1 + X^2 + X^5$  are perioadă maximă?*

*Răspuns:* Da. Polinomul de feedback este primitiv.

**Exercițiul 8.3.7** *O secvență determinată de polinomul de feedback  $1 + X + X^3 + X^4 + X^5$  are perioadă maximă?*

*Răspuns:* Da. Polinomul de feedback este primitiv.

**Exercițiul 8.3.8** *O secvență determinată de polinomul de feedback  $1 + X + X^3 + X^5$  are perioadă maximă?*

*Răspuns:* Nu. Polinomul nu este primitiv.

**Exercițiul 8.3.9** *O secvență determinată de polinomul de feedback  $1 + X + X^2 + X^3 + X^5$  are perioadă maximă?*

*Răspuns:* Da. Polinomul de feedback este primitiv.

**Exercițiul 8.3.10** *O secvență determinată de polinomul de feedback  $1 + X^2 + X^3 + X^4 + X^5$  are perioadă maximă?*

*Răspuns:* Da. Polinomul de feedback este primitiv.

# Capitolul 9

## Calculul în corpuri Galois

### 9.1 Breviar teoretic

Corpul Galois  $GF(2^n)$  este definit de un polinom  $f(X) \in \mathbb{Z}_2[X]$  de grad  $n$ . Elementele acestui corp sunt polinoame.

Operațiile între două polinoame  $a(X) = a_0 + a_1X + \dots + a_nX^n$  și  $b(X) = b_0 + b_1X + \dots + b_nX^n$  din  $GF(2^n)$  se definesc în modul următor:

- a)  $a(X) \oplus b(X) = c(X)$ ,  $c_i = (a_i + b_i) \bmod 2$ ;
- b)  $a(X) \bullet b(X) = a(X)b(X) \bmod f(X)$ .

Un element din  $GF(2^n)$  se poate reprezenta sub forma binară (și apoi hexazecimală) prin coeficienții săi :  $a_0 + a_1X + \dots + a_nX^n$  se identifică cu  $a_n \dots a_1a_0$ ,  $a_i \in \{0, 1\}$

Inversul unui element din  $GF(2^n)$  se determină cu algoritmul lui Euclid, exemplificat în continuare.

### 9.2 Exerciții rezolvate

**Exercițiul 9.2.1** *Care este inversul elementului  $\{45\}$  (reprezentat în format hexa) din  $GF(2^8)$  definit de polinomul  $f(X) = 1 + X + X^3 + X^4 + X^8$ .*

*Rezolvare:* Elementului  $\{45\}$  îi corespunde polinomul  $X^6 + X^2 + 1$ . Pentru a afla inversul lui  $\{45\} \bmod f(X)$  utilizăm algoritmul lui Euclid:

$$X^8 + X^4 + X^3 + X + 1 = X^2(X^6 + X^2 + 1) + X^3 + X^2 + X + 1,$$

$$X^6 + X^2 + 1 = (X^3 + X^2)(X^3 + X^2 + X + 1) + 1,$$

plecând de la ultima ecuație către prima, succesiv obținem:

$$1 = (X^3 + X^2)(X^3 + X^2 + X + 1) + X^6 + X^2 + 1$$

$$1 = (X^3 + X^2)(X^2(X^6 + X^2 + 1) + X^8 + X^4 + X^3 + X + 1) + X^6 + X^2 + 1$$

$$1 = (X^5 + X^4 + 1)(X^6 + X^2 + 1) + (X^3 + X^2 + 1)(X^8 + X^4 + X^3 + X + 1)$$

deci inversul polinomului  $X^6 + X^2 + 1$  este  $X^5 + X^4 + 1$ . Utilizând codificarea hexa ajungem la concluzia că inversul elementului  $\{45\}$  este  $\{31\}$ .

**Exercițiul 9.2.2** Să se adune elementele  $\{57\}$  și  $\{83\}$  în corpul Galois  $GF(2^8)$  definit de polinomul  $1 + X + X^3 + X^4 + X^8$ .

*Rezolvare:* Scrierea binară a celor două elemente este  $\{57\} = \{01010111\}$  respectiv  $\{83\} = \{10000011\}$ . Efectuând calculele obținem  $\{57\} \oplus \{83\} = \{11010100\} = \{D4\}$ .

**Exercițiul 9.2.3** Să se înmulțească elementele  $\{57\}$  și  $\{83\}$  în corpul Galois  $GF(2^8)$  definit de polinomul  $1 + X + X^3 + X^4 + X^8$ .

*Rezolvare:*  $\{57\} \bullet \{83\} = (X^6 + X^4 + X^2 + X + 1)(X^7 + X + 1) = X^{13} + X^{11} + X^9 + X^8 + X^6 + X^5 + X^4 + X^3 + 1 \bmod (X^8 + X^4 + X^3 + X + 1) = X^7 + X^6 + 1 = \{11000001\} = \{C1\}$ .

### 9.3 Exerciții propuse

**Exercițiul 9.3.1** Implementați proceduri de calcul în corp Galois.

**Exercițiul 9.3.2** Care este inversul elementului  $\{33\}$  (reprezentat în format hexa) din  $GF(2^8)$  definit de polinomul  $1 + X + X^3 + X^4 + X^8$ .

*Răspuns:*  $\{6C\}$ .

**Exercițiul 9.3.3** Care este inversul elementului  $\{12\}$  (reprezentat în format hexa) din  $GF(2^8)$  definit de polinomul  $1 + X + X^3 + X^4 + X^8$ .

*Răspuns:*  $\{AA\}$ .

**Exercițiul 9.3.4** Care este inversul elementului  $\{31\}$  (reprezentat în format hexa) din  $GF(2^8)$  definit de polinomul  $1 + X + X^3 + X^4 + X^8$ .

*Răspuns:*  $\{45\}$ .

**Exercițiul 9.3.5** Arătați că elementele  $\{12\}$  și  $\{AA\}$  (reprezentate în format hexa) sunt inverse în corpul Galois  $GF(2^8)$  definit de polinomul  $1 + X + X^3 + X^4 + X^8$ .

**Exercițiul 9.3.6** Să se adune elementele  $\{5\}$  și  $\{7\}$  în corpul Galois  $GF(2^4)$  definit de polinomul  $1 + X + X^4$ .

*Răspuns:*  $\{2\}$ .

**Exercițiul 9.3.7** Să se înmulțească elementele  $\{5\}$  și  $\{7\}$  în corpul Galois  $GF(2^4)$  definit de polinomul  $1 + X + X^4$ .

*Răspuns:*  $\{8\}$ .

**Exercițiul 9.3.8** *Se consideră transformarea dată de*

$$g(\mathbf{y}) = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \end{pmatrix} \mathbf{y}^{-1} \oplus \begin{pmatrix} 0 \\ 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \end{pmatrix} \quad (9.1)$$

unde  $\mathbf{y}^{-1}$  este inversul lui  $\mathbf{y}$  în corpul Galois  $GF(2^8)$  definit de polinomul  $1 + X + X^3 + X^4 + X^8$ . Calculați  $g(1), g(2), g(3), g(4), g(5)$ .

*Răspuns:* Transformarea indicată în problemă definește tabela de substituție a algoritmului RIJNDAEL. Valorile solicitate (în zecimal) sunt:  $g(1) = 124, g(2) = 119, g(3) = 123, g(4) = 242, g(5) = 107$ .



# Capitolul 10

## Algoritmul RIJNDAEL - Standardul AES

### 10.1 Breviar teoretic

Pentru rezolvarea următoarelor exerciții plecăm de la ipoteza cunoașterii standardului FIPS 197 - Advanced Encryption Standard compus din patru operații (sumare modulo 2 cu cheia de rundă, substituția la nivel de octet, shiftarea liniilor, mixarea coloanelor etc.) în cadrul procesului de transformare a stărilor și din generatorul de chei de rundă.

### 10.2 Exerciții rezolvate

**Exercițiul 10.2.1** Intrarea în runda  $i = 6$  a algoritmului AES 128/128 pentru cifrarea textului „zero peste tot”, cu ajutorul cheii „zero peste tot”, este:

$$\begin{bmatrix} D4 & 55 & 7E & 79 \\ 6F & B8 & 05 & 79 \\ 4F & 96 & BB & DE \\ 6C & 33 & 3D & 23 \end{bmatrix}$$

cheia de rundă fiind:

$$\begin{bmatrix} EC & 14 & 99 & 6A \\ 61 & 25 & FF & B4 \\ 4B & 75 & 09 & 9B \\ 85 & 8C & 37 & A7 \end{bmatrix}$$

Care este ieșirea după procesarea rutinelor *SubBytes*, *ShiftRows*, *MixColumns* și *AddRound-Key*?

*Rezolvare:*

Rutina SubBytes presupune folosirea următorului Sbox:

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
1	ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
2	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
3	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
4	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
5	53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
6	d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8
7	51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2
8	cd	0c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73
9	60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0b	db
a	e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
b	e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
c	ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
d	70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1d	9e
e	e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
f	8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16

Găsirea octetului din S-box corespunzător octetului din stare se face astfel: pentru octetul  $D4$  se caută în SBox elementul aflat la intersecția liniei  $D$  cu coloana 4 și se substituie în stare elementul găsit în Sbox.  $D4$  se va substitui cu 48. Procedeeul se aplică similar pentru ceilalți octeți din stare.

Rezultatul aplicării rutinei SubBytes se constituie în următoarea stare:

48	FC	F3	B6
A8	6C	6B	B6
84	90	EA	1D
50	C3	27	26

Rutina ShiftRows acționează în felul următor asupra stării: prima linie rămâne neschimbată, a doua linie se rotește la stânga cu un octet, a treia linie se rotește la stânga cu doi octeți iar a patra linie se rotește la stânga cu trei octeți.

După aplicarea rutinei ShiftRows, starea va fi următoarea:

48	FC	F3	B6
6C	6B	B6	A8
EA	1D	84	90
26	50	C3	27

Rutina MixColumns presupune înmulțirea fiecărei coloane din stare cu următoarea matrice fixată:

$$\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix}$$

Operațiile care rezultă din înmulțirea matricilor se fac în corpul Galois  $GF(2^8)$  și sunt înmulțiri de polinoame modulo polinomul generator al corpului  $GF(2^8)$  care este  $h(X) = X^8 + X^4 + X^3 + X + 1$ . Observăm că singurele înmulțiri care apar sunt cele cu 02 și 03. Înmulțirea cu polinomul 02 în  $GF(2^8)$  înseamnă înmulțirea cu polinomul  $X$ .

Fie  $f(X) = b_7X^7 + b_6X^6 + b_5X^5 + b_4X^4 + b_3X^3 + b_2X^2 + b_1X + b_0$  un polinom din  $GF(2^8)$ . Să vedem ce presupune înmulțirea  $02 * f(X)$  adică  $X * f(X)$ :

$$X * f(X) = b_7X^8 + b_6X^7 + b_5X^6 + b_4X^5 + b_3X^4 + b_2X^3 + b_1X^2 + b_0X \pmod{m(X)},$$

unde  $m(X)$  este polinomul generator  $m(X) = X^8 + X^3 + X + 1$  al corpului Galois  $GF(2^8)$ . Dacă  $b_7 = 0$ , atunci polinomul este în forma redusă în  $GF(2^8)$  (are gradul 7).

Dacă  $b_7 = 1$ , atunci:

$$X * f(X) = X^8 \pmod{m(X)} + b_6X^7 + b_5X^6 + b_4X^5 + b_3X^4 + b_2X^3 + b_1X^2 + b_0X.$$

Deci:

$$X * f(X) = (X^4 + X^3 + X + 1) + b_6X^7 + b_5X^6 + b_4X^5 + b_3X^4 + b_2X^3 + b_1X^2 + b_0X.$$

Prin urmare, înmulțirea cu polinomul  $X$  poate fi implementată, în cazul în care bitul cel mai semnificativ al polinomului  $f(X)$  este 1, ca o operație de shift la stânga cu 1 bit urmată de un XOR cu (00011011), care reprezintă polinomul  $(X^4 + X^3 + X + 1)$ .

Dacă bitul cel mai semnificativ al polinomului  $f(X)$  este 0, atunci înmulțirea presupune doar operație de shift la stânga cu un bit.

Pentru a trece starea curentă prin rutina MixColumns, se înmulțește pe rând fiecare coloană din stare cu matricea fixată de mai sus.

Vom prezenta doar modul de efectuare al înmulțirii:

$$\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \cdot \begin{bmatrix} 48 \\ 6C \\ EA \\ 26 \end{bmatrix}$$

Coloana rezultat va conține următoarele linii:

$$\begin{bmatrix} 02 * 48 \oplus 03 * 6C \oplus EA \oplus 26 \\ 01 * 48 \oplus 02 * 6C \oplus 03 * EA \oplus 26 \\ 48 \oplus 6C \oplus 02 * EA \oplus 03 * 26 \\ 03 * 48 \oplus 6C \oplus EA \oplus 02 * 26 \end{bmatrix}$$



Rămân de efectuat înmulțirile care apar pe fiecare linie:

$$02 * 48 = 02 * 01001000 = 10010000.$$

$$03 * 48 = 02 * 48 \oplus 48 = 11011000.$$

$$03 * 6C = 03 * 01101100 = 02 * 01101100 \oplus 01101100 = 11011000 \oplus 01101100 = 10110100.$$

$$02 * EA = 02 * 11101010 = 11010100 \oplus 00011011 = 11110001.$$

$$03 * EA = 02 * EA \oplus EA = 11110001 \oplus 11101010 = 00011011.$$

$$02 * 26 = 02 * 00100110 = 01001100.$$

$$03 * 26 = 02 * 26 \oplus 26 = 01001100 \oplus 00100110 = 01101010.$$

După calculele rămase, coloana rezultat va fi:

$$\begin{bmatrix} E8 \\ 93 \\ 81 \\ 12 \end{bmatrix}$$

Pentru celelalte coloane din stare se procedează similar.

Starea rezultată după aplicarea rutinei MixColumns este următoarea:

$$\begin{bmatrix} E8 & 13 & 7B & 23 \\ 93 & 5D & D0 & 71 \\ 81 & 5D & 08 & 4C \\ 12 & C9 & A1 & B7 \end{bmatrix}$$

Aplicarea rutinei AddRoundKey presupune o simplă operație de XOR pe fiecare octet din stare cu octet-ul corespunzător din cheia de rundă.

$$\begin{bmatrix} E8 & 13 & 7B & 23 \\ 93 & 5D & D0 & 71 \\ 81 & 5D & 08 & 4C \\ 12 & C9 & A1 & B7 \end{bmatrix} \oplus \begin{bmatrix} EC & 14 & 99 & 6A \\ 61 & 25 & FF & B4 \\ 4B & 75 & 09 & 9B \\ 85 & 8C & 37 & A7 \end{bmatrix} = \begin{bmatrix} 04 & 07 & E2 & 49 \\ F2 & 78 & 2F & C5 \\ CA & 28 & 01 & D7 \\ 97 & 45 & 96 & 10 \end{bmatrix}$$

## 10.3 Exerciții propuse

**Exercițiul 10.3.1** Intrarea în runda  $i = 7$  a algoritmului AES 128/128 pentru cifrarea textului „zero peste tot”, cu ajutorul cheii „zero peste tot”, este:

$$\begin{bmatrix} 04 & 07 & E2 & 49 \\ F2 & 78 & 2F & C5 \\ CA & 28 & 01 & D7 \\ 97 & 45 & 96 & 10 \end{bmatrix}$$

cheia de rundă fiind:

$$\begin{bmatrix} 21 & 35 & AC & C6 \\ 75 & 50 & AF & 1B \\ 17 & 62 & 6B & F0 \\ 87 & 0B & 3C & 9B \end{bmatrix}$$

Care este ieșirea după procesarea rutinelor *SubBytes*, *ShiftRows*, *MixColumns* și *AddRound-Key*?

*Răspuns:* Ieșirea din runda 7 este:

$$\begin{bmatrix} B7 & 1D & 6C & 94 \\ AA & 25 & 92 & E5 \\ E4 & 2D & 0F & 81 \\ C5 & 4F & 81 & 50 \end{bmatrix}$$

**Exercițiul 10.3.2** Intrarea în runda  $i = 8$  a algoritmului AES 128/128 pentru cifrarea textului „zero peste tot”, cu ajutorul cheii „zero peste tot”, este:

$$\begin{bmatrix} B7 & 1D & 6C & 94 \\ AA & 25 & 92 & E5 \\ E4 & 2D & 0F & 81 \\ C5 & 4F & 81 & 50 \end{bmatrix}$$

cheia de rundă fiind:

$$\begin{bmatrix} 0E & 3B & 97 & 51 \\ F9 & A9 & 06 & 1D \\ 03 & 61 & 0A & FA \\ 33 & 38 & 04 & 9F \end{bmatrix}$$

Care este ieșirea după procesarea rutinelor *SubBytes*, *ShiftRows*, *MixColumns* și *AddRound-Key*?

*Răspuns:* Ieșirea din runda 8 este:

$$\begin{bmatrix} 23 & 13 & AA & 2E \\ 37 & 21 & C0 & 03 \\ 8C & 63 & C6 & CB \\ 3C & DB & 57 & 95 \end{bmatrix}$$

**Exercițiul 10.3.3** Intrarea în runda  $i = 8$  a algoritmului AES 128/128 pentru cifrarea textului „zero peste tot”, cu ajutorul cheii „zero peste tot”, este:

$$\begin{bmatrix} 23 & 13 & AA & 2E \\ E7 & 21 & C0 & 03 \\ 8C & 63 & C6 & CB \\ 3C & DB & 57 & 95 \end{bmatrix}$$

cheia de rundă fiind:

$$\begin{bmatrix} B1 & 8A & 1D & 4C \\ D4 & 7D & 7B & 66 \\ D8 & B9 & B3 & 49 \\ E2 & DA & DE & 41 \end{bmatrix}$$

Care este ieșirea după procesarea rutinelor *SubBytes*, *ShiftRows*, *MixColumns* și *AddRound-Key*?

Răspuns: Ieșirea din runda 9 este:

$$\begin{bmatrix} 7F & 51 & 0E & 29 \\ FE & A5 & 34 & 29 \\ 0E & 66 & 7C & EC \\ 95 & 35 & 47 & CB \end{bmatrix}$$

**Exercițiul 10.3.4** Executați o rundă completă, pentru algoritmul RIJNDAEL (AES), cu următoarele intrări:

pentru starea curentă:

$$\begin{bmatrix} 00 & 00 & 00 & 00 \\ 00 & 00 & 00 & 00 \\ 00 & 00 & 00 & 00 \\ 00 & 00 & 00 & 01 \end{bmatrix}$$

pentru cheia de rundă:

$$\begin{bmatrix} 62 & 62 & 62 & 62 \\ 63 & 63 & 63 & 63 \\ 7C & 7C & 7C & 7C \\ 63 & 63 & 63 & 62 \end{bmatrix}$$

Răspuns: Ieșirea din rundă este:

$$\begin{bmatrix} 1E & 01 & 01 & 01 \\ 1F & 00 & 00 & 00 \\ 3E & 1F & 1F & 1F \\ 3E & 00 & 00 & 01 \end{bmatrix}$$

**Exercițiul 10.3.5** Executați o rundă completă, pentru algoritmul RIJNDAEL (AES), cu următoarele intrări:

pentru starea curentă:

$$\begin{bmatrix} 1E & 01 & 01 & 01 \\ 1F & 00 & 00 & 00 \\ 3E & 1F & 1F & 1F \\ 3E & 00 & 00 & 01 \end{bmatrix}$$

pentru cheia de rundă:

$$\begin{bmatrix} 9B & F9 & 9B & F9 \\ 73 & 10 & 73 & 10 \\ D6 & AA & D6 & AA \\ C9 & AA & C9 & AB \end{bmatrix}$$

Răspuns: ieșirea din rundă este:

$$\begin{bmatrix} 66 & D6 & 17 & F9 \\ E0 & 43 & 67 & CF \\ D8 & E3 & 13 & 28 \\ 04 & F2 & 5A & E9 \end{bmatrix}$$

**Exercițiul 10.3.6** Executați o rundă completă, pentru algoritmul RIJNDAEL (AES), cu următoarele intrări:

pentru starea curentă:

$$\begin{bmatrix} 66 & D6 & 17 & F9 \\ E0 & 43 & 67 & CF \\ D8 & E3 & 13 & 28 \\ 04 & F2 & 5A & E9 \end{bmatrix}$$

pentru cheia de rundă:

$$\begin{bmatrix} 55 & AC & 37 & CE \\ DF & CF & BC & AC \\ B4 & 1E & C8 & 62 \\ 50 & FA & 33 & 98 \end{bmatrix}$$

Răspuns: ieșirea din rundă este:

$$\begin{bmatrix} 7E & 09 & A1 & 70 \\ 41 & 86 & 69 & 61 \\ 45 & 08 & F0 & E1 \\ 5E & B5 & DA & BF \end{bmatrix}$$

**Exercițiul 10.3.7** Executați o rundă completă, pentru algoritmul RIJNDAEL (AES), cu următoarele intrări:

pentru starea curentă:

$$\begin{bmatrix} 7E & 09 & A1 & 70 \\ 41 & 86 & 69 & 61 \\ 45 & 08 & F0 & E1 \\ 5E & B5 & DA & BF \end{bmatrix}$$

pentru cheia de rundă:

$$\begin{bmatrix} CC & 60 & 57 & 99 \\ 75 & BA & 06 & AA \\ F2 & EC & 24 & 46 \\ DB & 21 & 12 & 8A \end{bmatrix}$$

Răspuns: ieșirea din rundă este:

$$\begin{bmatrix} 79 & D2 & A2 & C2 \\ 89 & 19 & 96 & E1 \\ 5E & 17 & 41 & 0D \\ 0D & 93 & 74 & 64 \end{bmatrix}$$

**Exercițiul 10.3.8** Executați o rundă completă, pentru algoritmul RIJNDAEL (AES), cu următoarele intrări:

pentru starea curentă:

$$\begin{bmatrix} 79 & D2 & A2 & C2 \\ 89 & 19 & 96 & E1 \\ 5E & 17 & 41 & 0D \\ 0D & 93 & 74 & 64 \end{bmatrix}$$

pentru cheia de rundă:

$$\begin{bmatrix} 70 & 10 & 47 & DE \\ 2F & 95 & 93 & 39 \\ 8C & 60 & 44 & 02 \\ 35 & 14 & 06 & 8C \end{bmatrix}$$

Răspuns: ieșirea din rundă este:

$$\begin{bmatrix} A0 & CA & A4 & 04 \\ F7 & AE & 76 & D0 \\ 36 & 92 & 49 & D6 \\ 25 & 22 & 4B & 8B \end{bmatrix}$$

# Capitolul 11

## Criptanaliza cifrurilor bloc

### 11.1 Breviar teoretic

Deoarece nu există o formulă matematică universală care să poată fi aplicată în operația de criptanaliză, am propus ca exerciții la acest capitol modificări ale unor algoritmi de cifruri bloc consacrate. Sunt date o serie de indicații precedate de o scurtă descriere a algoritmilor propriu-ziși.

### 11.2 Exerciții rezolvate

**Exercițiul 11.2.1** *Studiați următoarele simplificări ale algoritmului RC5:*

-RC5 cu 8 iterații dar fără rotații;

-RC5 cu 8 iterații iar numărul de rotații egal cu numărul de iterații.

*Răspuns.* În cele ce urmează facem o scurtă descriere a cifrului RC5 cu  $r$  iterații. Acesta are lungimea blocului de date variabilă dar vom considera în cele ce urmează că aceasta a fost setată la 64 biți. Operația de cifrare folosește  $2r + 2$  chei dependente de cuvintele pe 32 biți  $S_0, S_1, S_2, \dots, S_{2r+2}$  unde  $r$  este numărul de iterații. Pentru cifrare blocul de date se împarte în două părți de 32 biți notate cu  $L$  respectiv  $R$  (RC5 face apel la codificarea *little-endian* pentru împachetarea octeților în cuvinte: primul octet se transformă în cele mai puțin semnificative poziții ale lui  $L$ , etc.). Apoi avem:

$$\begin{cases} L = L + S_0, \\ R = R + S_1. \end{cases}$$

Pentru  $i = 1, \dots, r$  se execută:

$$\begin{cases} L = ((L \oplus R) \ll R) + S_{2i}, \\ R = ((R \oplus L) \ll L) + S_{2i+1}. \end{cases}$$

Ieșirea constă în registrele  $L$  și  $R$ . Simbolul  $\oplus$  are semnificația sumei mod 2, simbolul  $\ll$  semnifică rotire circulară și în fine simbolul  $+$  are semnificația sumei mod  $2^{32}$ . Operația de

decriptare este similară (intervin operatorii  $\oplus, \gg$  și  $-$ ). Modul de construcție al secvenței  $S$  (care derivă din cheie) nu este esențial în cadrul acestui exercițiu.

Dacă setăm numărul de iterații  $r = 8$  și nu facem nici un fel de rotații atunci pentru  $i = 1, \dots, 8$  se execută:

$$\begin{cases} L = (L \oplus R) + S_{2i}, \\ R = (R \oplus L) + S_{2i+1}. \end{cases}$$

Algoritmul astfel setat nu îndeplinește criteriul de avalanșă strictă (schimbarea unui bit în blocul de text clar produce, în medie, schimbări de 50% la ieșire). Schema de mai sus permite atacul cu ajutorul tehnicii criptanalizei liniare pentru aflarea lui  $S$ , deci a cheii efective.

Dacă setăm numărul de iterații  $r = 8$  și numărul de rotații egal cu  $r$  atunci pentru  $i = 1, \dots, 8$  se execută:

$$\begin{cases} L = ((L \oplus R) \ll 8) + S_{2i}, \\ R = ((R \oplus L) \ll 8) + S_{2i+1}. \end{cases}$$

Algoritmul astfel setat nu îndeplinește criteriul de avalanșă strictă. Schema de mai sus permite atacul cu ajutorul tehnicii criptanalizei diferențial/liniare pentru aflarea lui  $S$ .

**Exercițiul 11.2.2** *Studiați următoarele simplificări ale algoritmului DES:*

- DES cu 12 iterații dar fără aplicațiile  $S$ ;
- DES cu 4 iterații;
- DES cu 6 iterații.

*Răspuns.* Cifrul bloc DES (proiectat în 1977) este sub controlul unei chei efective de 56 biți (cheia de bază este de 64 biți, 8 biți fiind pentru detecția erorilor) iar mărimea blocului de date este de 64 biți. Textul clar este permutat iar apoi este împărțit în două blocuri  $L$  și  $R$  de lungime 32 biți. Se execută apoi iterativ operațiile (pentru  $i = 1, \dots, \text{numărul de iterații}$ ):

$$\begin{cases} L_i = R_i, \\ R_i = L_i \oplus f(R_{i-1}, K_i). \end{cases}$$

În final textul este supus permutării inverse. Ne concentrăm asupra descrierii funcției  $f : \mathbf{Z}_2^{32} \times \mathbf{Z}_2^{48} \rightarrow \mathbf{Z}_2^{32}$ . Inițial blocul  $R$  (32 biți) este extins cu ajutorul funcției  $E$  la un bloc pe 48 biți care este sumat mod2 cu cheia  $K$  (extinsă la 48 biți cu ajutorul algoritmului de producere a subcheilor). Opt aplicații  $S : \mathbf{Z}_2^6 \rightarrow \mathbf{Z}_2^4$  produc o ieșire pe 32 biți care este permutată pentru a produce ieșirea finală dintr-o iterație. Dacă aplicațiile  $S$  sunt fixe (se selectează 4 biți din 6 în mod fix) atunci se poate aplica tehnica criptanalizei diferențiale (biții de la ieșire sunt biții de la intrare (sumați mod2 cu cheia  $K$ ) dar într-o altă ordine).

Algoritmul DES cu 4 cât și cu 6 iterații poate fi spart cu ajutorul tehnicii atacului cu text clar cunoscut.

## 11.3 Exerciții propuse

**Exercițiul 11.3.1** *Studiați regula  $B$  a algoritmului Skipjack cu 8 iterații.*

**Exercițiul 11.3.2** *Ce defect are un algoritm de cifrare care este închis (un algoritm de cifrare se numește închis dacă pentru orice chei  $k_1$  și  $k_2$  există o cheie  $k_3$  astfel încât pentru orice text clar  $M$  avem  $E_{k_1}E_{k_2}(M) = E_{k_3}(M)$ )?*

*Răspuns.* Ca metodă de atac generică se poate opta pentru cifrarea repetitivă.

**Exercițiul 11.3.3** *Aplicați tehnica criptanalizei diferențiale și criptanalizei liniare asupra algoritmului FEAL.*

**Exercițiul 11.3.4** *Studiați tehnica criptanalizei diferențiale în cazul algoritmului DES cu 16 iterații.*

**Exercițiul 11.3.5** *Aplicați tehnica criptanalizei liniare în cazul algoritmului DES cu 16 iterații.*

**Exercițiul 11.3.6** *Având la dispoziție un cifru bloc  $E_k(\cdot)$  proiectați un cifru flux și viceversa.*

**Exercițiul 11.3.7** *Scrieți funcția analitică a celor opt funcții de substituție  $S$  ale cifrului DES.*

**Exercițiul 11.3.8** *Fie  $E_M(\cdot)$  și  $D_K(\cdot)$  funcțiile de cifrare respectiv descifrare ale unui cifru. Care este valoarea lui  $D_K(E_K(M))$ ?*

*Notă.* Descrierea algoritmilor RC5, DES, Skipjack și FEAL poate fi găsită în Schneier [8] sau Menezes [4].

**Exercițiul 11.3.9** *Implementați modalități de testare a cifrurilor bloc.*

**Exercițiul 11.3.10** *Implementați modalități de generare a tabelor de substituție.*

**Exercițiul 11.3.11** *Fie  $E(\cdot, \cdot)$  o funcție de cifrare pe  $m$  biți de cheie și  $n$  biți de date. Care este valoarea maximă a lui  $m$  astfel încât cheia efectivă a cifrului să fie  $m$ ?*





# Capitolul 12

## Lema chinezească a resturilor

### 12.1 Breviar teoretic

**Teorema 12.1** (*Lema chinezească a resturilor- CRT*) Fie  $m_1, \dots, m_k$  numere întregi cu  $(m_i, m_j) = 1$  pentru orice  $i \neq j$ . Atunci sistemul

$$x \equiv a_i \pmod{m_i}$$

are o soluție unică modulo  $\prod_{i=1}^k m_i$ .

*Demonstrație. Existența soluției.* Vom nota

$$M = \prod_{i=1}^k m_i$$

și

$$M_i = \frac{M}{m_i} \text{ pentru orice } i = 1, \dots, k.$$

Deoarece  $(m_i, m_j) = 1$  pentru orice  $i \neq j$  avem  $(M_j, m_j) = 1$  pentru orice  $j$  adică există  $N_j$  astfel ca  $M_j N_j = 1 \pmod{m_j}$ . Atunci dacă notăm

$$x = \sum_{i=1}^k a_i M_i N_i$$

și reducem modulo  $m_i$  avem:

$$x \equiv \sum_{j=1}^k a_j M_j N_j \pmod{m_i} \text{ pentru orice } i.$$

Folosind faptul că  $(M_i, m_j) \neq 1$  pentru  $i \neq j$  obținem:

$$\begin{aligned} x &= a_i M_i N_i \bmod m_i \\ &= a_i \bmod m_i \text{ pentru orice } i. \end{aligned}$$

*Unicitatea soluției.* Fie  $x'$  și  $x''$  două soluții atunci

$$x = x' - x'' = 0 \bmod m_i \text{ pentru orice } i$$

deci

$$x = 0 \bmod M.$$

## 12.2 Exerciții rezolvate

**Exercițiul 12.2.1** Să se rezolve sistemul de ecuații:

$$\begin{cases} x \equiv 3 \bmod 13 \\ x \equiv 34 \bmod 47 \\ x \equiv 2 \bmod 51 \end{cases}$$

*Rezolvare:*

Soluția sistemului de congruențe este dată de formula:

$$x = \sum_{j=1}^3 a_j M_j N_j \bmod M.$$

unde  $a_1 = 3, a_2 = 34, a_3 = 2$  iar  $m_1 = 13, m_2 = 47, m_3 = 51$ . Se observă că  $m_1, m_2$  și  $m_3$  sunt prime între ele.

Calculăm  $M = 13 \cdot 47 \cdot 51 = 31161$  și  $M_1 = 47 \cdot 51 = 2397, M_2 = 13 \cdot 51 = 663$  și  $M_3 = 13 \cdot 47 = 611$ .

Mai departe trebuie calculat inversul lui  $M_j$  pentru  $j = 1, j = 2$  și  $j = 3$ .

Cu algoritmul lui Euclid extins, se calculează  $N_1 = M_1^{-1} \bmod m_1 = 2397^{-1} \bmod 13 = 5^{-1} \bmod 13 = 8$ .

Similar se calculează  $N_2 = M_2^{-1} \bmod m_2 = 663^{-1} \bmod 47 = 5^{-1} \bmod 47 = 19$ , iar

$N_3 = M_3^{-1} \bmod m_3 = 611^{-1} \bmod 51 = 50^{-1} \bmod 51 = 50$ .

În acest moment, avem toate datele necesare pentru a calcula soluția  $x$  a sistemului de congruențe:

$$x = a_1 M_1 N_1 + a_2 M_2 N_2 + a_3 M_3 N_3 \bmod M.$$

Deci  $x = 3 \cdot 2397 \cdot 8 + 34 \cdot 663 \cdot 19 + 2 \cdot 611 \cdot 50 \bmod 31161 = 57528 + 428928 + 61100 \bmod 31161$  de unde  $x = 17819 \bmod 31161$ ; se poate verifica faptul că într-adevăr aceasta este soluția sistemului.

## 12.3 Exerciții propuse

**Exercițiul 12.3.1** *Să se rezolve sistemul de ecuații:*

$$\begin{cases} x \equiv 1 \pmod{13} \\ x \equiv 2 \pmod{17} \\ x \equiv 3 \pmod{11} \end{cases}$$

*Răspuns:*  $x = 1158 \pmod{2431}$ .

**Exercițiul 12.3.2** *Să se rezolve sistemul de ecuații:*

$$\begin{cases} x \equiv 3 \pmod{13} \\ x \equiv 2 \pmod{11} \\ x \equiv 2 \pmod{19} \end{cases}$$

*Răspuns:*  $x = 211 \pmod{2717}$ .

**Exercițiul 12.3.3** *Să se rezolve sistemul de ecuații:*

$$\begin{cases} x \equiv 3 \pmod{5} \\ x \equiv 5 \pmod{7} \\ x \equiv 7 \pmod{11} \end{cases}$$

*Răspuns:*  $x = 348 \pmod{385}$ .

**Exercițiul 12.3.4** *Să se rezolve sistemul de ecuații:*

$$\begin{cases} x \equiv 5 \pmod{17} \\ x \equiv 3 \pmod{19} \\ x \equiv 2 \pmod{23} \end{cases}$$

*Răspuns:*  $x = 991 \pmod{7429}$ .

**Exercițiul 12.3.5** *Să se rezolve sistemul de ecuații:*

$$\begin{cases} x \equiv 5 \pmod{11} \\ x \equiv 3 \pmod{19} \\ x \equiv 2 \pmod{23} \end{cases}$$

*Răspuns:*  $x = 3613 \pmod{4807}$ .

**Exercițiul 12.3.6** *Să se rezolve sistemul de ecuații:*

$$\begin{cases} x \equiv 5 \pmod{17} \\ x \equiv 3 \pmod{21} \\ x \equiv 2 \pmod{23} \end{cases}$$

*Răspuns:*  $x = 4119 \bmod 8211$ .

**Exercițiul 12.3.7** *Să se rezolve sistemul de ecuații:*

$$\begin{cases} x \equiv 4 \pmod{21} \\ x \equiv 9 \pmod{31} \\ x \equiv 14 \pmod{23} \end{cases}$$

*Răspuns:*  $x = 6178 \bmod 14973$ .

**Exercițiul 12.3.8** *Să se rezolve sistemul de ecuații:*

$$\begin{cases} x \equiv 4 \pmod{47} \\ x \equiv 9 \pmod{11} \\ x \equiv 3 \pmod{23} \end{cases}$$

*Răspuns:*  $x = 10767 \bmod 11891$ .

**Exercițiul 12.3.9** *Să se rezolve sistemul de ecuații:*

$$\begin{cases} x \equiv 11 \pmod{17} \\ x \equiv 12 \pmod{19} \\ x \equiv 13 \pmod{23} \end{cases}$$

*Răspuns:*  $x = 3394 \bmod 7429$ .

**Exercițiul 12.3.10** *Să se rezolve sistemul de ecuații:*

$$\begin{cases} x \equiv 8 \pmod{23} \\ x \equiv 14 \pmod{29} \\ x \equiv 17 \pmod{31} \end{cases}$$

*Răspuns:*  $x = 1319 \bmod 20677$ .

**Exercițiul 12.3.11** *Să se rezolve sistemul de ecuații:*

$$\begin{cases} x \equiv 15 \pmod{23} \\ x \equiv 3 \pmod{19} \\ x \equiv 13 \pmod{36} \end{cases}$$

*Răspuns:*  $x = 15241 \bmod 15732$ .

# Capitolul 13

## Sistemul de cifrare Merkle-Hellman

### 13.1 Breviar teoretic

Algoritmul de cifrare *Merkle-Hellman* constă în codificarea mesajului ca o soluție a unei probleme de tip rucsac pentru care ponderile  $\{M_1, \dots, M_n\}$  constituie cheia de cifrare, și textului clar  $\{b_1, \dots, b_n\}$  îi corespunde textul cifrat  $\sum_{i=1}^n b_i M_i$ .

**Definiția 13.1** Un șir de ponderi  $\{M_1, \dots, M_n\}$  se numește *supercrescător* dacă:

$$M_k > \sum_{i=1}^{k-1} M_i \text{ pentru orice } k. \quad (13.1)$$

Problema rucsacului supercrescător este ușor de rezolvat folosind următoarea schemă: pentru  $k = n, \dots, 1$ :

- dacă  $M_k < S$  atunci  $b_k = 1$  și  $S = S - M_k$ ;
- altfel  $b_k = 0$ .

Algoritmii de tip rucsac care nu sunt supercrescători nu sunt ușor de rezolvat și nu există niciun algoritm rapid care să rezolve problema. Singura modalitate cunoscută de a determina dacă  $b_i = 1$  constă în testarea tuturor soluțiilor. Cei mai rapizi algoritmi de testare au o complexitate exponențială.

Algoritmul Merkle-Hellman se bazează pe această proprietate: *cheia privată* este șirul ponderilor pentru un rucsac supercrescător iar *cheia publică* este șirul ponderilor pentru un rucsac care are aceeași soluție, dar nu este supercrescător. *Merkle* și *Hellman* au găsit o metodă prin care se poate transforma o problemă a rucsacului supercrescător într-o problemă normală a rucsacului. Tehnica de conversie face apel la aritmetica modulară.

Având la dispoziție o problemă de tip rucsac supercrescător (cheia privată) cu ponderile  $\{M_1, \dots, M_n\}$  atunci aceasta se transformă într-o problemă de tip rucsac normală (cheia publică) cu șirul ponderilor

$$\{mM_1 \bmod p, \dots, mM_n \bmod p\},$$

unde  $m$  și  $p$  sunt numere naturale prime între ele (acestea fac parte din cheia privată) și  $p > \sum_{i=1}^n M_i$ .

Pentru a cifra un mesaj binar acesta se va împărți în blocuri de lungimi egale cu cardinalul mulțimii ponderilor. Cifrarea unui bloc  $b_1 \dots b_n$  va fi numărul natural:

$$\sum_{i=1}^n b_i (mM_i \bmod p).$$

Pentru descifrare destinatarul mesajului cunoaște cheia privată: ponderile originale și valorile lui  $m$  și  $p$ . Acesta va calcula mai întâi pe  $m^{-1} \bmod p$ . Se va multiplica apoi textul cifrat cu  $m^{-1} \bmod p$  iar după aceea se va rezolva problema rucsacului supercrescător pentru a recupera textul original.

## 13.2 Exerciții rezolvate

**Exercițiul 13.2.1** Să se construiască cheia publică pentru algoritmul Merkle-Hellman reprezentat de cheia privată  $\{2, 3, 6, 13, 27, 52\}$ , modulul  $p = 105$  și multiplicatorul  $m = 31$ . Cifrați mesajul 101110.

*Rezolvare:*

Având la dispoziție cheia privată  $\{M_1, \dots, M_n\}$ , cheia publică se obține astfel  $\{mM_1 \bmod p, \dots, mM_n \bmod p\}$ .

Prin urmare, cheia privată pentru datele de mai sus este  $\{31 \cdot 2 \bmod 105, 31 \cdot 3 \bmod 105, 31 \cdot 6 \bmod 105, 31 \cdot 13 \bmod 105, 31 \cdot 27 \bmod 105, 31 \cdot 52 \bmod 105\}$  adică  $\{62, 93, 81, 88, 102, 37\}$ .

Cifrarea mesajului 101110 ( $(m_1, \dots, m_6)$ ) se face după formula  $\sum_{i=1}^n m_i (mM_i \bmod p)$ , adică pe baza cheii publice. Rezultatul va fi  $62 + 81 + 88 + 102$ , deci mesajul cifrat este  $c = 333$ .

**Exercițiul 13.2.2** Să se descifreze mesajul  $C = 4608$  cifrat cu ajutorul algoritmului Merkle-Hellman cu următorii parametri:  $n = 9$ , cheia privată  $\{1, 2, 5, 10, 19, 40, 98, 179, 355\}$ , modulul  $p = 1717$  și multiplicatorul  $m = 507$ .

*Rezolvare:* Se determină  $C \cdot m^{-1} \bmod 1717 = 4608 \cdot 507^{-1} \bmod 1717 = 4608 \cdot 657 \bmod 1717 = 385$ .

Apoi se rezolvă problema supercrescătoare a rucsacului de dimensiune  $385 : 385 = 355 + 19 + 10 + 1$ . Mesajul clar va conține 1 pe pozițiile corespunzătoare acestor ponderi, deci se obține 100110001.

## 13.3 Exerciții propuse

**Exercițiul 13.3.1** Dezvoltați o aplicație care să implementeze funcțiile de cifrare/descifrare ale sistemului Merkle-Hellman.

**Exercițiul 13.3.2** Să se construiască cheia publică pentru algoritmul Merkle-Hellman reprezentat de cheia privată  $\{2, 3, 6, 13, 27, 52\}$ , modulul  $p = 105$  și multiplicatorul  $m = 31$ . Cifrați mesajul 011111.

*Răspuns:* Cheia publică  $\{62, 93, 81, 88, 102, 37\}$ , mesajul cifrat  $c = 401$ .

**Exercițiul 13.3.3** Să se construiască cheia publică pentru algoritmul Merkle-Hellman reprezentat de cheia privată  $\{2, 3, 6, 13, 27, 52\}$ , modulul  $p = 105$  și multiplicatorul  $m = 31$ . Cifrați mesajul 111110.

*Răspuns:* Cheia publică  $\{62, 93, 81, 88, 102, 37\}$ , mesajul cifrat  $c = 426$ .

**Exercițiul 13.3.4** Să se construiască cheia publică pentru algoritmul Merkle-Hellman reprezentat de cheia privată  $\{2, 3, 6, 13, 27, 52\}$ , modulul  $p = 105$  și multiplicatorul  $m = 31$ . Cifrați mesajul 001110.

*Răspuns:* Cheia publică  $\{62, 93, 81, 88, 102, 37\}$ , mesajul cifrat  $c = 271$ .

**Exercițiul 13.3.5** Să se descifreze mesajul 333 cifrat cu ajutorul algoritmului Merkle-Hellman cu următorii parametrii:  $n = 6$ , cheia privată  $\{2, 3, 6, 13, 27, 52\}$ , modulul  $p = 105$  și multiplicatorul  $m = 31$ .

*Răspuns:* Cheia publică  $\{62, 93, 81, 88, 102, 37\}$ , mesajul clar 101110.

**Exercițiul 13.3.6** Să se descifreze mesajul 320 cifrat cu ajutorul algoritmului Merkle-Hellman cu următorii parametrii:  $n = 6$ , cheia privată  $\{2, 5, 14, 23, 56, 125\}$ , modulul  $p = 228$  și multiplicatorul  $m = 191$ .

*Răspuns:* Cheia publică  $\{154, 43, 166, 61, 208, 163\}$ ,  $m^{-1} \bmod p = 191$ , mesajul clar 101000.

**Exercițiul 13.3.7** Să se construiască cheia publică pentru algoritmul Merkle-Hellman cu următorii parametrii:  $n = 6$ , cheia privată  $\{3, 4, 11, 25, 50, 113\}$ , modulul  $p = 209$  și multiplicatorul  $m = 20$ . Cifrați mesajul 27.

*Răspuns:* Cheia publică este  $\{60, 80, 11, 82, 164, 170\}$ , mesajul cifrat 425.

**Exercițiul 13.3.8** Să se descifreze mesajul 425 cifrat cu ajutorul algoritmului Merkle-Hellman cu următorii parametrii:  $n = 6$ , cheia privată  $\{3, 4, 11, 25, 50, 113\}$ , modulul  $p = 209$  și multiplicatorul  $m = 20$ .



*Răspuns:* Cheia publică  $\{60, 80, 11, 82, 164, 170\}$ ,  $m^{-1} \bmod p = 115$ , mesajul clar 011011.

**Exercițiul 13.3.9** *Să se construiască cheia publică pentru algoritmul Merkle-Hellman cu următorii parametrii:  $n = 6$ , cheia privată  $\{3, 4, 11, 26, 58, 106\}$ , modulul  $p = 238$  și multiplicatorul  $m = 167$ . Cifrați mesajul 29.*

*Răspuns:* Cheia publică este  $\{25, 192, 171, 58, 166, 90\}$ , mesajul cifrat 511.

**Exercițiul 13.3.10** *Să se descifreze mesajul 511 cifrat cu ajutorul algoritmului Merkle-Hellman cu următorii parametrii:  $n = 6$ , cheia privată  $\{3, 4, 11, 26, 58, 106\}$ , modulul  $p = 238$  și multiplicatorul  $m = 167$ .*

*Răspuns:* Cheia publică  $\{25, 192, 171, 58, 166, 90\}$ ,  $m^{-1} \bmod p = 181$ , mesajul clar 011101.

# Capitolul 14

## Sistemul de cifrare RSA

### 14.1 Breviar teoretic

Algoritmul *RSA* a fost inventat de către *Ron Rivest*, *Adi Shamir* și *Leonard Adleman* și a fost studiat în cadrul unor studii criptanalitice extinse. Securitatea RSA-ului se bazează pe dificultatea factorizării numerelor mari. Cheia publică și cheia privată sunt funcție de o pereche de numere prime mari (de 200 de cifre sau chiar mai mari). Factorizarea produsului a două numere prime implică recuperarea textului clar din textul cifrat, cunoscând cheia publică.

Pentru generarea a două chei (publică și privată) se aleg aleatoriu două numere prime mari  $p$  și  $q$ . Din raționamente de securitate  $p$  și  $q$  au același ordin de mărime. Se va calcula produsul  $n = p \cdot q$ . Se va alege apoi, aleatoriu, exponentul public (de cifrare)  $e$  astfel ca  $e$  și  $(p - 1)(q - 1)$  să fie relativ prime. Utilizând algoritmul extins al lui Euclid vom calcula exponentul privat (de descifrare)  $d$  astfel ca

$$ed \equiv 1 \pmod{(p - 1)(q - 1)}.$$

Cu alte cuvinte

$$d \equiv e^{-1} \pmod{(p - 1)(q - 1)}.$$

Remarcăm faptul că  $d$  și  $n$  sunt relativ prime. Perechea  $(e, n)$  constituie cheia publică iar  $(d, p, q)$  este cheia privată. Cele două numere  $p$  și  $q$  nu mai sunt necesare la cifrare/descifrare, dar nu vor fi niciodată făcute publice (cunoașterea lor și a exponentului de cifrare  $e$  conduce imediat la determinarea coeficientului de descifrare  $d$ , deci sistemul de criptare devine inutil).

Pentru a cifra un mesaj  $M$  îl vom diviza în blocuri de lungime mai mică  $n$  (cu date binare vom alege cea mai mare putere a lui 2 mai mică decât  $n$ ). Dacă  $p$  și  $q$  sunt numere prime de 100 cifre atunci  $n$  va avea sub 200 de cifre iar fiecare mesaj bloc  $M_i$  va avea sub 200 de cifre. Dacă trebuie cifrate blocuri de lungime fixă atunci vom apela la operația de padding cu zero. Mesajul cifrat  $C$  se va obține prin concatenarea mesajelor  $C_i$  care au aproximativ aceeași lungime. Formula de cifrare va fi:

$$C_i \equiv M_i^e \pmod{n}.$$

Pentru a descifra un mesaj se calculează:

$$M_i \equiv C_i^d \pmod{n},$$

deoarece

$$\begin{aligned} C_i^d &\equiv (M_i^e)^d \equiv M_i^{ed} \equiv M_i^{k(p-1)(q-1)+1} \\ &\equiv M_i M_i^{k(p-1)(q-1)} \equiv M_i \pmod{n}. \end{aligned}$$

**Observația 14.1** Pentru a evita metodele de factorizare cunoscute numerele  $p$  și  $q$  trebuie să fie numere prime tari. Un număr prim  $p$  se numește număr prim tare dacă:

- i)  $p - 1$  are un factor mare  $r$ ;
- ii)  $p + 1$  are un factor mare  $s$ ;
- iii)  $r - 1$  are un factor mare  $t$ .

Operația de semnare a unui mesaj  $M$  se realizează prin exponențierea amprentei  $H(M)$  cu ajutorul cheii private:  $s = H(M)^d \pmod{n}$ . Verificarea semnăturii se realizează prin comparația lui  $H(M)$  cu  $s^e \pmod{n}$ .

În cazurile practice valoarea lui  $e$  este un număr relativ mic, deci  $d$  are o valoare mare. Acest lucru conduce la timpi de rulare diferiți între operațiile private (descifrare/semnare) și cele publice(cifrare/verificare semnătură).

Pentru optimizarea calculelor de verificare a semnăturii se poate utiliza lema chinezească a resturilor (CRT), însă acest lucru induce vulnerabilități în mediul de implementare.

Astfel, dacă  $p > q$ , sunt **precalculate** valorile:

$$\begin{aligned} dP &= (e^{-1} \pmod{n}) \pmod{(p-1)}, \\ dQ &= (e^{-1} \pmod{n}) \pmod{(q-1)}, \\ qInv &= q^{-1} \pmod{p}. \end{aligned}$$

În faza de calcul se execută:

$$\begin{aligned} m_1 &= c^{dP} \pmod{p}, \\ m_2 &= c^{dQ} \pmod{q}, \\ h &= qInv(m_1 - m_2) \pmod{p}, \\ m &= m_2 + hq. \end{aligned}$$

Cheia privată ce se stochează fiind  $(p, q, dP, dQ, qInv)$ .

## 14.2 Exerciții rezolvate

**Exercițiul 14.2.1** Se dă numărul  $n = 36187829$  despre care se cunoaște faptul că este un produs de două numere cu valoarea  $\phi(n) = 36175776$ . Factorizați numărul  $n$ .

*Rezolvare:* Folosim relațiile  $p + q = n - (p - 1)(q - 1) + 1$  și  $p - q = \sqrt{(p + q)^2 - 4n}$ . Obținem  $p = 5657$  și  $q = 6397$ .

**Exercițiul 14.2.2** Să se cifreze mesajul  $M = 3$ , utilizând sistemul RSA cu următorii parametri:  $N = 187$  (modulul de cifrare),  $e = 7$  (exponentul de cifrare).

*Rezolvare:* Criptograma este:  $C = M^e = 3^7 = 2187 = 130 \bmod 187$ .

**Exercițiul 14.2.3** Să se descifreze mesajul  $C = 130$ , utilizând sistemul RSA cu următorii parametri:  $N = 187 = 11 \cdot 17$  (modulul de cifrare),  $e = 7$  (exponentul de cifrare).

*Rezolvare:* Deoarece se cunoaște factorizarea  $N = 11 \cdot 17$ , se poate calcula  $\varphi(N) = 16 \cdot 10 = 160$ ,  $\varphi(\varphi(N)) = 64$ .

Exponentul de descifrare va fi:

$$d = e^{\varphi(\varphi(N)) - 1} = 7^{63} = (7^9)^7 = (40353607)^7 = 7^7 = 823543 = 23 \bmod 160.$$

Descifrarea mesajului cifrat  $C$  va fi:  $C^d = 130^{23} = 3 = M \bmod 187$ .

**Exercițiul 14.2.4** Să se descifreze, utilizând CRT, mesajul cifrat  $c = 8363$ , pentru cazul în care  $p = 137$ ,  $q = 131$ ,  $n = p \cdot q = 17947$ ,  $e = 3$ ,  $d = 11787$ .

*Rezolvare:* În faza de precalcul avem:

$$\begin{aligned} dP &= (e^{-1} \bmod n) \bmod (p - 1) = 91, \\ dQ &= (e^{-1} \bmod n) \bmod (q - 1) = 87, \\ qInv &= q^{-1} \bmod p = 114. \end{aligned}$$

Calculăm apoi:

$$\begin{aligned} m_1 &= c^{dP} \bmod p = 102, \\ m_2 &= c^{dQ} \bmod q = 120, \\ h &= qInv(m_1 - m_2) \bmod p = 3, \\ m &= m_2 + hq = 513. \end{aligned}$$

## 14.3 Exerciții propuse

**Exercițiul 14.3.1** Fie numerele prime  $p = 211$  și  $q = 167$ . Să se cifreze mesajul TEST cu ajutorul algoritmului RSA, utilizând exponentul public  $e = 2^8 + 1$ . Elementele din mesajul clar se codifică conform codului ASCII.

*Răspuns:*  $N = 35237$ ,  $\phi(N) = 34860$ ,  $d = 23873$ , mesajul cifrat este: 01154 05746 04357 01154.

**Exercițiul 14.3.2** Să se descifreze mesajul 01154 05746 04357 01154 cu ajutorul algoritmului RSA ( $p = 211$  și  $q = 167$ ), utilizând exponentul public  $e = 2^8 + 1$ . Elementele din mesajul clar se decodifică conform codului ASCII.

Răspuns:  $N = 35237$ ,  $\phi(N) = 34860$ ,  $d = 23873$ , mesajul clar este TEST.

**Exercițiul 14.3.3** Să se cifreze mesajul  $M = 146$ , utilizând sistemul RSA cu următorii parametrii:  $n = 187$  (modulul de cifrare),  $e = 7$  (exponentul de cifrare).

Răspuns:  $C = 141$ .

**Exercițiul 14.3.4** Să se descifreze mesajul  $C = 141$ , utilizând sistemul RSA cu următorii parametrii:  $n = 187$  (modulul de cifrare),  $d = 23$  (exponentul de descifrare).

Răspuns:  $M = 146$ .

**Exercițiul 14.3.5** Să se cifreze mesajul  $M = 9$ , utilizând sistemul RSA cu următorii parametrii:  $n = 187$  (modulul de cifrare),  $e = 7$  (exponentul de cifrare).

Răspuns:  $C = 70$ .

**Exercițiul 14.3.6** Să se descifreze mesajul  $C = 70$ , utilizând sistemul RSA cu următorii parametrii:  $n = 187$  (modulul de cifrare),  $d = 23$  (exponentul de descifrare).

Răspuns:  $M = 9$ .

**Exercițiul 14.3.7** Să se cifreze mesajul  $M = 3$ , utilizând sistemul RSA cu următorii parametrii:  $n = 35237$  (modulul de cifrare),  $e = 11$  (exponentul de cifrare).

Răspuns:  $C = 962$ .

**Exercițiul 14.3.8** Să se descifreze mesajul  $C = 962$ , utilizând sistemul RSA cu următorii parametrii:  $n = 35237$  (modulul de cifrare),  $d = 31691$  (exponentul de descifrare).

Răspuns:  $M = 3$ .

**Exercițiul 14.3.9** Să se cifreze mesajul  $M = 5$ , utilizând sistemul RSA cu următorii parametrii:  $n = 221$  (modulul de cifrare),  $e = 11$  (exponentul de cifrare).

Răspuns:  $C = 164$ .

**Exercițiul 14.3.10** Să se descifreze mesajul  $C = 164$ , utilizând sistemul RSA cu următorii parametrii:  $n = 221 = 13 \cdot 17$  (modulul de cifrare),  $e = 11$  (exponentul de cifrare).

Răspuns:  $M = 5$ ,  $d = 35$ .

**Exercițiul 14.3.11** Să se cifreze mesajul  $M = 4$ , utilizând sistemul RSA cu următorii parametrii:  $N = 209$  (modulul de cifrare),  $e = 11$  (exponentul de cifrare).

Rezolvare: Criptograma este:  $C = M^e = 4^{11} = 92 \bmod 209$ .

**Exercițiul 14.3.12** Să se descifreze mesajul  $C = 92$ , utilizând sistemul RSA cu următorii parametrii:  $N = 209 = 11 \cdot 19$  (modulul de cifrare),  $e = 11$  (exponentul de cifrare).

Rezolvare: Deoarece se cunoaște factorizarea  $N = 11 \cdot 19$ , se poate calcula  $\varphi(N) = 18 \cdot 10 = 180$ ,  $d = 131$ ,  $M = 4$ .

# Capitolul 15

## Sistemul de cifrare ElGamal

### 15.1 Breviar teoretic

Algoritmul de cifrare ElGamal este definit de un număr prim  $p$  și un element  $g \in Z_p^*$  primitiv, numit generator. Pentru cheia privată  $x \in Z_p^*$  se calculează  $y = g^x \bmod p$ , cheia publică fiind tripletul  $(y, g, p)$ .

Pentru a cifra un mesaj  $M \in Z_p$  se alege aleatoriu  $k \in Z_{p-1}$ , textul cifrat fiind  $(y_1, y_2) = (g^k \bmod p, My^k \bmod p)$ .

Pentru a descifra mesajul  $(y_1, y_2)$  se calculează  $y_2(y_1^x)^{-1} \bmod p$ .

### 15.2 Exerciții rezolvate

**Exercițiul 15.2.1** *Să se cifreze mesajul  $M = 4$  cu ajutorul algoritmului ElGamal cu parametrii  $p = 17$ ,  $g = 14$ ,  $x = 2$ .*

*Rezolvare:* Cheia publică este  $(y, g, p) = (14^2 \bmod 17, 14, 17) = (9, 14, 17)$ , cheia privată  $x = 2$ . Alegem, spre exemplu,  $k = 7$  relativ prim cu  $16 = p - 1$ . Obținem mesajul cifrat  $C = (14^7 \bmod 17, 4 \cdot 9^7 \bmod 17) = \{6, 8\}$ .

**Exercițiul 15.2.2** *Să se descifreze mesajul  $\{6, 8\}$ , știind că a fost cifrat cu ajutorul algoritmului ElGamal cu parametrii  $p = 17$ ,  $g = 14$ ,  $x = 2$ .*

*Rezolvare:* Cheia publică este  $\{y, g, p\} = \{9, 14, 17\}$ , cheia privată  $x = 2$ . Mesajul clar se obține aplicând formula  $y_2 y_1^{-x} \bmod p = 4$ .

### 15.3 Exerciții propuse

**Exercițiul 15.3.1** *Să se cifreze mesajul 5 cu ajutorul algoritmului ElGamal cu parametrii  $p = 23$ ,  $g = 14$ ,  $x = 2$ . Valoarea  $k$  utilizată pentru cifrare este 7.*

*Răspuns:* Mesajul cifrat este (19, 11).

**Exercițiul 15.3.2** *Să se cifreze mesajul 5 cu ajutorul algoritmului ElGamal cu parametrii  $p = 23$ ,  $g = 14$ ,  $x = 2$ . Valoarea  $k$  utilizată pentru cifrare este 9.*

*Răspuns:* Mesajul cifrat este (21, 20).

**Exercițiul 15.3.3** *Să se cifreze mesajul 3 cu ajutorul algoritmului ElGamal cu parametrii  $p = 47$ ,  $g = 14$ ,  $x = 3$ . Valoarea  $k$  utilizată pentru cifrare este 5.*

*Răspuns:* Mesajul cifrat este (3, 34).

**Exercițiul 15.3.4** *Să se cifreze mesajul 8 cu ajutorul algoritmului ElGamal cu parametrii  $p = 47$ ,  $g = 4$ ,  $x = 2$ . Valoarea  $k$  utilizată pentru cifrare este 3.*

*Răspuns:* Mesajul cifrat este (17, 9).

**Exercițiul 15.3.5** *Să se cifreze mesajul 4 cu ajutorul algoritmului ElGamal cu parametrii  $p = 23$ ,  $g = 7$ ,  $x = 3$ . Valoarea  $k$  utilizată pentru cifrare este 3.*

*Răspuns:* Mesajul cifrat este (21, 14).

**Exercițiul 15.3.6** *Să se descifreze mesajul (17, 9) cu ajutorul algoritmului ElGamal cu parametrii  $p = 47$ ,  $g = 4$ ,  $x = 2$ .*

*Răspuns:* Mesajul clar este 8.

**Exercițiul 15.3.7** *Să se descifreze mesajul (3, 34) cu ajutorul algoritmului ElGamal cu parametrii  $p = 47$ ,  $g = 14$ ,  $x = 3$ .*

*Răspuns:* Mesajul clar este 3.

**Exercițiul 15.3.8** *Să se descifreze mesajul (21, 14) cu ajutorul algoritmului ElGamal cu parametrii  $p = 23$ ,  $g = 7$ ,  $x = 3$ .*

*Răspuns:* Mesajul clar este 4.

# Capitolul 16

## Aritmetica pe curbe eliptice

### 16.1 Breviar teoretic

**Definiția 16.1** O curbă eliptică  $E$  este constituită din elemente (numite puncte) de tipul  $(x, y)$  ce satisfac ecuația:

$$y^2 \equiv x^3 + ax + b \pmod{p}$$

unde  $a$  și  $b$  sunt constante astfel încât  $4a^3 + 27b^2 \not\equiv 0 \pmod{p}$  și  $p$  este un număr prim, împreună cu un element singular, notat  $\mathcal{O}$  și numit punctul de la infinit. Acest punct poate fi privit ca fiind punctul din vârful și de la baza oricărei linii verticale.

O curbă eliptică  $E$  are o structură de grup abelian împreună cu operația adunare. Adunarea a două puncte de pe o curbă eliptică este definită în concordanță cu o mulțime simplă de reguli (vezi figura 16.1).

Fiind date două puncte pe  $E$ ,  $P_1(x_1, y_1)$  și  $P_2(x_2, y_2)$ , avem următoarele cazuri:

- dacă  $x_2 = x_1$  și  $y_2 = -y_1$  atunci  $P_1 + P_2 = \mathcal{O}$ .
- altfel  $P_1 + P_2 = (x_3, y_3)$ , unde:

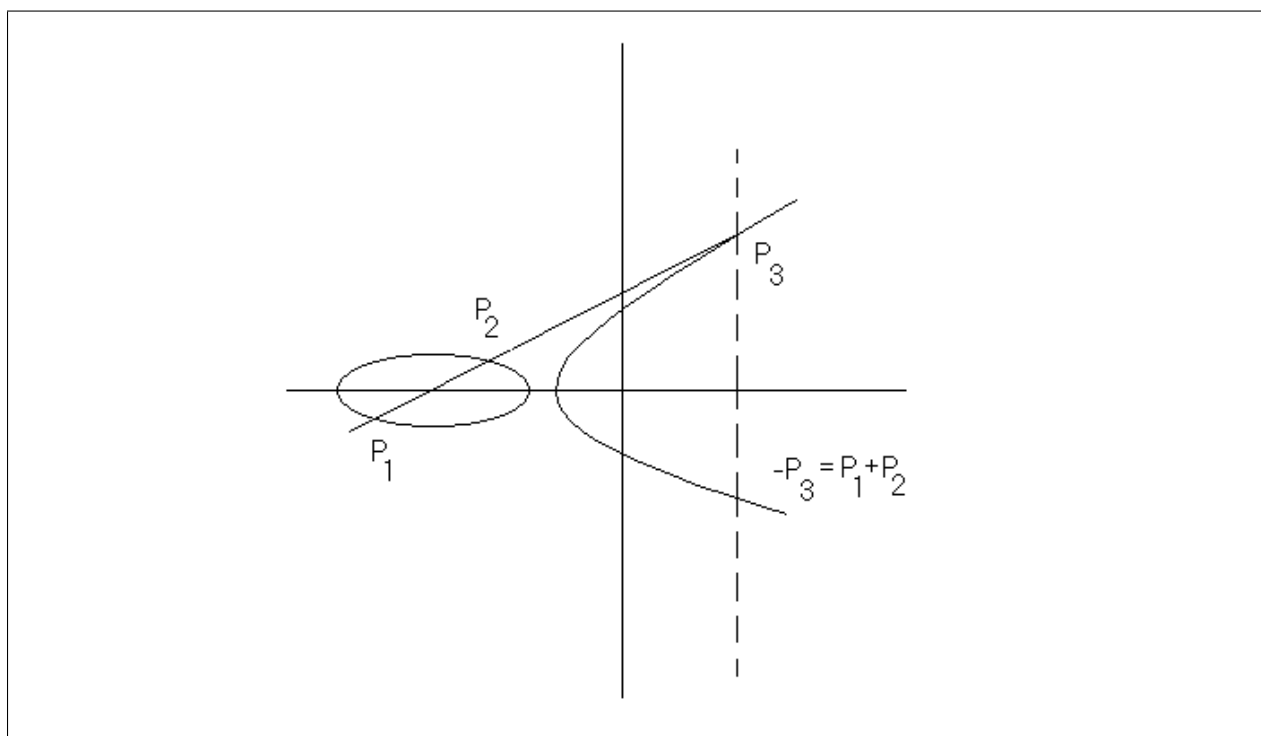
$$\begin{cases} x_3 = \lambda^2 - x_1 - x_2 \\ y_3 = \lambda(x_1 - x_3) - y_1 \end{cases}$$

cu

$$\lambda = \begin{cases} \frac{y_2 - y_1}{x_2 - x_1}, & \text{dacă } P_1 \neq P_2 \\ \frac{3x_1^2 + a}{2y_1}, & \text{dacă } P_1 = P_2. \end{cases}$$

**Observația 16.1** A nu se confunda punctul la infinit  $\mathcal{O}$  cu perechea  $(0, 0)$ . Punctul la infinit aparține tuturor curbelor eliptice, în timp ce punctul  $(0, 0)$  este un element doar pentru curbele eliptice cu parametrul  $b = 0$ .





**Figura 16.1:** Operația de adunare pe o curbă eliptică.

## 16.2 Exerciții rezolvate

**Exercițiul 16.2.1** Fie curba eliptică  $y^2 = x^3 + 7x + 4$  definită peste  $F_{71}$ . Să se adune punctele  $P(15, 17)$  și  $Q(43, 24)$ .

*Rezolvare:*

Coordoantele punctului  $P + Q = (x_3, y_3)$ , sunt date de formulele:

$$\begin{cases} x_3 = \lambda^2 - x_1 - x_2 \\ y_3 = \lambda(x_1 - x_3) - y_1 \end{cases}$$

unde  $\lambda = \frac{y_2 - y_1}{x_2 - x_1}$ .

Pentru calculul  $\lambda = 7 \cdot (28^{-1} \bmod 71)$ , se folosește algoritmul lui Euclid care găsește  $33 = 28^{-1} \bmod 71$ , deci  $\lambda = 231$ .

Atunci  $x_3 = 231^2 - 15 - 43 \bmod 71 = 53$  iar  $y_3 = 231(15 - 53) - 17 \bmod 71 = 9$ . În concluzie, coordoantele punctului care reprezintă suma celor două puncte de pe curba eliptică dată sunt  $(53, 9)$ .

**Exercițiul 16.2.2** Fie curba eliptică  $y^2 = x^3 + x + 3$  definită peste  $F_{17}$ . Arătați că punctul  $(2, 8)$  este un generator al punctelor de pe curba eliptică.

*Rezolvare:* Succesiv putem scrie  $1P = (2, 8)$ ,  $2P = (12, 3)$ ,  $3P = (16, 16)$ ,  $4P = (8, 8)$ ,  $5P = (7, 9)$ ,  $6P = (6, 15)$ ,  $7P = (11, 6)$ ,  $8P = (3, 13)$ ,  $9P = (3, 4)$ ,  $10P = (11, 11)$ ,  $11P = (6, 2)$ ,  $12P = (7, 8)$ ,  $13P = (8, 9)$ ,  $14P = (16, 1)$ ,  $15P = (12, 14)$ ,  $16P = (2, 9)$ ,  $17P = O$ .

## 16.3 Exerciții propuse

**Exercițiul 16.3.1** Fie curba eliptică  $y^2 = x^3 + 2x + 3$  definită peste  $F_{23}$ . Să se adune punctele  $P(6, 1)$  și  $Q(13, 8)$ .

*Răspuns:*  $R(5, 0)$ .

**Exercițiul 16.3.2** Fie curba eliptică  $y^2 = x^3 + 7x + 4$  definită peste  $F_{71}$ . Se dă punctul  $P(15, 17)$ . Aflați  $2P$ .

*Răspuns:*  $(66, 25)$ .

**Exercițiul 16.3.3** Fie curba eliptică  $y^2 = x^3 + x + 6$  definită peste  $F_{11}$ . Să se arate că punctul  $(2, 7)$  este un generator al punctelor de pe curba eliptică.

*Răspuns:*  $1P = (2, 7)$ ,  $2P = (5, 2)$ ,  $3P = (8, 3)$ ,  $4P = (10, 2)$ ,  $5P = (3, 6)$ ,  $6P = (7, 9)$ ,  $7P = (7, 2)$ ,  $8P = (3, 5)$ ,  $9P = (10, 9)$ ,  $10P = (8, 8)$ ,  $11P = (5, 9)$ ,  $12P = (2, 4)$ ,  $13P = O$ .

**Exercițiul 16.3.4** Fie curba eliptică  $y^2 = x^3 + 6x + 11$  definită peste  $F_{17}$ . Se dă punctul  $P(6, 5)$ . Aflați  $2P$ .

*Răspuns:*  $(1, 1)$ .

**Exercițiul 16.3.5** Fie curba eliptică  $y^2 = x^3 + x + 3$  definită peste  $F_7$ . Arătați că punctul  $(4, 6)$  este un generator al punctelor de pe curba eliptică.

*Răspuns:* Succesiv obținem  $1P = (4, 6)$ ,  $2P = (6, 1)$ ,  $3P = (5, 0)$ ,  $4P = (6, 6)$ ,  $5P = (4, 1)$ ,  $6P = O$ .

**Exercițiul 16.3.6** Fie curba eliptică  $y^2 = x^3 + 9$  definită peste  $F_{37}$ . Se dă punctul  $P(6, 22)$ . Aflați  $2P$ .

*Răspuns:*  $(35, 1)$ .

**Exercițiul 16.3.7** Fie curba eliptică  $y^2 = x^3 + 9$  definită peste  $F_{37}$ . Se dau punctele  $P(6, 22)$  și  $Q(8, 15)$ . Aflați  $P + Q$ .

*Răspuns:*  $(26, 11)$ .

**Exercițiul 16.3.8** Fie curba eliptică  $y^2 = x^3 + 11x + 20$  definită peste  $F_{23}$ . Se dau punctele  $P(7, 7)$  și  $Q(15, 15)$ . Aflați  $P + Q$ .

*Răspuns:*  $(2, 21)$ .

**Exercițiul 16.3.9** Fie curba eliptică  $y^2 = x^3 + 7x + 11$  definită peste  $F_{23}$ . Se dau punctele  $P(21, 14)$  și  $Q(7, 9)$ . Aflați  $P + Q$ .

*Răspuns:*  $(22, 7)$ .

**Exercițiul 16.3.10** Fie curba eliptică  $y^2 = x^3 + 5x + 5$  definită peste  $F_{17}$ . Se dă punctul  $P(3, 8)$ . Aflați  $3P$ .

*Răspuns:*  $(12, 5)$ .

**Exercițiul 16.3.11** Fie curba eliptică  $y^2 = x^3 + 3x$  definită peste  $F_{11}$ . Arătați că punctele  $P(0, 0)$  și  $Q(1, 2)$  aparțin curbei. Aflați  $P + Q$ .

*Răspuns:* Cele 2 puncte satisfac fiecare ecuația curbei eliptice. Suma lor este  $(3, 5)$ .

**Exercițiul 16.3.12** Fie curba eliptică  $y^2 = x^3 + 6x + 11$  definită peste  $F_{17}$ . Se dau punctele  $P(12, 3)$  și  $Q(6, 12)$ . Aflați  $P + Q$ .

*Răspuns:*  $(14, 0)$ .

# Capitolul 17

## Sistemul de cifrare ElGamal bazat pe curbe eliptice

### 17.1 Breviar teoretic

Algoritmul ElGamal poate fi extins pe orice grup finit  $(G, \circ)$ , în care problema logaritmului discret este dificilă, în particular și pe grupul punctelor de pe o curbă eliptică.

Astfel, fie  $\alpha \in G$  pentru care problema logaritmului în subgrupul  $H = \{\alpha^i | i \geq 0\}$  este dificilă. Pe baza cheii private  $x \in Z$ , se construiește  $\beta = \alpha^x$ , cheia publică fiind  $\{G, \alpha, \beta\}$ .

Pentru a cifra un mesaj  $M$  se alege aleatoriu  $k \in Z_{|H|}$  și se aplică regula de cifrare:  $E(M, k) = (\alpha^k, M \circ \beta^k)$ .

Mesajul clar  $m$  se recuperează din mesajul cifrat  $(y_1, y_2)$  după regula:  $y_2 \circ (y_1^x)^{-1}$ . Într-adevăr  $y_2 \circ (y_1^x)^{-1} = M \circ \beta^k \circ ((\alpha^k)^x)^{-1} = M \circ \alpha^{kx} \circ (\alpha^{kx})^{-1} = M$ .

### 17.2 Exerciții rezolvate

**Exercițiul 17.2.1** Să se cifreze mesajul  $(10, 9)$  utilizând curba eliptică (publică)  $E : y^2 = x^3 + x + 6$  pe  $\mathbf{Z}_{11}$  cu ajutorul algoritmului ElGamal.

*Rezolvare:* Pentru a calcula punctele curbei eliptice se calculează valorile  $z = x^3 + x + 6 \pmod{11}$ , se vede care din aceste valori sunt reziduri pătratică cu ajutorul teoremei lui Euler ( $z$  este reziduu pătratic dacă și numai dacă  $z^{\frac{p-1}{2}} \equiv 1 \pmod{p}$ ) și apoi se calculează rădăcinile pătrate ale acestor reziduri prin formula  $y = \pm z^{\frac{p+1}{2}} \pmod{p}$ . Punctele curbei eliptice vor fi:  $\{(2, 7), (2, 4), (3, 5), (3, 6), (5, 2), (5, 9), (7, 2), (7, 9), (8, 3), (8, 8), (10, 2), (10, 9), \mathcal{O}\}$ .

Grupul  $E$  este grup ciclic (numărul de elemente este al grupului este număr prim) și se ia ca generator pentru acesta elementul (public)  $\alpha = (2, 7)$ . Cheia privată de descifrare, notată prin  $d$ , este o valoare între 1 și numărul de puncte de pe o curbă eliptică  $-1$ . Cheia publică, notată prin  $\beta$ , se obține din  $\alpha$  și exponentul secret  $d$  prin formula  $\beta = d\alpha$ .

Operația de cifrare a mesajul  $M$  cu ajutorul cheii (secrete)  $k$  este:

$$E(M, k) = (k\alpha, M + k\beta).$$

Operația de descifrare pentru a obține  $M$  este:

$$D_k(y_1, y_2) = y_2 - dy_1.$$

Fie  $d = 3$ . Se determină  $\beta = 3(2, 7) = (8, 3)$ .

Considerând valoarea aleatoare  $k = 4$ , se obține:  $E(M, k) = (4(2, 7), (10, 9) + 4(8, 3)) = ((10, 2), (10, 9) + (2, 4)) = ((10, 2), (3, 5))$

**Exercițiul 17.2.2** *Să se descifreze mesajul  $((10, 2), (3, 5))$  știind că a fost cifrat cu algoritmul ElGamal utilizând curba eliptică(publică)  $E : y^2 = x^3 + x + 6$  pe  $\mathbf{Z}_{11}$  și cheia privată  $d = 3$ .*

*Rezolvare:* Se determină mesajul clar ca fiind:  $M = y_2 - dy_1 = (3, 5) - 3(10, 2) = (3, 5) - (2, 4) = (3, 5) + (2, 7) = (10, 9)$ .

## 17.3 Exerciții propuse

**Exercițiul 17.3.1** *Se consideră algoritmul ElGamal precizat de parametrul  $E : y^2 = x^3 + x + 6$  peste  $\mathbf{Z}_{11}$ . Arătați că  $\alpha = (2, 7)$  este un generator al grupului  $E$ . Se consideră cheia privată  $d = 5$ . Să se cifreze mesajul  $(10, 9)$  cu valoarea aleatoare  $k = 3$ .*

*Răspuns:* Valoarea cheii publice este  $\beta = d\alpha = (3, 6)$ . Mesajul cifrat este  $(k\alpha, M + k\beta) = ((8, 3), (10, 9) + (5, 2)) = ((8, 3), (5, 9))$ .

**Exercițiul 17.3.2** *Se consideră algoritmul ElGamal precizat de parametrul  $E : y^2 = x^3 + x + 6$  peste  $\mathbf{Z}_{11}$ . Arătați că  $\alpha = (2, 7)$  este un generator al grupului  $E$ . Să se descifreze mesajul  $((8, 3), (5, 9))$  cu ajutorul cheii private  $d = 5$ .*

*Răspuns:*  $D_k(y_1, y_2) = (y_2 - dy_1) = ((5, 9) - 5(8, 3)) = ((5, 9) - (5, 2)) = ((5, 9) + (5, 9)) = (10, 9)$ .

**Exercițiul 17.3.3** *Se consideră algoritmul ElGamal precizat de parametrul  $E : y^2 = x^3 + x + 6$  peste  $\mathbf{Z}_{13}$ . Arătați că  $\alpha = (4, 3)$  este un generator al grupului  $E$ . Se consideră cheia privată  $d = 3$ . Să se cifreze mesajul  $(3, 7)$  cu valoarea aleatoare  $k = 4$ .*

*Răspuns:* Valoarea cheii publice este  $\beta = d\alpha = (3, 7)$ . Mesajul cifrat este  $(k\alpha, M + k\beta) = ((9, 4), (3, 7) + (4, 10)) = ((9, 4), (2, 9))$ .

**Exercițiul 17.3.4** *Se consideră algoritmul ElGamal precizat de parametrul  $E : y^2 = x^3 + x + 6$  peste  $\mathbf{Z}_{13}$ . Arătați că  $\alpha = (4, 3)$  este un generator al grupului  $E$ . Să se descifreze mesajul  $((9, 4), (2, 9))$  cu ajutorul cheii private  $d = 3$ .*

*Răspuns:*  $D_k(y_1, y_2) = (y_2 - dy_1) = ((2, 9) - 3(9, 4)) = ((2, 9) - (4, 10)) = ((2, 9) + (4, 3)) = (3, 7)$ .

**Exercițiul 17.3.5** *Se consideră algoritmul ElGamal precizat de parametrii  $E : y^2 = x^3 + x + 6$  peste  $\mathbf{Z}_{17}$ . Se alege generatorul subgrupului ciclic  $\alpha = (1, 5)$  al lui  $E$ . Se consideră cheia privată  $d = 7$ . Să se cifreze mesajul  $(8, 4)$  utilizând valoarea aleatoare  $k = 3$ .*

*Răspuns:* Valoarea cheii publice este  $\beta = d\alpha = (7, 13)$ . Mesajul cifrat este  $(k\alpha, M + k\beta) = ((7, 4), (8, 4) + (1, 5)) = ((7, 4), (16, 2))$ .

**Exercițiul 17.3.6** *Se consideră algoritmul ElGamal precizat de parametrii  $E : y^2 = x^3 + x + 6$  peste  $\mathbf{Z}_{17}$ . Să se descifreze mesajul  $((7, 4), (16, 2))$  cu ajutorul cheii private  $d = 3$ .*

*Răspuns:*  $D_k(y_1, y_2) = (y_2 - dy_1) = ((16, 2) - 7(7, 4)) = ((16, 2) - (1, 5)) = ((2, 9) + (1, 12)) = (8, 4)$ .

**Exercițiul 17.3.7** *Se consideră algoritmul ElGamal precizat de parametrii  $E : y^2 = x^3 + 2x + 6$  peste  $\mathbf{Z}_{29}$ . Se alege generatorul subgrupului ciclic  $\alpha = (20, 10)$  al lui  $E$ . Se consideră cheia privată  $d = 5$ . Să se cifreze mesajul  $(10, 9)$  utilizând valoarea aleatoare  $k = 6$ .*

*Răspuns:* Valoarea cheii publice este  $\beta = d\alpha = (1, 3)$ . Mesajul cifrat este  $(k\alpha, M + k\beta) = ((14, 9), (10, 9) + (14, 9)) = ((14, 9), (5, 20))$ .

**Exercițiul 17.3.8** *Se consideră algoritmul ElGamal precizat de parametrii  $E : y^2 = x^3 + 2x + 6$  peste  $\mathbf{Z}_{29}$ . Să se descifreze mesajul  $((14, 9), (5, 20))$  cu ajutorul cheii private  $d = 5$ .*

*Răspuns:*  $D_k(y_1, y_2) = (y_2 - dy_1) = ((5, 20) - 5(14, 9)) = ((5, 20) - (14, 9)) = ((5, 20) + (14, 20)) = (10, 9)$ .

**Exercițiul 17.3.9** *Se consideră algoritmul ElGamal precizat de parametrii  $E : y^2 = x^3 + 2x + 7$  peste  $\mathbf{Z}_{17}$ . Să se descifreze mesajul  $((16, 2), (2, 6))$  cu ajutorul cheii private  $d = 5$ .*

*Răspuns:*  $D_k(y_1, y_2) = (y_2 - dy_1) = ((2, 6) - 5(16, 2)) = ((2, 6) - (16, 2)) = ((2, 6) + (16, 15)) = (8, 12)$ .



# Capitolul 18

## Sistemul de cifrare Menezes-Vanstone

### 18.1 Breviar teoretic

În acest sistem de cifrare - de fapt o variantă a lui ElGamal - curba eliptică este utilizată pentru mascare, textele clare și cele cifrate putând fi formate din orice elemente nenule (nu neapărat puncte din  $E$ ).

Fie  $E$  o curbă eliptică peste  $Z_p$ ,  $p > 3$  număr prim care conține un subgrup ciclic  $G$  în care problema logaritmului discret este dificilă. Pe baza cheii private  $d \in Z$ , se construiește  $\beta = d\alpha$ , cheia publică fiind  $\{E, \alpha, \beta\}$ .

Pentru a cifra mesajul  $m = (m_1, m_2) \in Z_p^* \times Z_p^*$  se alege aleatoriu  $k$  și se construiește textul cifrat  $(y_0, y_1, y_2)$  după regulile:

$$y_0 = k\alpha, (c_1, c_2) = k\beta, y_i = c_i m_i, i = 1, 2.$$

La descifrare, cunoscând  $(y_0, y_1, y_2)$  și cheia privată  $d$  se determină textul clar astfel:

$$(m_1, m_2) = (y_1 c_1^{-1} \bmod p, y_2 c_2^{-1} \bmod p), \text{ unde } dy_0 = (c_1, c_2)$$

### 18.2 Exerciții rezolvate

**Exercițiul 18.2.1** Se consideră algoritmul Menezes-Vanstone precizat de parametrii  $E : y^2 = x^3 + x + 6$  peste  $Z_{13}$ . Arătați că  $\alpha = (4, 3)$  este un generator al grupului  $E$ . Se consideră cheia privată  $d = 3$ . Să se cifreze mesajul  $(3, 7)$  cu valoarea aleatoare  $k = 4$ .

*Rezolvare:* Curba eliptică are 13 puncte deci grupul  $E$  este ciclic și orice element este generator.

Se calculează  $\beta = 3\alpha = 3 \cdot (4, 3) = (3, 7)$

Cifrarea mesajului  $(3, 7)$  cu valoarea aleatoare  $k = 4$  se face după următoarea formulă  $e_k(x, k) = (y_0, y_1, y_2)$  unde  $y_0 = k \cdot \alpha, (c_1, c_2) = k \cdot \beta, y_i = c_i \cdot x_i \pmod{p}$  pentru  $i = 1, 2$ .

Calculăm  $y_0 = 4 \cdot (4, 3) = (9, 4)$  iar  $(c_1, c_2) = 4 \cdot \beta = 12\alpha = (4, 10)$  deci  $c_1 = 4$  iar  $c_2 = 10$

Se calculează și  $y_1 = 4 \cdot 3 \bmod 13 = 12$  și  $y_2 = 10 \cdot 7 \bmod 13 = 5$ . Rezultatul cifrării mesajului  $(3, 7)$  cu valoarea aleatoare  $k = 4$  este  $((9, 4), 12, 5)$ .



### 18.3 Exerciții propuse

**Exercițiul 18.3.1** Se consideră algoritmul Menezes-Vanstone precizat de parametrii  $E : y^2 = x^3 + x + 6$  peste  $\mathbf{Z}_{13}$ . Arătați că  $\alpha = (4, 3)$  este un generator al grupului  $E$ . Se consideră cheia privată  $d = 3$ . Să se cifreze mesajul  $(1, 1)$  cu valoarea aleatoare  $k = 2$ .

Răspuns:  $\beta = (3, 7)$ ,  $(y_0, y_1, y_2) = ((2, 9), 11, 3)$ .

**Exercițiul 18.3.2** Se consideră curba eliptică  $E : y^2 = x^3 + x + 6$  peste  $\mathbf{Z}_{13}$ . Câte puncte are această curbă? Gasiți un generator al punctelor de pe curba eliptică. Câte elemente se pot cifra prin algoritmul ElGamal? Dar cu ajutorul algoritmului Menezes-Vanstone?

Răspuns: Curba are 13 puncte. Cum numărul de puncte este prim, grupul  $E$  este ciclic și deci orice punct din  $E$  este generator. Folosind sistemul ElGamal se pot cifra numai punctele de pe curbă, deci 13. Cu Menezes-Vanstone se poate cifra orice punct din  $\mathbf{Z}_{13} \times \mathbf{Z}_{13}$ .

**Exercițiul 18.3.3** Se consideră algoritmul Menezes-Vanstone precizat de parametrii  $E : y^2 = x^3 + 2x + 5$  peste  $\mathbf{Z}_{11}$ . Cunoscând cheia publică  $(\alpha, \beta) = ((3, 7), (4, 0))$ , să se cifreze mesajul  $(5, 2)$  cu valoarea aleatoare  $k = 7$ .

Răspuns:  $(y_0, y_1, y_2) = ((0, 7), 9, 0)$ .

**Exercițiul 18.3.4** Se consideră algoritmul Menezes-Vanstone precizat de parametrii  $E : y^2 = x^3 + 7x + 3$  peste  $\mathbf{Z}_{17}$ . Cunoscând cheia publică  $(\alpha, \beta) = ((12, 8), (7, 2))$ , să se cifreze mesajul  $(14, 7)$  cu valoarea aleatoare  $k = 7$ .

Răspuns:  $(y_0, y_1, y_2) = ((12, 9), 13, 3)$ .

**Exercițiul 18.3.5** Se consideră algoritmul Menezes-Vanstone precizat de parametrii  $E : y^2 = x^3 + 3x + 2$  peste  $\mathbf{Z}_{23}$ . Cunoscând cheia publică  $(\alpha, \beta) = ((15, 15), (15, 8))$ , să se cifreze mesajul  $(13, 19)$  cu valoarea aleatoare  $k = 2$ .

Răspuns:  $(y_0, y_1, y_2) = ((18, 0), 4, 0)$ .

**Exercițiul 18.3.6** Se consideră algoritmul Menezes-Vanstone precizat de parametrii  $E : y^2 = x^3 + 2x + 7$  peste  $\mathbf{Z}_{23}$ . Cunoscând cheia publică  $(\alpha, \beta) = ((5, 21), (16, 15))$ , să se cifreze mesajul  $(8, 10)$  cu valoarea aleatoare  $k = 4$ .

Răspuns:  $(y_0, y_1, y_2) = ((21, 8), 13, 12)$ .

**Exercițiul 18.3.7** Se consideră algoritmul Menezes-Vanstone precizat de parametrii  $E : y^2 = x^3 + 2x + 7$  peste  $\mathbf{Z}_{23}$ . Cunoscând cheia publică  $(\alpha, \beta) = ((5, 21), (16, 15))$ , să se cifreze mesajul  $(19, 2)$  cu valoarea aleatoare  $k = 5$ .

Răspuns:  $(y_0, y_1, y_2) = ((15, 13), 5, 16)$ .

**Exercițiul 18.3.8** *Se consideră algoritmul Menezes-Vanstone precizat de parametrii  $E : y^2 = x^3 + 2x + 5$  peste  $\mathbf{Z}_{17}$ . Cunoșcând cheia privată  $d = 3$ , să se descifreze mesajul  $(y_0, y_1, y_2) = ((1, 12), 2, 10)$ .*

*Răspuns:*  $(m_1, m_2) = (12, 5)$ .

**Exercițiul 18.3.9** *Se consideră algoritmul Menezes-Vanstone precizat de parametrii  $E : y^2 = x^3 + 5x + 4$  peste  $\mathbf{Z}_{19}$ . Cunoșcând cheia privată  $d = 2$ , să se descifreze mesajul  $(y_0, y_1, y_2) = ((17, 9), 12, 14)$ .*

*Răspuns:*  $(m_1, m_2) = (11, 11)$ .

**Exercițiul 18.3.10** *Se consideră algoritmul Menezes-Vanstone precizat de parametrii  $E : y^2 = x^3 + 2x + 7$  peste  $\mathbf{Z}_{23}$ . Cunoșcând cheia privată  $d = 7$ , să se descifreze mesajul  $(y_0, y_1, y_2) = ((21, 8), 8, 4)$ .*

*Răspuns:*  $(m_1, m_2) = (12, 11)$ .



# Capitolul 19

## Funcții de dispersie

### 19.1 Breviar teoretic

Problematica funcțiilor hash fiind deosebit de vastă, în cele ce urmează ne vom opri numai asupra aspectelor strict necesare înțelegerii utilizării acestor funcții în cadrul algoritmilor de semnătură digitală.

**Definiția 19.1** *O funcție  $f$  se numește funcție unidirecțională dacă:*

- a) fiind dat  $x$ , este ușor de calculat  $f(x)$ ;*
- b) fiind dat  $f(x)$ , este greu de calculat  $x$ .*

**Definiția 19.2** *O funcție  $f$  se numește funcție unidirecțională cu trapă (trap-door) dacă:*

- a) fiind dat  $x$ , este ușor de calculat  $f(x)$ ;*
- b) fiind dat  $f(x)$ , este greu de calculat  $x$ ;*
- c) pe baza unei informații secrete  $y$ , este ușor de calculat  $x$  din  $f(x)$ .*

**Definiția 19.3** *Funcția hash este o funcție care se aplică unui șir de lungime oarecare obținându-se un șir de lungime fixată (de obicei, mai mică decât lungimea șirului de intrare).*

**Definiția 19.4** *O funcție  $H$  se numește funcție hash unidirecțională dacă:*

- a)  $H$  este funcție hash;*
- b)  $H$  este funcție unidirecțională.*

Pentru a putea fi folosite pentru semnături digitale, funcțiile hash unidirecționale trebuie să mai îndeplinească, printre altele una din următoarele două condiții:

- 1) oricare ar fi  $M$  dat, este greu de găsit  $M'$  astfel încât  $H(M') = H(M)$ ;
- 2) este greu de găsit o pereche oarecare  $M, M'$  astfel încât  $H(M) = H(M')$ .

Funcțiile hash unidirecționale care îndeplinesc condiția (1) se numesc funcții hash unidirecționale slabe (sau universale), iar cele care îndeplinesc condiția (2) se numesc funcții hash unidirecționale tari (sau fără coliziuni).

Prima condiție este ușor de justificat: dacă  $A$  a semnat mesajul  $M$  cu  $H(M)$ , iar  $B$  obține  $M'$  astfel încât  $H(M') = H(M)$ , atunci  $B$  ar putea pretinde că  $A$  ar fi semnat mesajul  $M'$ .

A doua condiție este justificată de existența atacului *birthday*, metodă generală de atac aplicabilă oricărei funcții hash, atac inspirat de paradoxul matematic al zilei de naștere.

Datorită atacului birthday, pentru o funcție hash care are la ieșire un șir cu o lungime de  $m$  biți ( $2^m$  posibilități) se pot găsi coliziuni generând doar  $2^{m/2}$  perechi de mesaje-valori hash.

În aceste condiții, algoritmi hash care produc valori hash de 64 biți se consideră nesiguri deoarece, cu tehnologia actuală, se pot genera  $2^{64/2} = 2^{32}$  mesaje și deci este posibilă găsirea de mesaje care să intre în coliziune. De aceea se recomandă ca valoarea hash să fie de lungime de cel puțin 128 biți.

În cele ce urmează vom descrie funcția de dispersie Chaum -van Heijl-Pfitzmann. Fie  $p$  un număr prim mare astfel ca  $q = \frac{p-1}{2}$  să fie de asemenea prim. Considerăm  $\alpha, \beta \in \mathbf{Z}_p$  elemente primitive. Calculul valorii logaritmului discret  $\log_\alpha \beta$  este dificil din punct de vedere computațional. Vom defini funcția de dispersie *Chaum -van Heijl-Pfitzmann*  $h : \mathbf{Z}_q \times \mathbf{Z}_q \rightarrow \mathbf{Z}_p^*$  prin

$$h(x_1, x_2) = \alpha^{x_1} \beta^{x_2} \bmod p.$$

Dacă există o coliziune pentru funcția Chaum -van Heijl-Pfitzmann atunci calculul logaritmului discret  $\log_\alpha \beta$  este ușor.

Să vedem cum anume se poate determina valoarea logaritmului discret  $\log_\alpha \beta$ . Să presupunem că avem coliziunea  $h(x_1, x_2) = h(x_3, x_4)$  cu  $(x_1, x_2) \neq (x_3, x_4)$ . Deci  $\alpha^{x_1} \beta^{x_2} = \alpha^{x_3} \beta^{x_4} \bmod p$  sau echivalent  $\alpha^{x_1-x_3} = \beta^{x_4-x_2} \bmod p$ . Fie  $d = (x_4 - x_2, p-1)$ . Deoarece  $p-1 = 2q$  iar  $q$  este număr prim avem  $d \in \{1, 2, q, p-1\}$ .

**Cazul**  $d = 1$ . Deoarece  $(x_4 - x_2, p-1) = 1$  există  $y = (x_4 - x_2)^{-1} \bmod p$ . Deci:

$$\beta = \beta^{(x_4-x_2)y} \bmod p = \alpha^{(x_1-x_3)y} \bmod p.$$

Deci  $\log_\alpha \beta = (x_1 - x_3)(x_4 - x_2)^{-1} \bmod (p-1)$ .

**Cazul**  $d = 2$ . Deoarece  $p-1 = 2q$ ,  $q$  număr prim, rezultă  $(x_4 - x_2, q) = 1$ . Fie  $y = (x_4 - x_2)^{-1} \bmod q$ . Deci, există  $k$  număr întreg astfel încât  $(x_4 - x_2)y = kq + 1$ . Deoarece  $\beta^q = -1 \bmod p$ , rezultă:

$$\beta^{(x_4-x_2)y} = \beta^{(kq+1)} = (-1)^k \beta \bmod p = \pm \beta \bmod p.$$

Acest lucru conduce la:

$$\alpha^{(x_1-x_3)y} = \beta^{(x_4-x_2)y} \bmod p = \pm \beta \bmod p.$$

Suntem în una din următoarele două situații:

$$\log_\alpha \beta = (x_1 - x_3)(x_4 - x_2)^{-1} \bmod (p-1),$$

$$\log_\alpha \beta = (x_1 - x_3)(x_4 - x_2)^{-1} + q \bmod (p-1),$$

Se verifică direct care dintre rezultate este cel corect.

**Cazul**  $d = q$ . Deoarece  $0 \leq x_2 \leq q - 1$  și  $0 \leq x_4 \leq q - 1$  rezultă faptul că  $-(q - 1) \leq x_4 - x_2 \leq q - 1$ . Acest lucru arată faptul că este imposibil să avem  $(x_4 - x_2, p - 1) = q$ .

**Cazul**  $d = p - 1$ . Acest caz este posibil numai dacă  $x_4 = x_2$ , rezultă  $x_1 = x_3$ . S-a ajuns la  $(x_1, x_2) = (x_3, x_4)$ , ceea ce contrazice ipoteza.

## 19.2 Exerciții propuse

**Exercițiul 19.2.1** Fie  $f : \mathbf{Z}_2^n \rightarrow \mathbf{Z}_2^n$  o funcție hash pentru care problema **CSP**<sup>1</sup> este satisfăcută. Definim funcția  $g : \mathbf{Z}_2^{2n} \rightarrow \mathbf{Z}_2^n$  prin  $g(x_1 || x_2) = f(x_1 \oplus x_2)$ . Arătați că  $g$  nu satisface problema **CSP**.

**Exercițiul 19.2.2** Fie  $p = 12347$ ,  $\alpha = 2$ ,  $\beta = 8461$  parametrii pentru funcția de dispersie Chaum - van Heijst - Pfitzmann. Fiind dată coliziunea  $\alpha^{5692} \beta^{144} = \alpha^{212} \beta^{4214} \pmod{p}$ , să se calculeze  $\log_\alpha \beta$ .

Răspuns:  $\log_\alpha \beta = 5689$ .

**Exercițiul 19.2.3** Fie  $p = 15083$ ,  $\alpha = 154$ ,  $\beta = 2307$  parametrii pentru funcția de dispersie Chaum - van Heijst - Pfitzmann. Fiind dată coliziunea  $\alpha^{7431} \beta^{5564} = \alpha^{1459} \beta^{954} \pmod{p}$ , să se calculeze  $\log_\alpha \beta$ .

---

<sup>1</sup>Fiind dată o pereche validă  $(x, y)$  este dificil de aflat  $x_1 \neq x$  astfel încât  $f(x_1) = f(x)$ .



# Capitolul 20

## Semnătura ElGamal

### 20.1 Breviar teoretic

Fie  $p$  un număr prim pentru care problema logaritmului discret în  $Z_p$  este dificilă și  $\alpha \in Z_p^*$  un element primitiv. Cheia publică  $\beta$  se construiește din cheia privată  $a$ :  $\beta = \alpha^a \bmod p$ .

Semnătura mesajului  $x$ , calculată cu ajutorul valorii aleatoare (secrete)  $k \in Z_{p-1}$ , este definită ca fiind  $(\gamma, \delta)$  unde:

$$\gamma = \alpha^k \bmod p \text{ și } \delta = (H(x) - a\gamma)k^{-1} \bmod (p-1),$$

$H(\cdot)$  fiind o funcție hash ( $H(x) = x$  dacă nu este specificată funcția hash).

Semnătura  $(\gamma, \delta)$  a mesajului  $x$  este verificată dacă are loc:

$$\beta^\gamma \gamma^\delta = \alpha^{H(x)} \bmod p.$$

### 20.2 Exerciții rezolvate

**Exercițiul 20.2.1** Să se semneze mesajul  $x = 101$  cu ajutorul algoritmului ElGamal specificat de parametrii următori:  $p = 467$ ,  $\alpha = 2$ , cheia privată  $a = 127$ , alegând valoarea  $k = 213$ .

*Rezolvare:* Se calculează  $\beta = \alpha^a \bmod p = 2^{127} \bmod 467 = 132$

Semnătura mesajului  $x = 101$  cu  $k = 213$  (de remarcat faptul că  $(213, 466) = 1$  și  $213^{-1} \bmod 466 = 431$ ) este:

$$\gamma = \alpha^k \bmod p = 2^{213} \bmod 467 = 29 \text{ și } \delta = (101 - 127 \cdot 29) \cdot 431 \bmod 466 = 16.$$

### 20.3 Exerciții propuse

**Exercițiul 20.3.1** Să se semneze mesajul  $x = 100$  cu ajutorul algoritmului ElGamal specificat de parametrii următori:  $p = 163$ ,  $\alpha = 2$ , cheia privată  $a = 127$ , alegând valoarea  $k = 215$ .

*Răspuns:* Semnătura mesajului este  $(\gamma, \delta) = (52, 24)$ .



**Exercițiul 20.3.2** Să se semneze mesajul  $x = 102$  cu ajutorul algoritmului ElGamal specificat de parametrul următor:  $p = 467$ ,  $\alpha = 2$ , cheia privată  $a = 127$ , alegând valoarea  $k = 213$ .

*Răspuns:* Semnătura mesajului este  $(\gamma, \delta) = (29, 447)$ .

**Exercițiul 20.3.3** Să se semneze mesajul  $x = 57$  cu ajutorul algoritmului ElGamal specificat de parametrul următor:  $p = 97$ ,  $\alpha = 3$ , cheia privată  $a = 27$ , alegând valoarea  $k = 37$ .

*Răspuns:* Semnătura mesajului este  $(\gamma, \delta) = (66, 39)$ .

**Exercițiul 20.3.4** Să se semneze mesajul  $x = 29$  cu ajutorul algoritmului ElGamal specificat de parametrul următor:  $p = 127$ ,  $\alpha = 5$ , cheia privată  $a = 13$ , alegând valoarea  $k = 19$ .

*Răspuns:* Semnătura mesajului este  $(\gamma, \delta) = (66, 89)$ .

**Exercițiul 20.3.5** Să se semneze mesajul  $x = 78$  cu ajutorul algoritmului ElGamal specificat de parametrul următor:  $p = 131$ ,  $\alpha = 7$ , cheia privată  $a = 19$ , alegând valoarea  $k = 17$ .

*Răspuns:* Semnătura mesajului este  $(\gamma, \delta) = (3, 93)$ .

**Exercițiul 20.3.6** Mesajul  $x = 57$  a fost semnat cu ajutorul algoritmului ElGamal specificat de parametrul următor:  $p = 97$ ,  $\alpha = 3$ ,  $\beta = 70$ , obținându-se semnătura  $(\gamma, \delta) = (66, 39)$ . Este aceasta o semnătură validă?

*Răspuns:* Semnătura este validă deoarece se satisface relația de verificare  $\beta^\gamma \gamma^\delta \bmod p = \alpha^x \bmod p = 89$ .

**Exercițiul 20.3.7** Mesajul  $x = 34$  a fost semnat cu ajutorul algoritmului ElGamal specificat de parametrul următor:  $p = 131$ ,  $\alpha = 7$ ,  $\beta = 16$ , obținându-se semnătura  $(\gamma, \delta) = (3, 110)$ . Este aceasta o semnătură validă?

*Răspuns:* Semnătura nu este validă deoarece nu se satisface relația de verificare  $\beta^\gamma \gamma^\delta \bmod p = 4$ ;  $\alpha^x \bmod p = 9$ .

**Exercițiul 20.3.8** Mesajul  $x = 78$  a fost semnat cu ajutorul algoritmului ElGamal specificat de parametrul următor:  $p = 131$ ,  $\alpha = 7$ ,  $\beta = 16$ , obținându-se semnătura  $(\gamma, \delta) = (3, 93)$ . Este aceasta o semnătură validă?

*Răspuns:* Semnătura este validă deoarece se satisface relația de verificare  $\beta^\gamma \gamma^\delta \bmod p = \alpha^x \bmod p = 61$ .

# Capitolul 21

## Semnătura DSA/ECDSA

### 21.1 Breviar teoretic

Fie  $p$  un număr prim de 512 biți și  $q$  un factor prim de 160 biți ai lui  $p - 1$  și  $\alpha \in Z_p^*$  o rădăcină primitivă de ordin  $q$  a unității.

Cheia publică  $\beta$  se construiește din cheia privată  $a$ :  $\beta = \alpha^a \bmod p$ . Semnătura mesajului  $x$ , calculată cu ajutorul valorii aleatoare (secrete)  $k \in Z_q^*$ , este definită ca fiind  $(\gamma, \delta)$  unde:

$$(\gamma, \delta) = ((\alpha^k \bmod p) \bmod q, (H(x) + a\gamma)k^{-1} \bmod q),$$

$H(\cdot)$  fiind o funcție hash ( $H(x) = x$  dacă nu este specificată funcția hash).

Semnătura  $(\gamma, \delta)$  a mesajului  $x$  este verificată dacă are loc următoarea egalitate, unde  $e_1 = H(x)\delta^{-1} \bmod q$  și  $e_2 = \gamma\delta^{-1} \bmod q$ :

$$(\alpha^{e_1}\beta^{e_2} \bmod p) \bmod q = \gamma.$$

O variantă a DSA-ului este reprezentată de extensia acesteia pe curbele eliptice (ECDSA). În această situație se lucrează pe curbă eliptică  $E$  peste  $Z_q$ . Elementele necesare algoritmului sunt:

$G(x_G, y_G)$  generatorul punctelor de pe curba eliptică;

$n$  numărul elementelor de pe curba eliptică (sau ordinul lui  $G$  dacă  $G$  nu este generator);

$L_n$  numărul de biți ai lui  $n$ ;

$d_A$  cheia privată,  $d_A \in [1, n]$ ;

$Q_A = d_A G$  cheia publică.

În contextul celor de mai sus, algoritmul ECDSA este următorul:

**PASUL 1.** Se calculează  $e = H(M)$ . Fie  $z$  cei mai semnificativi  $L_n$  biți ai lui  $e$ .

**PASUL 2.** Se alege valoarea aleatoare<sup>1</sup>  $k$  în intervalul  $[1, n - 1]$ .

**PASUL 3.**  $r = x_1 \bmod n$ , unde  $(x_1, y_1) = kG$ . Dacă  $r = 0$  atunci revenim la PASUL 2.

**PASUL 4.**  $s = k^{-1}(z + rd_A) \bmod n$ . Dacă  $r = 0$  atunci revenim la PASUL 2.

**PASUL 5.** Semnătura este  $(r, s)$ .

Verificarea semnăturii ECDSA  $(r, s)$  se realizează după următorul algoritm.

**PASUL 1.** Dacă  $r, s \notin [1, n]$  semnătura este invalidă.

---

<sup>1</sup>valoarea  $k$  se numește cheie efemeră.

**PASUL 2.** Fie  $e = H(M)$ ,  $z$  cei mai semnificativi  $L_n$  biți ai lui  $e$ .

**PASUL 3.** Se calculează:  $w = s^{-1} \bmod n$ .

**PASUL 4.** Se calculează:  $u_1 = zw \bmod n$  și  $u_2 = rw \bmod n$ .

**PASUL 5.** Fie  $(x_1, y_1) = u_1G + u_2Q_A$ .

**PASUL 6.** Semnătura este validă dacă și numai dacă  $r = x_1 \bmod n$ .

## 21.2 Exerciții rezolvate

**Exercițiul 21.2.1** Să se semneze mesajul  $x = 100$  cu ajutorul algoritmului DSA specificat de parametrii următori:  $p = 7879$ ,  $q = 101$ ,  $\alpha = 170$ , valoarea aleatoare utilizată  $k = 50$ , cheia secretă fiind  $a = 75$ . Verificați rezultatul obținut.

*Rezolvare:* Se calculează:

$$\gamma = (\alpha^k \bmod p) \bmod q = (170^{50} \bmod 7879) \bmod 101 = 2518 \bmod 101 = 94.$$

$$\delta = (x + a\gamma)k^{-1} \bmod q = (100 + 75 \cdot 94)50^{-1} \bmod 101 = 7150 \cdot 50^{-1} \bmod 101 = 7150 \cdot 99 \bmod 101 = 42.$$

S-a folosit  $50^{-1}(\bmod 101) = -2 \bmod 101 = 99$  (fiindcă  $101 = 50 \cdot 2 + 1$ ).

Verificare:

$$\beta = \alpha^a \bmod p = 170^{75} \bmod 7879 = 4567.$$

$$e_1 = x\delta^{-1} \bmod q = 100 \cdot 42^{-1} \bmod 101 = 100 \cdot 89 \bmod 101 = 12.$$

$$e_2 = \gamma\delta^{-1} \bmod q = 94 \cdot 42^{-1} \bmod 101 = 94 \cdot 89 \bmod 101 = 84.$$

Se obține:

$$(\alpha^{e_1}\beta^{e_2} \bmod p) \bmod q = (170^{12} \cdot 4567^{84} \bmod 7879) \bmod 101 = 2518 \bmod 101 = 94 = \gamma.$$

## 21.3 Exerciții propuse

**Exercițiul 21.3.1** Să se semneze mesajul  $x = 101$  cu ajutorul algoritmului DSA specificat de parametrii următori:  $p = 7879$ ,  $q = 101$ ,  $\alpha = 170$ , valoarea aleatoare utilizată  $k = 50$ , cheia secretă fiind  $a = 75$ . Verificați rezultatul obținut.

*Răspuns:* Semnătura mesajului este  $(\gamma, \delta) = (94, 40)$ . Cheia publică este  $\beta = 4567$ .

**Exercițiul 21.3.2** Să se semneze mesajul  $x = 102$  cu ajutorul algoritmului DSA specificat de parametrii următori:  $p = 7879$ ,  $q = 101$ ,  $\alpha = 170$ , valoarea aleatoare utilizată  $k = 50$ , cheia secretă fiind  $a = 75$ . Verificați rezultatul obținut.

*Răspuns:* Semnătura mesajului este  $(\gamma, \delta) = (94, 38)$ . Cheia publică este  $\beta = 4567$ .

**Exercițiul 21.3.3** Să se semneze mesajul  $x = 75$  cu ajutorul algoritmului DSA specificat de parametrii următori:  $p = 131$ ,  $q = 13$ ,  $\alpha = 7$ ,  $a = 3$ , valoarea aleatoare utilizată  $k = 11$ . Verificați rezultatul obținut.

*Răspuns:* Semnătura mesajului este  $(\gamma, \delta) = (10, 6)$ . Totuși, semnătura nu se verifică pentru ca  $\text{ord}(\alpha) = 65$  și nu  $q = 13$ . În concluzie, algoritmul DSA este setat impropriu.

**Exercițiul 21.3.4** Mesajul  $x = 502$  a fost semnat cu ajutorul algoritmului DSA specificat de parametrii următori:  $p = 617$ ,  $q = 11$ ,  $\alpha = 113$ ,  $\beta = 489$ , valoarea aleatoare utilizată  $k = 21$  și s-a obținut semnătura  $(\gamma, \delta) = (3, 10)$ . Este această semnătură validă?

*Răspuns:* Semnătura este validă deoarece se satisface relația de verificare  $(\alpha^{e_1} \beta^{e_2} \bmod p) \bmod q = \gamma = 3$ .

**Exercițiul 21.3.5** Mesajul  $x = 99$  a fost semnat cu ajutorul algoritmului DSA specificat de parametrii următori:  $p = 7879$ ,  $q = 101$ ,  $\alpha = 170$ ,  $\beta = 4567$ , valoarea aleatoare utilizată  $k = 50$  și s-a obținut semnătura  $(\gamma, \delta) = (94, 78)$ . Este această semnătură validă?

*Răspuns:* Semnătura nu este validă deoarece nu se satisface relația de verificare  $(\alpha^{e_1} \beta^{e_2} \bmod p) \bmod q \neq \gamma$ .

**Exercițiul 21.3.6** Mesajul  $x = 99$  a fost semnat cu ajutorul algoritmului DSA specificat de parametrii următori:  $p = 7879$ ,  $q = 101$ ,  $\alpha = 170$ ,  $\beta = 4567$ , valoarea aleatoare utilizată  $k = 50$  și s-a obținut semnătura  $(\gamma, \delta) = (94, 44)$ . Este această semnătură validă?

*Răspuns:* Semnătura este validă deoarece se satisface relația de verificare  $(\alpha^{e_1} \beta^{e_2} \bmod p) \bmod q = \gamma = 94$ .



# Capitolul 22

## Protocolul Diffie-Hellman de stabilire a cheilor

### 22.1 Breviar teoretic

Fie  $p$  un număr prim,  $q$  un divizor prim al lui  $p - 1$  și  $\alpha \in Z_p^*$ , element de ordin  $q$ . Protocolul Diffie-Hellman (DH), ce returnează o cheie comună de sesiune  $K$  este următorul:

**PASUL 1.**  $A$  generează aleator  $a \in Z_q^*$  și trimite lui  $B$  valoarea  $R_A = \alpha^a \pmod{p}$ .

**PASUL 2.**  $B$  generează aleator  $b \in Z_q^*$  și trimite lui  $A$  valoarea  $R_B = \alpha^b \pmod{p}$ .

**PASUL 3.**  $A$  calculează  $K = K_{A,B} = R_B^a = \alpha^{ab}$ .

**PASUL 4.**  $B$  calculează  $K = K_{B,A} = R_A^b = \alpha^{ab}$ .

### 22.2 Exerciții rezolvate

**Exercițiul 22.2.1** Să se specifice cheia rezultată în urma aplicării protocolului Diffie-Hellman specificat de parametrii:  $p = 25307$ ,  $\alpha = 2$ ,  $a = 2009$ ,  $b = 2010$ .

*Răspuns:*  $k = 21554$ .

*Rezolvare:*

**PASUL 1.**  $A$  trimite lui  $B$  valoarea  $R_A = \alpha^a \pmod{p} = 2^{2009} \pmod{25307} = 5755$ .

**PASUL 2.**  $B$  trimite lui  $A$  valoarea  $R_B = \alpha^b \pmod{p} = 2^{2010} \pmod{25307} = 11510$ .

**PASUL 3.**  $A$  calculează  $K = K_{A,B} = R_B^a = 11510^{2009} \pmod{25307} = 21554$ .

**PASUL 4.**  $B$  calculează  $K = K_{B,A} = R_A^b = 5755^{2010} \pmod{25307} = 21554$ .

### 22.3 Exerciții propuse

**Exercițiul 22.3.1** Să se specifice cheia rezultată în urma aplicării protocolului Diffie-Hellman specificat de parametrii:  $p = 25307$ ,  $\alpha = 2$ ,  $a = 3578$ ,  $b = 19956$ .

*Răspuns:  $k = 3694$ .*

**Exercițiul 22.3.2** *Să se specifice cheia rezultată în urma aplicării protocolului Diffie-Hellman specificat de parametrii:  $p = 25307$ ,  $\alpha = 2$ ,  $a = 1989$ ,  $b = 2009$ .*

*Răspuns:  $k = 12034$ .*

**Exercițiul 22.3.3** *Să se specifice cheia rezultată în urma aplicării protocolului Diffie-Hellman specificat de parametrii:  $p = 17$ ,  $\alpha = 7$ ,  $a = 9$ ,  $b = 3$ .*

*Răspuns:  $k = 14$ .*

**Exercițiul 22.3.4** *Să se specifice cheia rezultată în urma aplicării protocolului Diffie-Hellman specificat de parametrii:  $p = 10163$ ,  $\alpha = 652$ ,  $a = 6026$ ,  $b = 3510$ .*

*Răspuns:  $k = 7944$ .*

**Exercițiul 22.3.5** *Să se specifice cheia rezultată în urma aplicării protocolului Diffie-Hellman specificat de parametrii:  $p = 63299$ ,  $\alpha = 49297$ ,  $a = 5671$ ,  $b = 59073$ .*

*Răspuns:  $k = 57286$ .*

**Exercițiul 22.3.6** *Să se specifice cheia rezultată în urma aplicării protocolului Diffie-Hellman specificat de parametrii:  $p = 1319$ ,  $\alpha = 527$ ,  $a = 1088$ ,  $b = 584$ .*

*Răspuns:  $k = 352$ .*

**Exercițiul 22.3.7** *Să se specifice cheia rezultată în urma aplicării protocolului Diffie-Hellman specificat de parametrii:  $p = 2099$ ,  $\alpha = 1023$ ,  $a = 1496$ ,  $b = 648$ .*

*Răspuns:  $k = 612$ .*

**Exercițiul 22.3.8** *Să se specifice cheia rezultată în urma aplicării protocolului Diffie-Hellman specificat de parametrii:  $p = 1823$ ,  $\alpha = 776$ ,  $a = 1515$ ,  $b = 476$ .*

*Răspuns:  $k = 1555$ .*

**Exercițiul 22.3.9** *Să se specifice cheia rezultată în urma aplicării protocolului Diffie-Hellman specificat de parametrii:  $p = 2207$ ,  $\alpha = 371$ ,  $a = 839$ ,  $b = 1358$ .*

*Răspuns:  $k = 731$ .*

**Exercițiul 22.3.10** *În urma aplicării protocolului Diffie-Hellman, una dintre entitățile care doresc să genereze o cheie comună alege parametrul secret  $a = 1$  (sau  $b = 1$ ). Cum poate un atacator determina cheia în acest caz?*

# Capitolul 23

## Protocolul Blom

### 23.1 Breviar teoretic

Protocolul lui Blom asigură implementarea principiului compartimentării, între oricare doi participanți, dintr-o mulțime de  $n$  utilizatori. Protocolul se bazează pe existența unei autorități de încredere  $T$ . Fie  $n \geq 3$  numărul de utilizatori și  $p \geq n$  un număr prim. Cheia, ce urmează a fi calculată de oricare doi participanți este un element din  $\mathbf{Z}_p^*$ . Vom nota prin  $k$  numărul maxim de intruși<sup>1</sup> împotriva cărora poate fi asigurată protecția. Vom exemplifica protocolul pentru  $k = 1$ .

**PASUL 0.**  $T$  face public: numărul prim  $p$  și pentru fiecare utilizator  $A$  un număr aleator  $r_A \in \mathbf{Z}_p$ ,  $r_A \neq r_B$  pentru orice  $A \neq B$ .

**PASUL 1.**  $T$  generează aleatoriu trei numere  $a, b, c \in \mathbf{Z}_p$  și formează polinomul<sup>2</sup>:

$$f(X, Y) = a + b(X + Y) + cXY \bmod p.$$

**PASUL 2.** Pentru fiecare utilizator  $A$ ,  $T$  va construi polinomul:

$$g_A(X) = f(X, r_A) \bmod p,$$

pe care îl va transmite, cu asigurarea confidențialității, către  $A$ .

**PASUL 3.** Cheia stabilită de către  $A$  și  $B$  va fi:

$$K_{A,B} = K_{B,A} = f(r_A, r_B).$$

**Observația 23.1** Protocolul Blom, pentru  $k = 1$ , este necondiționat sigur împotriva oricărui atac individual. Cu alte cuvinte, orice alt participant  $C$  nu poate determina, din valorile publice  $r_A$  și  $r_B$ , cheia  $K_{A,B}$ . Acesta este utilizat în schema de protecție, utilizată de HDCP (High-bandwidth Digital Content Protection), în generarea cheilor dintre sursă și destinație (playere HD DVD sau televiziunea HD).

---

<sup>1</sup>numit și nivel de compartimentare.

<sup>2</sup>pentru  $k$  arbitrar polinomul utilizat în cadrul protocolului este  $f(X, Y) = \sum_{i,j=0}^k a_{i,j} X^i Y^j \bmod p$ ,  $a_{i,j} \in \mathbf{Z}_p$ ,  $a_{i,j} = a_{j,i}$  pentru orice  $i, j$ .



## 23.2 Exerciții rezolvate

**Exercițiul 23.2.1** *Specificați elementele de securitate pentru protocolul Blom, ce asigură compartimentarea între trei utilizatori  $A, B, C$ , caracterizat de  $p = 17$ ,  $k = 1$ , cheile publice ale acestora fiind  $r_A = 12$ ,  $r_B = 7$  și  $r_C = 1$ . Valorile alese de către  $T$  fiind  $a = 8$ ,  $b = 7$ ,  $c = 2$ .*

*Rezolvare:*  $T$  construiește polinomul:

$$f(X, Y) = 8 + 7(X + Y) + 2XY.$$

Polinoamele specifice fiecărui utilizator sunt:

$$g_A(X) = 7 + 14X, g_B(X) = 6 + 4X, g_C(X) = 15 + 9X.$$

Cheile de compartimentare (secrete) sunt:

$$K_{A,B} = 3, K_{A,C} = 4, K_{B,C} = 10.$$

$A$  poate calcula  $K_{AB}$  prin:

$$g_A(r_B) = 7 + 14 \cdot 7 \bmod 17 = 3.$$

$B$  poate calcula  $K_{BA}$  prin:

$$g_B(r_A) = 6 + 4 \cdot 12 \bmod 17 = 3.$$

## 23.3 Exerciții propuse

**Exercițiul 23.3.1** *Specificați cheile rezultate în urma protocolului Blom, ce asigură compartimentarea între trei utilizatori  $A, B, C$ , caracterizat de  $p = 29$ ,  $k = 1$ , cheile publice ale acestora fiind  $r_A = 1$ ,  $r_B = 2$  și  $r_C = 3$ . Valorile alese de către  $T$  fiind  $a = 13$ ,  $b = 11$ ,  $c = 17$ .*

Răspuns. Polinoamele secrete sunt  $g_A(X) = 324 + 28X$ ,  $g_B(X) = 6 + 16X$ ,  $g_C(X) = 17 + 4X$ . Cheile rezultate sunt  $K_{AB} = 22$ ,  $K_{AC} = 21$ ,  $K_{BC} = 25$ .

**Exercițiul 23.3.2** *Specificați cheile rezultate în urma protocolului Blom, ce asigură compartimentarea între trei utilizatori  $A, B, C$ , caracterizat de  $p = 29$ ,  $k = 1$ , cheile publice ale acestora fiind  $r_A = 13$ ,  $r_B = 11$  și  $r_C = 17$ . Valorile alese de către  $T$  fiind  $a = 1$ ,  $b = 2$ ,  $c = 3$ .*

Răspuns. Polinoamele secrete sunt  $g_A(X) = 27 + 12X$ ,  $g_B(X) = 23 + 6X$ ,  $g_C(X) = 6 + 24X$ . Cheile rezultate sunt  $K_{AB} = 14$ ,  $K_{AC} = 28$ ,  $K_{BC} = 9$ .

# Capitolul 24

## Protocolul Shamir de partajare a secretelor

### 24.1 Breviar teoretic

Schema lui Shamir își propune să partajeze cheia de cifrare  $S \in \mathcal{K} = Z_q$  la o mulțime de  $n$  participanți ( $q \geq n + 1$ ) astfel încât pentru reconstrucția cheii să fie nevoie de cooperarea a cel puțin  $k$  dintre participanți.

**Inițializare.**  $n$  numărul participanților,  $k$  pragul minim de reconstrucție al secretului  $S$ . Se aleg  $n$  valori (publice) distincte  $x_1, \dots, x_n$  și se distribuie fiecărui participant  $i$  valoarea  $x_i$ .

**PASUL 1.** Se alege de către *autoritatea de distribuție a secretului TP* (Trusted Party) un număr prim  $q$  suficient de mare ( $q \geq n + 1$ ). Se generează *aleatoriu*, de către autoritatea de distribuție a secretului  $TP$ , un polinom de grad  $k - 1$ :

$$P(X) = \sum_{i=1}^{k-1} a_i X^i + S \bmod q.$$

**PASUL 2** (distribuția secretului). Autoritatea  $TP$  distribuie participantului  $i$  valoarea  $y_i = P(x_i)$ ,  $i = 1, \dots, n$ .

**PASUL 3** (recuperarea secretului). Cu informația oferită de  $k$  participanți se poate recupera, prin rezolvarea unui sistem liniar de  $k$  ecuații, valoarea  $S$ . Dacă numărul participanților care pun la dispoziție informația  $y_i$  este mai mic decât  $k$ , atunci *nu se poate* determina  $S$ .

### 24.2 Exerciții rezolvate

**Exercițiul 24.2.1** Să se partajeze secretul  $S = 13$ , pentru o schema majoritară  $k = 3$  din  $n = 5$  participanți, utilizând algoritmul lui Shamir specificat de  $q = 17$ , valorile publice  $x_i = i$ ,  $i = 1, \dots, 5$  și valorile aleatoare  $a[1] = 10$ ,  $a[2] = 2$ .

*Rezolvare:* Se obține polinomul  $P(X) = a_2 X^2 + a_1 X + S = 2X^2 + 10X + 13$ .

Secretul se partajează în:

$$y_1 = P(1) = (2 + 10 + 13) \bmod 17 = 8;$$

$$y_2 = P(2) = (8 + 20 + 13) \bmod 17 = 7;$$

$$y_3 = P(3) = (18 + 30 + 13) \bmod 17 = 10;$$

$$y_4 = P(4) = (32 + 40 + 13) \bmod 17 = 0;$$

$$y_5 = P(5) = (50 + 50 + 13) \bmod 17 = 11.$$

## 24.3 Exerciții propuse

**Exercițiul 24.3.1** Să se partajaze secretul  $S = 4$ , pentru o schema majoritară  $k = 3$  din  $n = 5$  participanți, utilizând algoritmul lui Shamir specificat de  $q = 17$ , valorile publice  $x_i = i$ ,  $i = 1, \dots, 5$  și valorile aleatoare  $a[1] = 10$ ,  $a[2] = 2$ .

Răspuns:  $\{16, 15, 1, 8, 2\}$ .

**Exercițiul 24.3.2** Să se partajaze secretul  $S = 0$ , pentru o schema majoritară  $k = 3$  din  $n = 5$  participanți, utilizând algoritmul lui Shamir specificat de  $q = 17$ , valorile publice  $x_i = i$ ,  $i = 1, \dots, 5$  și valorile aleatoare  $a[1] = 10$ ,  $a[2] = 2$ .

Răspuns:  $\{12, 11, 14, 4, 15\}$ .

**Exercițiul 24.3.3** Să se reconstituie secretul  $S$ , din valorile  $\{12, 4, 15\}$ , știind că acestea au fost obținute cu ajutorul schemei majoritare  $(5, 3)$  a lui Shamir specificată de  $q = 17$  și valorile publice  $\{1, 4, 5\}$ .

Răspuns:  $S = 0$ .

**Exercițiul 24.3.4** Să se reconstituie secretul  $S$ , din valorile  $\{1, 8, 2\}$ , știind că acestea au fost obținute cu ajutorul schemei majoritare  $(5, 3)$  a lui Shamir specificată de  $q = 17$  și valorile publice  $\{3, 4, 5\}$ .

Răspuns:  $S = 4$ .

**Exercițiul 24.3.5** Să se reconstituie secretul  $S$ , din valorile  $\{10, 0, 11\}$ , știind că acestea au fost obținute cu ajutorul schemei majoritare  $(5, 3)$  a lui Shamir specificată de  $q = 17$  și valorile publice  $\{3, 4, 5\}$ .

Răspuns:  $S = 13$ .

**Exercițiul 24.3.6** Ce se întâmplă dacă în protocolul lui Shamir se renunță la condiția de primalitate asupra lui  $q$ ?

# Capitolul 25

## Scheme de partajare a secretelor bazate pe CRT

### 25.1 Breviar teoretic

Una dintre primele scheme de partajare a secretelor, bazate pe CRT, este *schema Mignotte*. Aceasta presupune faptul că șirul  $p_1 < p_2 < \dots < p_n$  este un șir Mignotte:

$$\prod_{i=0}^{k-2} p_{n-i} < \prod_{i=0}^k p_i.$$

Secretul  $S$ , ce trebuie partajat, trebuie să aparțină intervalului<sup>1</sup>  $(\beta, \alpha)$ , unde  $\alpha = \prod_{i=0}^k p_i$  și  $\beta = \prod_{i=0}^{k-2} p_{n-i}$ . Valorile ce se distribuie fiecăruia dintr-acei  $n$  participanți sunt  $S \bmod p_i$ ,  $i = 1, \dots, n$ . Recuperarea secretului se realizează, de către  $k$  participanți, prin rezolvarea, cu ajutorul CRT, a sistemului  $S = S_{i_j} \bmod p_i$ ,  $j = 1, \dots, k$ .

### 25.2 Exerciții rezolvate

**Exercițiul 25.2.1** Fie șirul  $\{5, 7, 9, 11, 13\}$  o secvență  $(5, 3)$  Mignotte,  $\alpha = 11 \cdot 13$ ,  $\beta = 5 \cdot 7 \cdot 9$ , secretul  $S = 235 \in (\alpha, \beta)$ . Care sunt secretele ce sunt distribuite celor cinci participanți?

*Rezolvare:*  $S_1 = S \bmod 5 = 0$ ,  $S_2 = S \bmod 7 = 5$ ,  $S_3 = S \bmod 9 = 6$ ,  $S_4 = S \bmod 11 = 10$ ,  $S_5 = S \bmod 13 = 12$ . Spre exemplu, grupul  $\{P_1, P_3, P_4\}$  trebuie să rezolve problema:

$$\begin{cases} x \equiv 0 & \bmod 5 \\ x \equiv 6 & \bmod 9 \\ x \equiv 10 & \bmod 11 \end{cases}$$

ce are soluție unică 285.

---

<sup>1</sup>Dacă lungimea intervalului este mică, atunci schema nu este practică, existând posibilitatea ca printre valorile distribuite să existe coliziuni.



# Capitolul 26

## Canale subliminale

### 26.1 Breviar teoretic

În sistemul de autentificare ElGamal,  $A$  alege un număr prim mare  $q$  și un element primitiv  $\alpha \in Z_q$ . Valorile  $q$  și  $\alpha$  sunt publice. Printr-un canal sigur,  $A$  și  $B$  stabilesc un număr  $p \in Z_q$ . Protocolul prin care  $A$  transmite lui  $B$  mesajul subliminal  $y \in Z_q$  prin utilizarea textului  $x$  este următorul:

**PASUL 0.**  $A$  calculează  $\beta = \alpha^p \bmod q$ .

**PASUL 1.** Se determină  $\gamma$  ca soluție a ecuației  $x = p \cdot \beta + y \cdot \gamma \bmod (q - 1)$ .

**PASUL 2.**  $A$  trimite lui  $B$  tripletul  $(x, \beta, \gamma)$ .

**PASUL 3.**  $B$  calculează  $a = (\alpha^p)^\beta \cdot \beta^\gamma \bmod q$ .

**PASUL 4.** Dacă  $a = \alpha^x \bmod q$  atunci  $B$  decide că mesajul este autentic.

**PASUL 5.**  $B$  recuperează mesajul subliminal:  $y = (x - p \cdot \beta) \cdot \gamma^{-1} \bmod (q - 1)$ .

### 26.2 Exerciții rezolvate

**Exercițiul 26.2.1** *Se consideră canalul subliminal ElGamal dat de  $q = 11$  și  $\alpha = 2$ . Să presupunem că se dorește transmiterea mesajului  $y = 9$  folosind cheia secretă  $k = 0$  și textul cifrat  $x = 5$ . Care este mesajul ce se va transmite pe canalul de comunicație?*

*Rezolvare:*

**PASUL 0.**  $A$  calculează  $\beta = \alpha^p \bmod q = 2^0 \bmod 11 = 1$ .

**PASUL 1.** Se determină  $\gamma$  ca soluție a ecuației  $x = p \cdot \beta + y \cdot \gamma \bmod (q - 1)$ , echivalent cu  $5 = 0 + 9\gamma \bmod 10$  de unde rezultă  $\gamma = 5 \cdot 9^{-1} \bmod 10 = 5 \cdot 9 \bmod 10 = 5$ .

**PASUL 2.**  $A$  trimite lui  $B$  tripletul  $(x, \beta, \gamma) = (5, 1, 5)$ .

### 26.3 Exerciții propuse

**Exercițiul 26.3.1** *Se consideră canalul subliminal ElGamal dat de  $q = 11$  și  $\alpha = 2$ . Să presupunem că se dorește transmiterea mesajului  $y = 9$  folosind cheia secretă  $k = 8$  și textul*

cifrat  $x = 5$ . Care este mesajul ce se va transmite pe canalul de comunicație?

*Răspuns:*  $\{5, 6, 3\}$ .

**Exercițiul 26.3.2** Se consideră canalul subliminal ElGamal dat de  $q = 11$  și  $\alpha = 2$ . Să presupunem că se dorește transmiterea mesajului  $y = 1$  folosind cheia secretă  $k = 8$  și textul cifrat  $x = 5$ . Care este mesajul ce se va transmite pe canalul de comunicație?

*Răspuns:*  $\{5, 2, 9\}$ .

**Exercițiul 26.3.3** Se consideră canalul subliminal ElGamal dat de  $q = 11$ ,  $\alpha = 2$  și cheia secretă  $k = 8$ . Se recepționează mesajul  $\{5, 6, 3\}$ . Acesta conține mesaje ascunse?

*Răspuns:* Mesajul recepționat este autentic, mesajul subliminal fiind  $y = 9$ .

**Exercițiul 26.3.4** Se consideră canalul subliminal ElGamal dat de  $q = 11$ ,  $\alpha = 2$  și cheia secretă  $k = 8$ . Se recepționează mesajul  $\{5, 6, 2\}$ . Acesta conține mesaje ascunse?

*Răspuns:* Mesajul recepționat nu este autentic.

**Exercițiul 26.3.5** Se consideră canalul subliminal ElGamal dat de  $q = 11$ ,  $\alpha = 2$  și cheia secretă  $k = 8$ . Se recepționează mesajul  $\{5, 2, 9\}$ . Acesta conține mesaje ascunse?

*Răspuns:* Mesajul recepționat este autentic, mesajul subliminal fiind  $y = 1$ .

**Exercițiul 26.3.6** Se consideră canalul subliminal ElGamal dat de  $q = 11$ ,  $\alpha = 2$  și cheia secretă  $k = 0$ . Se recepționează mesajul  $\{5, 6, 5\}$ . Acesta conține mesaje ascunse?

*Răspuns:* Mesajul recepționat este autentic, se ajunge la rezolvarea următoarei ecuații  $5 = 5 \times y \bmod 10$  ce nu are soluție unică, verificarea autenticității se face prin repetarea procedurii de construcție a mesajului ce se transmite. Se obține mesajul ascuns  $y = 9$ .

**Exercițiul 26.3.7** În cadrul protocolului ElGamal, de transmitere a mesajelor subliminale, autentificatorul obținut  $\gamma$  nu este relativ prim cu  $q - 1$ . Cum se rezolvă această speță?

# Capitolul 27

## Principii criptografice

**Exercițiul 27.1** *Metoda one-time pad (OTP) cifrează un mesaj  $m$  prin aplicarea operației XOR cu o cheie secretă  $k$ . Având în vedere că o cheie bună are, statistic, jumătate din biți zero și că operația XOR cu zero nu modifică nimic, rezultă că metoda OTP lasă jumătate din mesaj în clar. Cu alte cuvinte, prin simpla observare a unui text cifrat cu această metodă, un atacator cunoaște jumătate din biții textului clar. Acest lucru înseamnă, de fapt, că metoda OTP este una foarte slabă? Cum poate fi considerat ”perfect” un cifru bloc care cifrează numai jumătate din textul clar?*

**Exercițiul 27.2** *Verificarea semnăturii El Gamal presupune efectuarea operației  $a^x b^y \bmod p$  unde  $a, b$  sunt fixate iar  $x, y$  sunt variabile. Arăți că numărul de înmulțiri necesare pentru efectuarea acestui calcul este mai mic decât numărul de operații necesare pentru a calcula  $a^x b^y \bmod p$  prin două exponențieri succesive.*

**Exercițiul 27.3** *Considerăm două numere prime  $p$  și  $q$ . Fie  $i_p = p^{-1} \bmod q$  și  $i_q = q^{-1} \bmod p$  iar  $n = p \cdot q$ . Care este valoarea rezultată în urma operației  $q \cdot i_q + p \cdot i_p$ ? Puteți explica cum poate fi folosită această valoare pentru a reduce stocarea cheii secrete la implementarea RSA CRT?*

**Exercițiul 27.4** *Se dorește semnarea a două mesaje cu algoritmul de semnătură El Gamal. Cum putem calcula valorile  $g^{k_1}$  și  $g^{k_2}$  pentru a produce semnăturile într-un timp mai scurt decât cel necesar pentru a calcula două semnături secvențiale?*

**Exercițiul 27.5** *Considerăm protocolul Fiat-Shamir unde secretul  $s$  este ales astfel încât  $vs^2 = 1 \bmod n$ ,  $v$  fiind cheia publică. Protocolul este după cum urmează:*

- Alice alege un  $r$  aleator și îi trimite lui Bob  $x = r^2 \bmod n$ ;
- Bob răspunde cu un bit aleator  $e$ ;
- Alice răspunde cu  $y = s^e r \bmod n$ ;



- Bob verifică dacă  $y^2 = v^e x \bmod n$ .

Arătați că valorile rezultate în urma protocolului, adică  $\{x, r, y\}$ , definesc o distribuție ce poate fi simulată fără a-l folosi pe  $s$ . Explicați de ce acest lucru asigură protocolului o securitate foarte bună.

**Exercițiul 27.6** Se dă o cutie neagră care rulează algoritmul AES (12 runde pentru o cheie de 192 biți); cutia conține o cheie necunoscută  $k$  și acceptă ca parametru un întreg  $r$  a cărui valoare poate fi setată la 12, 11 sau 10 de către utilizator. Vi se permite să introduceți în cutie texte clare după cum doriți. Cum ați proceda pentru a ataca această implementare?

**Exercițiul 27.7** Un administrator de sistem are o cheie de 100 de biți pe care dorește să o împartă celor doi utilizatori în care are încredere în mod egal. El dorește ca accesul la informație să fie posibilă numai când cei doi cooperează. Câți biți din cheie ar trebui să dea fiecăruia din cei doi utilizatori?

**Exercițiul 27.8** Pentru a grăbi verificarea semnăturilor  $s_i$  de tip RSA a mesajelor  $m_i$ , se folosește următoarea idee: se verifică dacă  $(\prod s_i)^e = \prod \text{hash}(m_i) \bmod n$  unde "hash" reprezintă full domain hash - o schemă de semnătură bazată pe RSA care mai întâi aplică o funcție hash și apoi semnătura RSA. Arătați că această idee nu este sigură pentru un exponent  $e$  mic și propuneți o contramăsură.

**Exercițiul 27.9** De ce următorul context este nesigur? O autoritate de încredere generează un modul RSA  $n$  a cărui factorizare rămâne secretă. Autoritatea furnizează fiecărui utilizator din sistem o pereche  $(e_i, d_i)$  așa încât  $e_i d_i = 1 \bmod \phi(n)$  unde  $i \neq j \Rightarrow d_i \neq d_j$ .

**Exercițiul 27.10** Să presupunem că cineva trimite mesaje cifrate utilizând DES în modul de operare OFB cu o valoare inițială secretă (fixată)  $IV$ .

- 1) Arătați cum poate fi efectuat un atac cu text clar pentru a decripta mesajele transmise?
- 2) Este mai bun modul de operare CFB?
- 3) Dar modul de operare CBC?

**Exercițiul 27.11** După ce a studiat protocolul Diffie-Hellman, un tânăr criptograf decide să îl implementeze. Pentru a simplifica implementarea, el hotărăște să folosească grupul aditiv  $(\mathbb{Z}_p, +)$  în locul grupului multiplicativ  $(\mathbb{Z}_p^*, \cdot)$ . În calitate de criptograf cu experiență, ce credeți despre acest protocol?

**Exercițiul 27.12** Să presupunem că Alice și Bob folosesc chei publice RSA cu același modul  $n$  dar cu exponenți publici diferiți  $e_1$  și  $e_2$ .

- 1) Arătați că Alice poate decripta mesajele trimise lui Bob;
- 2) Arătați că Alice poate decripta mesajele trimise către Alice și Bob dacă  $\gcd(e_1, e_2) = 1$ .

**Exercițiul 27.13** Presupunem că  $n = p \cdot q$ , unde  $p$  și  $q$  sunt numere prime distincte.

- 1) Calculați  $S = n + 1 - \phi(n)$ .
- 2) Care sunt rădăcinile ecuației  $x^2 - Sx + n$ ? Dați expresiile acestor rădăcini și explicați cum pot fi găsite  $p$  și  $q$  cu ajutorul unui simplu algoritm pentru calculul rădăcinilor pătrate întregi?
- 3) Factorizați  $n$  în următoarele două cazuri:
  - a)  $n = 667, \phi(n) = 616$ ;
  - b)  $n = 15049, \phi(n) = 14800$ .

**Exercițiul 27.14** Să construim un MAC folosind modul CFB de implementare, în loc de modul CBC: fiind date blocurile de text clar  $\alpha_1, \dots, \alpha_n$ , definim vectorul de inițializare  $\beta_0 = \alpha_1$ . Apoi cifrăm secvența de blocuri  $\alpha_2, \dots, \alpha_n$  după formulele:

$$\beta_i = \alpha_{i+1} \oplus E(\beta_{i-1}; K).$$

În final,  $MAC(\alpha_1 || \dots || \alpha_n) = E(\beta_{i-1}; K)$ . Arătați că acesta este identic cu CBC MAC.

**Exercițiul 27.15** Pentru S-boxul  $S_5$  din DES calculați tendința variabilei aleatoare:

$$X_2 \oplus Y_1 \oplus Y_2 \oplus Y_3 \oplus Y_4.$$

**Exercițiul 27.16** Într-un sistem de cifrare simetric, o cheie  $k$  este slabă dacă  $e_k = d_k$ . Determinați toate cheile slabe ale sistemelor afine peste  $\mathbf{Z}_{15}$ .



# Capitolul 28

## Atacuri în mediul de implementare

### 28.1 Breviar teoretic

Atacurile în mediul de implementare presupun o serie de măsurători hardware asupra modului criptografic:

*Atacuri prin măsurarea timpului de execuție.* Prin măsurarea *timpului necesar efectuării unor operații asupra cheii private*, atacatorul poate determina exponenții utilizați în protocolul Diffie-Hellman, factorul RSA (în special asupra algoritmului RSA ce folosește pentru semnătură lema chinezescă a resturilor CRT), precum și o serie de alte sisteme criptografice cum ar fi algoritmul de semnătură digitală DSS.

*Atacuri prin măsurarea puterii consumate.* Atacul cu ajutorul *analizei simple a puterii* (SPA) constă în măsurarea puterii consumate de dispozitiv în timpul operației criptografice. Acest tip de atac se aplică, de regulă, dispozitivelor cu sursă de tensiune exterioară (ca de exemplu smart-cardurile). Consumul de putere depinde de instrucțiunea executată. Astfel, monitorizând consumul de putere, se poate deduce secvența de instrucțiuni (codul sursă). Dacă secvența de instrucțiuni depinde de lungimea cheii, atunci consumul de putere poate da informații despre cheie. În majoritatea procesoarelor, patternul puterii consumate de o instrucțiune depinde și de valoarea operanzilor (de exemplu setarea unui bit într-un registru consumă mai multă energie decât ștergerea acestuia). Măsurători efectuate asupra mai multor intrări pot deduce valoarea operandului. Tehnica se numește *analiza diferențială a puterii* (DPA).

*Atacuri cu ajutorul defecțiunilor (erorilor) hardware.* Echipamentele hardware pot genera erori (tranziente, latente sau induse) în timpul efectuării unor operații aritmetice. Prin exploatarea rațională a acestor erori se pot recupera cheia privată pentru algoritmi de semnătură RSA și Rabin. O serie de protocole criptografice cum ar fi Fiat-Schamir și Schnorr se pot sparge prin folosirea judicioasă a rezultatelor acestor erori.

*Analiza diferențială a defecțiunilor.* *Analiza diferențială a defecțiunilor* (DFA) este o schemă ce se utilizează pentru recuperarea cheilor secrete ale unui sistem criptografic dintr-un dispozitiv HSM (Hardware Security Module) securizat fizic. Modelul de defect este acela al defectelor tranziente (aleatoare) și al defectelor induse. Metoda folosește la identificarea

cheilor în cazul utilizării unor cifruri cunoscute (de exemplu DES) și/sau a unor cifruri cu algoritm necunoscut sau la reconstrucția algoritmului (cu o structură cunoscută).

## 28.2 Exerciții propuse

**Exercițiul 28.2.1** *Arătați că tehnica DPA poate fi accelerată folosind un compromis spațiu-timp.*

*Rezolvare:* Faceți referire la articolul *Computational Improvements to Differential Side Channel Analysis*, NATO Advanced Research Workshop on Security and Embedded Systems, August 2005.

**Exercițiul 28.2.2** *Descrieți un atac prin măsurarea timpului de execuție asupra unei proceduri de comparație a parolelor.*

**Exercițiul 28.2.3** *Pentru a proteja implementarea RSA de un atac prin măsurarea timpului de execuție, dezvoltatorii decid să adauge la finalul procedurii un timp de așteptare de durată aleatoare, cuprins între 0 și  $n$  tacturi de ceas. În acest fel, se va elimina total riscul atacului sau acesta va fi doar încetinit?*

**Exercițiul 28.2.4** *Numiți 3 factori care determină forma graficului puterii consumate de un microprocesor.*

*Rezolvare:* Instrucțiunea, datele manipulate de instrucțiune și adresa instrucțiunii.

# Capitolul 29

## Resurse software

### 29.1 CrypTool

CrypTool este un pachet software dedicat simulării și analizei de mecanisme criptologice într-un mod ilustrativ. De la rolul inițial de instruire în domeniul securității personalului diverselor companii private, CrypTool a evoluat într-un proiect educațional de tip open source cu aplicații în domeniul criptografiei și majoritatea domeniilor conexe. Produsul vizează în primul rând studenții facultăților de matematică și informatică, a firmelor ce activează în domeniul securității informațiilor precum și a dezvoltatorilor de aplicații sau utilizatorilor de calculatoare în general care doresc să-și dobândească bagajul minimal de cunoștințe criptografice.

În prezent produsul este gratuit și disponibil în mai multe versiuni, prima dintre acestea fiind CrypTool 1.4.x dezvoltată integral în mediul C++. Aceasta s-a extins ulterior în alte două versiuni, încă aflate la nivel beta, ce folosesc standarde de dezvoltare de ultimă generație aflându-se într-o continuă actualizare. Astfel, în iulie 2008, s-a lansat CrypTool 2.0 dezvoltat în mediul C#, versiune ce furnizează o paletă mai largă de funcționalități combinată cu o interfață grafică cu facilități de tip "drag-and-drop". La începutul lui 2010 s-a lansat versiunea JCrypTool dezvoltată în mediul Java, avantajele acestei versiuni fiind că este independentă de platforma pe care rulează (Windows, Linux, Mac) și că folosește din plin puternicul instrument FlexiProvider prin care se pot încărca cu ușurință module criptografice în orice aplicație construită peste JCA (Java Cryptography Architecture).

CrypTool a fost dezvoltat în colaborare cu instituții de învățământ devenind astfel un soft educațional și un bun instrument de inițiere în domeniul criptologiei, folosindu-se în prezent cu succes în multe universități de prestigiu. Datorită manipulării facile a mecanismelor criptologice precum și a vizualizării și prezentării într-o manieră facilă și inedită a rezultatelor, CrypTool poate reprezenta componenta practică a cursurilor teoretice din domeniul criptologiei precum și o metodă rapidă de familiarizare cu componente esențiale ale acestui domeniu.

Produsul acoperă ambele ramuri ale criptologiei și anume *criptografia* și *criptanaliza*.

Sunt tratate majoritatea aspectelor fundamentale ale criptografiei. Astfel, produsul are

implementate facilități în cadrul fiecărui subdomeniu după cum urmează:

- criptografia clasică: cifrurile Caesar, substituție monoalfabetică, substituție omofonică, Vigenère, Hill, Playfair, ADFGVX, Addition, XOR, Vernam, Solitaire etc;
- criptografia simetrică modernă: cifrurile IDEA, RC2, RC4, DES, 3DES, DESX precum și toții finaliștii cifrului AES și anume MARS, RC6, Rijndael, Serpent and Twofish;
- criptografia asimetrică: RSA;
- criptografia hibridă: cifrarea datelor realizându-se cu algoritmi simetrici (AES), protecția cheii de cifrare fiind asigurată prin metode asimetrice (RSA);
- semnături digitale: RSA, DSA, ECDSA (Elliptic Curve Digital Signature Algorithm), Nyberg-Rueppel;
- funcții hash: MD2, MD4, MD5, SHA, SHA-1, SHA-2, RIPEMD-160;
- generatoare aleatoare: secude,  $x^2 \bmod n$ , LCG (linear congruence generator), ICG (inverse congruence generator).

În cadrul criptanalizei se regăsesc implementate majoritatea atacurilor standard după cum urmează:

- atac cu text cifrat: Caesar, Vigenère, Addition, XOR, Substitution, Playfair;
- atac cu text clar: Hill, Single-column transposition;
- atac manual: substituție mono alfabetică, Playfair, ADFGVX, Solitaire;
- atac prin forță brută: pentru toți algoritmii; se presupune fie că entropia textului clar este mică sau cheia este parțial cunoscută sau alfabetului textului clar este cunoscut;
- atacuri asupra RSA: bazate pe factorizare sau tehnici care apelează la structurile algebrice (latice);
- atacuri asupra sistemelor hibride: atacuri asupra RSA sau AES(side channels attacks);
- atacuri asupra semnăturilor digitale: RSA prin factorizare; viabil până la lungime de 250 biți (adica 75 cifre);
- atacuri asupra funcțiilor hash: generare coliziuni texte ASCII cu paradoxul zilelor de naștere (până la 40 biți);
- analiză aleatorism: bateria de teste FIPS-PUB-140-1, periodicitate, Vitany, entropie, histograme, autocorelații, testul de compresie ZIP etc.

În sprijinul utilizatorilor, CrypTool are implementate o serie de demo-uri și animații prin care sunt exemplificate diverse facilități pe care produsul le oferă folosindu-se primitive criptografice suportate și implementate în aplicație ca de exemplu Caesar, Vigenère, Nihilist, DES (toate patru cu ANIMAL), Enigma (Flash), Rijdael/AES (Flash and Java), criptare hibridă și decriptare (AES-RSA și AES-ECC), generare și verificare de semnături digitale, protocolul de schimb de chei Diffie-Hellman, secret sharing (CRT sau Shamir), metoda challenge-response (autentificare), atacuri tip side-channel, securizarea e-mail-ului prin protocolul S/MIME (Java și Flash), prezentări grafice 3D pentru date (pseudo)aleatoare, sensibilitatea funcțiilor hash privind modificări ale textului clar, teoria numerelor și cripto sisteme RSA (Authorware).

CrypTool conține și un modul educațional interactiv dedicat aplicațiilor criptografice ce necesită aspecte elementare de teoria numerelor denumit "NT". Acest modul introduce utilizatorul în probleme elementare de teoria numerelor precum algoritmul lui Euclid pentru

găsirea celui mai mare divizor comun, testul Fermat pentru primalitate, factorizarea Fermat, factorizarea Pollard Rho și altele.

Un alt avantaj al produsului CrypTool îl reprezintă existența unui meniu de documentare consistent și o extindere online a acestuia conținând în plus explicații privind noțiuni generale de criptografie, o cronologie privind dezvoltarea domeniului, exemple de utilizare a facilităților aplicației, index sortat pe topicuri criptografice și listă de referințe.

Faptul că pachetul software este open source, că acoperă aspecte legate atât de criptografia clasică cât și cea modernă, a modalităților multiple de simulare și vizualizare originale, precum și a modului facil de aplicare și analiză a mecanismelor criptografice ne conduc la concluzia că pachetul CrypTool reprezintă atât o modalitate rapidă de inițiere în domeniul criptografiei cât și un instrument de lucru puternic pentru specialiști în vederea studierii și aplicării în același mediu a a diverse probleme concrete ce pot apărea în criptografie și criptanaliză.

## 29.2 OpenSSL

OpenSSL este o suită de aplicații ce implementează protocoalele Secure Sockets Layer (SSL v2/v3) și Transport Layer Security (TLS v1) precum și o librărie dedicată ce acoperă o gamă largă de primitive criptografice. Proiectul este manageriat de o comunitate de voluntari din întreaga lume ce comunică, folosind Internetul, în vederea planificării și dezvoltării continue a toolkit-ului OpenSSL precum și a documentației aferente.

OpenSSL este bazat pe librăria SSLeay dezvoltată de Eric A. Young și Tim J. Hudson, proiect încheiat la sfârșitul anului 1998. Asupra produsului acționează o dublă licențiere, atât cea de OpenSSL cât și cea originală a librăriei SSLeay. Ambele tipuri de licențe sunt de tipul BSD open-source, toolkit-ul putând astfel fi folosit atât pentru scopuri comerciale cât și non-comerciale. Pachetul software folosește instrumente criptografice puternice, fiind dezvoltat continuu și distribuit legal de câteva țări europene, supunându-se însă unor restricții de import/export și uz în unele țări din lume.

OpenSSL este disponibil în numeroase versiuni fiind într-o continuă dezvoltare, bug-uri fiind des semnalate și corectate. Versiunea stabilă curentă este OpenSSL 0.9.8m aceasta fiind disponibilă din luna februarie 2010; în plus utilizatorii beneficiază de acces online permanent pentru studierea dezvoltărilor ulterioare ultimei versiuni stabile. Versiunile sunt disponibile pentru majoritatea sistemelor de operare tip UNIX (incluzând Solaris, Linux, Mac OS X și cele patru sisteme de operare BSD open source), Open VMS și Microsoft Windows.

OpenSSL implementează protocoalele SSL și TLS. Transport Layer Security (TLS) și predecesorul său Secure Sockets Layer (SSL), sunt protocoale criptografice ce furnizează securitatea comunicațiilor peste rețele similare Internetului. Cele două protocoale permit aplicațiilor de tip client/server să comunice securizat. TLS furnizează autentificare endpoint precum și confidențialitatea comunicațiilor peste Internet folosindu-se securizare RSA suportând lungimi de chei de până la 2048 de biți. Protocoale sunt utilizate pentru navigare pe Internet, poștă electronică, voice-over-IP (VoIP) etc.



Librăria criptografică OpenSSL implemenează o gamă largă de algoritmi utilizați în diverse standarde utilizate în Internet. Facilitățile furnizate de această librărie sunt folosite pentru a implementa SSL, TLS și S/MIME, precum și pentru SSH, OpenPGP și alte standarde criptografice. Librăria are implementate o varietate de primitive criptografice și alte facilități după cum urmează:

- Algoritmi de cifrare simetrice: Blowfish, CAST, DES, IDEA, RC2, RC4, RC5;
- Algoritmi de cifrare asimetrice: RSA (bazat pe factorizarea numerelor mari), DSA (bazat pe problema logaritmului discret), EC (curbe eliptice) Diffie-Hellman key exchange;
- Certificate digitale: X509, X509v3;
- Funcții hash și coduri de autentificare: HMAC, MD2, MD4, MD5, MDC2, RIPEMD, SHA;
- Funcții de control a intrărilor și ieșirilor, funcții de codificare a datelor: PKCS7, PKCS12, ASN1, BIO, EVP, PEM.

Utilitarul *OpenSSL* este un tool linie comandă utilizat în gestionarea diverselor funcții criptografice din librăria OpenSSL. Acesta poate fi folosit pentru:

- Creare și management de chei private, chei publice și parametrii;
- Operații ce implică criptografia cu chei publice;
- Creare de certificate X.509 , CSRs și CRLs;
- Calculare de rezumate de mesaj;
- Cifrare și descifrare folosind diverse cifruri;
- testare clienți/serve (SSL/TLS);
- Semnături și cifrare de mail (S/MIME);
- Cereri, generări și verificări de mărci temporare.

OpenSSL este unul dintre puținele proiecte open source supuse validării de conformitate cu standardului FIPS 140-2, utilizat în securitatea calculatoarelor, dezvoltat de National Institute of Standards and Technology (NIST). Pachetul software în sine nu este validat, fiind dezvoltată o componentă software a acestuia denumită OpenSSL FIPS Object Module, aceasta fiind compatibilă cu OpenSSL fiind creată pentru a oferi posibilitatea produselor ce folosesc API de tip OpenSSL de a fi supuse validării de conformitate FIPS 140-2. În ianuarie 2006 această componentă fost certificată, aceasta fiind însă revocată în iulie 2006 datorită unor nelămuriri privind validitatea interacționării modulului cu software extern. În februarie 2007 produsul a fost recertificat.

Validarea OpenSSL FIPS Object Module este unică printre toate validările FIPS 140-2 prin faptul că producătorul pune la dispoziție întreg codul sursă. Prin urmare, folosit fără nicio modificare și construit pe orice platformă conform documentației pusă la dispoziție se obține direct un modul criptografic validat. Orice modificare minoră asupra codului implică necesitatea revalidării, proces costisitor (aproximativ 50000\$) și îndelungat (între 6 și 12 luni). Cea mai recentă validare open source este OpenSSL FIPS Object Module (Software Version: 1.2), FIPS 140-2 certificate #1051. În prezent nu există niciun alt produs open source supus validării FIPS 140-2 datorită lipsei de finanțare. Validarea versiunilor precedente au fost finanțate de sectorul comercial și sponsori guvernamentali, o parte dintre aceștia preferând să rămână anonimi.

## 29.3 Studiu de caz: Implementarea funcțiilor criptografice în MAPLE

În cadrul acestei secțiuni vom exemplifica, printr-o serie de exemple, modalitățile de rezolvare a problemelor propuse, în cadrul acestei culegeri, cu ajutorul aplicației software MAPLE.

**Exemplu 29.1** *Algoritmul de cifrare ElGamal.*

$p$  (ordinul grupului),  $\alpha$  (generatorul) numere prime publice;  
 $a$  cheia privată;  
 $\beta := \alpha^a \bmod p$  cheia publică;  
 $m$  mesajul clar;  
 $k$  număr aleator secret;  
 regula de cifrare:  $y_1 := \alpha^k \bmod p$ ;  $y_2 := (m * \beta^k) \bmod p$ ;  
 regula de descifrare:  $des := y_2 * (y_1^a)^{-1} \bmod p$ .

```
> p:=17;
> alpha:=14;
> a:=2;
> beta:=alpha^a mod p;
> m:=4;
> k:=4;
> y1:=alpha^k mod p;
> y2:=(m*(beta^k)) mod p;
> text_cifrat:=(y1,y2);
> text_descifrat:=y2*(y1^a)^(-1) mod p;
```

**Exemplu 29.2** *Algoritmul de semnătură ElGamal.*

$p$  și  $\alpha$  numere prime publice;  
 $a$  cheia secretă;  
 $\beta := \alpha^a \bmod p$  cheia publică;  
 $x$  mesajul ce trebuie semnat;  
 $k$  număr secret;  
 $\gamma := \alpha^k \bmod p$ ;  
 $\delta := (x - a * \gamma)k^{-1} \bmod (p - 1)$ ;  
 $sign := (\gamma, \delta)$ ;  
 verificarea semnăturii:  $\beta^\gamma * \gamma^\delta \bmod p = \alpha^x \bmod p$ .

```
> p:=467;
> alpha:=2;
```

```

> a:=127;
> beta:=alpha^a mod p;
> x:=102;
> k:=15;
> gamma:=alpha^k mod p;
> delta:=(x-a*gamma)*k^(-1) mod (p-1);
> (beta^gamma*gamma^delta - alpha^x) mod p;

```

**Exemplu 29.3** *Algoritmul de semnătură DSA.*

$p$  număr prim (public);  
 $q$  număr prim (public);  
 $\alpha$  (public) rădăcina de ordin  $q$  a unității;  
 $a$  cheia secretă;  
 $\beta = (\alpha^a) \bmod p$ ;  
 $x$  mesajul;  
 $k$  număr aleatoriu (secret);  
 $sign = (\gamma, \delta)$  unde  $\gamma = (\alpha^k \bmod p) \bmod q$  și  $\delta = (x + a * \gamma) * k^{-1} \bmod q$ .

```

> p:=7879;
> q:=101;
> alpha:=170;
> a:=75;
> beta:=(alpha^a) mod p;
> x:=1234;
> k:=50;
> gamma:=(alpha^k mod p) mod q;
> delta:=(x+a*gamma)*k^(-1) mod q;

```

**Exemplu 29.4** *Protocolul Diffie-Hellman.*

Caracteristicile protocolului:  
 $p$  număr prim (minim 1024 biți);  
 $q$  divizor prim al lui  $p - 1$  (minim 160 biți);  
 $\alpha$  element de ordin  $q$ ;  
 $a$  număr generat de  $A$  și trimis lui  $B$ ;  
 $b$  număr generat de  $B$  și trimis lui  $A$ ;  
 cheia comună este  $k := \alpha^{a*b} \bmod p$ .

```

> p:=25307;
> alpha:=2;

```

```

> a:=3578;
> b:=19956;
> k:=((alpha^a) mod p)^b mod p;

```

**Exemplu 29.5** *Protocolul Blom.*

$p$  număr prim,  $n$  numărul de utilizatori;  
 $k = 1$  nivel de compartimentare (protocolul este neconditionat sigur împotriva atacului unui utilizator);  
 $a, b, c$  coeficienții polinomului;  
 $A$  denumire generică participant protocol,  $r_A$  cheia publică a lui  $A$ ;  
 $f(X, Y) = a + b(X + Y) + cXY$  polinom (simetric),  $g_A(X) = f(X, r_A)$  polinomul secret al lui  $A$ .  
 $K$  matricea cheilor de compartimentare (simetrică).

```

> p:=29;
> a:=1;
> b:=2;
> c:=3;
> n:=3;
> r:=array(1..n, [13,11,17]);
> f(X,Y):=a+b*(X+Y)+c*X*Y;
> g:=array(1..n);
> for i from 1 to n do:
>   g[i]:=eval(f(X,Y),Y=r[i]) mod p;
> end do;
> K:=array(1..n, 1..n);
> for i from 1 to n do:
>   for j from 1 to n do:
>     K[i,j]:=eval(g[i],X=r[j]) mod p;
>   end do;
> end do;
> print(K);

```

**Exemplu 29.6** *Schema de partajare a lui Shamir.*

$n$  numărul de participanți;  
 $k$  numărul minim de participanți care pot reconstitui secretul;  
 $q$  număr prim (identifică corpul  $Z[q]$  în care se lucrează);  
 $S$  secretul care se dorește partajat;

$x_i$  (publice) se distribuie utilizatorilor,  $i = 1, \dots, n$ ;  
 $a_i$  (aleatoare),  $i = 1, \dots, k - 1$ .  
 > n:=5;  
 > k:=3;  
 > q:=17;  
 > S:=13;  
 > x[1]:=1;  
 > x[2]:=2;  
 > x[3]:=3;  
 > x[4]:=4;  
 > x[5]:=5;  
 > a[1]:=10;  
 > a[2]:=2;  
 > p:=S+a[1]\*x+a[2]\*x^2 mod q;  
 > for i from 1 to n do subs(x=x[i],p) mod q  
 > od;

**Exemplu 29.7** *Recuperarea secretului din schema lui Shamir.*

$n$  numărul de participanți;  
 $k$  numărul minim de participanți care pot reconstitui secretul;  $q$  număr prim (identifica corpul  $Z[q]$  în care se lucrează);  
 $S$  secretul care se dorește partajat;  
 $x_i$  (publice) se distribuie utilizatorilor,  $i = 1, \dots, n$ ;  
 $s_i$  secretul distribuit,  $i = 1, \dots, k - 1$ ;  
 > n:=5;  
 > k:=3;  
 > q:=17;  
 > x[1]:=1;  
 > x[2]:=2;  
 > x[3]:=3;  
 > x[4]:=4;  
 > x[5]:=5;  
 > s[1]:=8;  
 > s[2]:=7;  
 > s[3]:=10;  
 > p:=S+a[1]\*x+a[2]\*x^2 mod q;

```
> solve({subs(x=x[1],p)=s[1],subs(x=x[2],p)=s[2],subs(x=x[3],p)=s[3]
}, {S,a[1],a[2]});
```

**Exemplu 29.8** *Canalul subliminal ElGamal.*

$q$  număr prim;  
 $\alpha$  element primitiv;  
 $x$  mesaj cifrat;  
 $y$  mesaj subliminal;  
 $k$  cheia secretă;  
 $\beta$  autentificator;  
 $\gamma$  autentificator;  
 mesajul subliminal  $(x, \beta, \gamma)$ .

```
> q:=11;
> alpha:=2;
> y:=9;
> x:=5;
> k:=0;
> beta:=alpha^y mod q;
> gama:=y^(-1)*(x-k*beta) mod (q-1);
> M:=(x,beta,gama);
```

**Exemplu 29.9** *Extragerea datelor din canalul subliminal ElGamal.*

$q$  număr prim;  
 $\alpha$  element primitiv;  
 $x$  mesaj cifrat;  
 $y$  mesaj subliminal;  
 $k$  cheia secretă;  
 $\beta$  autentificator;  
 $\gamma$  autentificator;  
 mesajul subliminal  $(x, \beta, \gamma)$ .

```
> q:=11;
> alpha:=2;
> k:=0;
> x:=5;
> beta:=6;
> gamma:=5;
> a:=alpha^x mod q;
> b:=((alpha^k)^beta)*beta^gamma mod q;
```

```

> if( a= b) then print("Mesaj_Auth_OK");
> Mesaj_subliminal:=(x-k*beta)*gamma^(-1) mod (q-1);
> else print("Mesaj_Auth_FAIL")
> fi;

```

## 29.4 PARI/GP

PARI/GP a fost inițial dezvoltat în 1985 de o echipă condusă de către Henri Cohen, iar în prezent este menținut de Karim Belabas, ajutat de o mulțime de voluntari. Numele PARI provine de la faptul că la început inițiatorii proiectului au dorit să implementeze o librărie pentru aritmetica în limbajul de programare Pascal, "Pascal ARithmetic", iar partea GP vine de la Great Programmable Computer. Produsul este gratuit, versiunea stabilă curentă fiind 2.5.0, disponibilă din iunie 2011.

Scopul acestui program este facilitarea calculelor din teoria numerelor (factorizări, teoria algebrică a numerelor, curbe eliptice etc.), însă sunt incluse și alte funcții utile pentru calcule cu polinoame, matrice, numere algebrice etc.

PARI/GP recunoaște mai multe tipuri de elemente, dintre care menționăm:

- numere întregi;
- numere raționale: scriem  $a/b$ , cu  $a$  și  $b$  întregi;
- numere reale;
- numere complexe: scrise sub forma  $a+b*I$ , cu  $a$  și  $b$  reale;
- întregi modulo  $n$ : pentru  $n \bmod m$  scriem  $\text{Mod}(n,m)$ ;
- polinoame: de exemplu, pentru  $x^9 + 7x + \frac{6}{5}$  vom scrie  $x^9+7*x+6/5$ ;
- funcții polinomiale:  $P/Q$ , cu  $P$  și  $Q$  polinoame;
- polinoame modulo un polinom  $P$ :  $\text{Mod}(P,Q)$ , unde  $Q$  este un polinom;
- vectori: scriem  $v=[1,2,3]$  pentru un vector linie, și  $w=[1,2,3]$  pentru un vector coloană; prin  $v[i]$  se înțelege a  $i$ -a componentă a vectorului  $v$ ;
- matrice: liniile sunt separate prin punct și virgulă, iar elementele unei linii sunt separate prin virgulă; pentru o matrice  $A$ , prin  $A[i,j]$  înțelegem elementul aflat la intersecția dintre linia  $i$  și coloana  $j$ .

Funcțiile disponibile în PARI sunt numeroase:

*Funcții aritmetice.* Acestea sunt funcții al căror domeniu de definiție este  $\mathbb{Z}$  sau  $\mathbb{Z}_n$ . Dăm câteva exemple de astfel de funcții:

- **binary(x)**: transformă numărul întreg  $x$  din baza 10 în baza 2;
- **contfrac(x)**: scrierea întregului  $x$  ca fracție continuă;
- **bezout(x,y)**: returnează un vector  $v=[a,b,d]$ , unde  $d=\text{c.m.m.d.c.}(x,y)$ , iar  $a$  și  $b$  sunt astfel încât  $ax + by = d$ ;
- **divisors(x)**: vector care are drept componente divizorii lui  $x$ , în ordine crescătoare;
- **divrem(x,y)**: returnează câtul și restul împărțirii lui  $x$  la  $y$ ;
- **eulerphi(n)**: calculează valoarea funcției lui Euler  $\varphi(n)$ ;

- **factorint(n)**: returnează toți factorii primi ai lui  $n$ , împreună cu multiplicitățile lor;
- **gcd(x,y)**, **lcm(x,y)**: c.m.m.d.c( $x, y$ ), respectiv c.m.m.mc( $x, y$ );
- **isprime(x)**: returnează 1 dacă  $x$  este prim și 0 în caz contrar;
- **kronecker(x,y)**: returnează valoarea simbolului Legendre (sau a generalizării sale, simbolul lui Jacobi)  $(\frac{x}{y})$ ;
- **omega(x)**: numărul de divizori primi distincți ai lui  $x$ ;
- **znorder(x)**: calculează ordinul elementului  $x \in \mathbb{Z}_n$  în grupul  $\mathbb{Z}_n$ .

*Funții referitoare la polinoame.*

- **algdep(z,k)**: găsește un polinom din  $\mathbb{Z}[x]$  cu gradul cel mult egal cu  $K$  și care are pe  $z$  ca rădăcină;
- **factormod(f,p)**: factorizează un polinom din  $\mathbb{Z}[x]$  modulo numărul prim  $p$ ;
- **polidisc(f,x)**: returnează discriminatul polinomului  $f$ , privit ca polinom în nedeterminata  $x$ ;
- **polisirreducible(f)**: verifică ireductibilitatea polinomului  $f$ ;
- **polrecip(f)**: returnează polinomul obținut din  $f$  scriindu-se coeficienții în ordine inversă;
- **polresultant(f,g,x)**: calculează rezultanta polinoamelor  $f$  și  $g$ , privite ca polinoame în nedeterminata  $x$ ;
- **polroots**: returnează un vector coloană ale cărui componente sunt rădăcinile lui  $f$ , repetate conform multiplicității fiecăreia;
- **prod(x=a,b,f(x))**: calculează  $\prod_{x=a}^b f(x)$ ;
- **prodeuler(x=a,b,f(x))**: calculează produsul  $\prod_{a \leq p \leq b} f(p)$ , unde  $p$  este prim;
- **solve(x=a,b,f(x))**: găsește o rădăcină pentru  $f(x)$  cuprinsă între  $a$  și  $b$ , presupunând că  $f(a)f(b) \leq 0$ ;
- **sum(x=a,b,f(x))**: calculează  $\sum_{x=a}^b f(x)$ ;
- **sumdiv(n,x,f(x))**: sumează  $f(x)$  după toți divizorii pozitivi ai lui  $n$ .

*Funții referitoare la curbe eliptice.* PARI/GP are implementate câteva funcții care sunt foarte folositoare atunci când lucrăm peste curbe eliptice. Aceste funcții presupun o curbă eliptică în formă Weierstrass generalizată:

$$y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6$$

Astfel, o curbă eliptică poate fi creată dându-se un vector cu 5 componente. Punctele curbei sunt reprezentate ca vectori cu 2 componente, mai puțin punctul de la infinit, acesta fiind reprezentat prin [0].

Exemple de funcții referitoare la curbe eliptice sunt:

- **E=ellinit[a1,a2,a3,a4,a6]**: creează o curbă eliptică  $\mathcal{E}$  cu coeficienții  $a_1, a_2, a_3, a_4, a_6$ ;
- **E.disc**: returnează discriminantul curbei  $\mathcal{E}$ ;
- **E.j**: returnează  $j$ -invariantul curbei  $\mathcal{E}$ ;
- **elladd(E,P,Q)**: adună punctele  $P$  și  $Q$ , puncte ce aparțin curbei  $\mathcal{E}$ ;
- **ellap(E,p)**: returnează urma Frobenius ( $p$  este un număr prim);
- **ellisoncurve(E,P)**: adevărat dacă și numai dacă punctul  $P$  este pe curba  $\mathcal{E}$ ;



- `ellorder(E,P)`: returnează ordinul punctului  $P$ , dacă acesta este un punct de torsiune, altfel returnează 0;
- `ellordinate(E,x)`: găsește  $y$  astfel încât punctul  $(x,y)$  aparține curbei  $\mathcal{E}$ ; dacă nu există un astfel de  $y$ , returnează [];
- `ellpow(E,P,n)`: calculează punctul  $nP \in \mathcal{E}$ ;
- `ellsub(E,P,Q)`: calculează  $P - Q \in \mathcal{E}$ .

# Capitolul 30

## Concursuri publice

În acest capitol ne propunem să facem o scurtă descriere a celor 4 probleme date la MITRE Cyber Challenge <sup>1</sup>, în perioada 9-12 ianuarie 2012. Pentru fiecare problemă prezentăm și câte o sugestie de rezolvare.

Primele trei probleme sunt legate între ele, în sensul că pentru rezolvarea celei de-a doua probleme este nevoie de parola obținută în urma rezolvării primei probleme, iar rezolvarea celei de-a doua probleme ne conduce la un indiciu folositor în rezolvarea problemei cu numărul trei. Ultima problemă este independentă de primele trei, aceasta având de fapt rolul de a scoate în evidență o vulnerabilitate a ECDSA (același tip de vulnerabilitate care a fost folosită și pentru aflarea cheii de semnare de la PlayStation3).

**Problema 1.** Obiectivul primei probleme este acela de a recunoaște când s-a folosit criptografia clasică (cifrurile Caesar, Vigenère, Hill etc) în mediul digital.

Scenariul ipotetic este următorul: găsim un fișier “ciudat, pe care nu l-am creat noi, în calculatorul personal. Acest fișier, `neededinformation.txt`, este pus la dispoziție în cadrul problemei.

Se cere decriptarea informației conținute în acest fișier și găsirea parolei ascunse în interiorul său. Știm că această parolă începe cu "S", se termină cu "D" și este formată numai din majuscule.

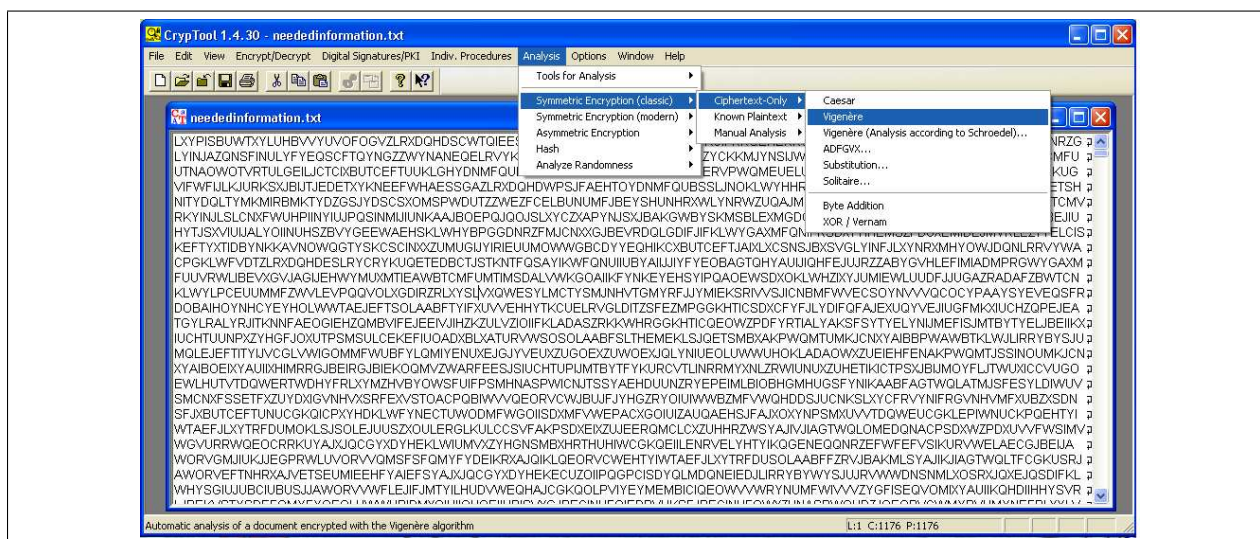
Problema se poate rezolva foarte ușor folosind pachetul software **CrypTool** pentru a face o criptanaliză a `neededinformation.txt`: *Analysis*  $\rightsquigarrow$  *Symmetric encryption(classic)*  $\rightsquigarrow$  *Ciphertext-Only*  $\rightsquigarrow$  *Vigenère*.

În urma acestei criptanalize rezultă pentru început că lungimea cheii folosite este 6, iar la următorul pas obținem cheia "SQUARE" cu ajutorul căreia putem decripta textul conținut în `neededinformation.txt`. La sfârșitul textului decriptat se află și parola pe care o căutam: "[...]THEPASSWORDFORTOMMOROWISSTRONGPASSWORDSAREGOOD".

**Problema 2.** Această problemă își propune să arate posibilele locuri în care un adversar poate ascunde informații, precum și modurile în care acest lucru se poate face. Mai precis

---

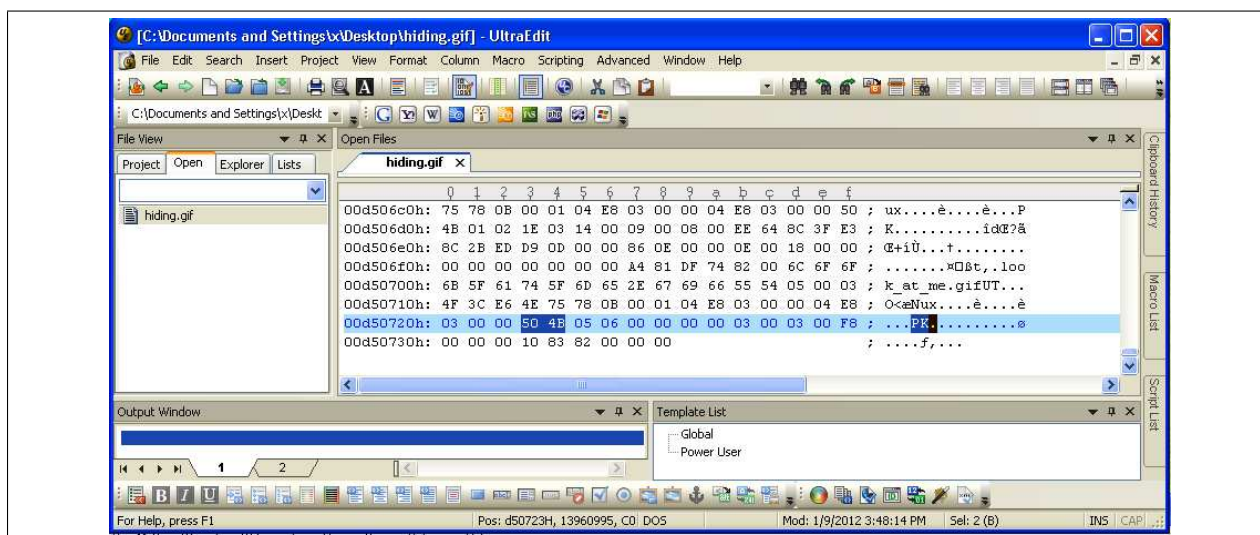
<sup>1</sup><http://www.iccs.fordham.edu/mitre/>



problema presupune găsirea unor informații ascunse în interiorul unei imagini.

Presupunem că avem o imagine `hiding.gif`. Cerința problemei este aceea de a găsi informația ascunsă în această imagine, știind că aceasta începe cu "h", se termină cu "l", iar mărimea fiecărei litere contează. De asemenea, așa cum am menționat anterior, vom avea nevoie de parola obținută la prima problemă.

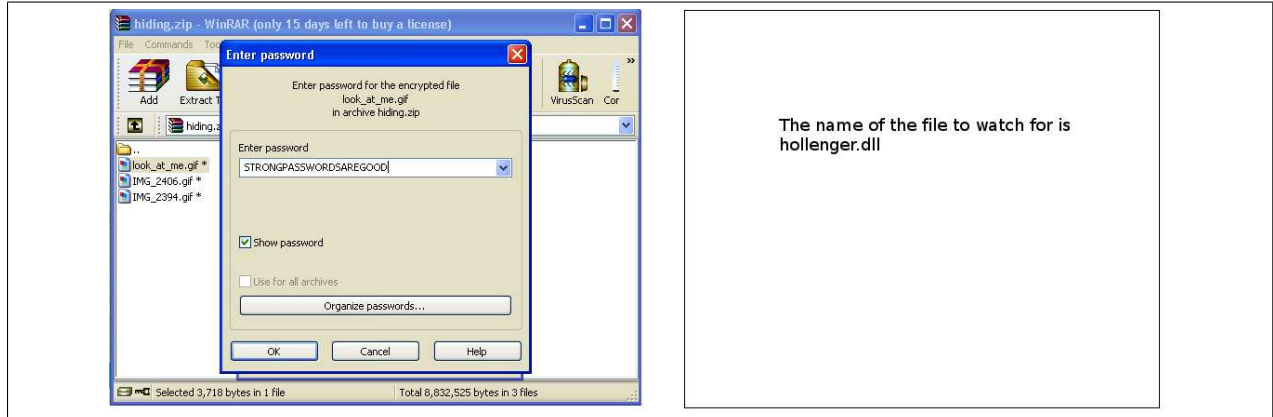
Uitându-ne la proprietățile imaginii `hiding.gif`, observăm că aceasta are 13.3 MB, ceea ce ni se pare suspect de mult. Pentru a vedea mai multe detalii, deschidem `hiding.gif` cu **UltraEdit** și observăm că apare "PK" în format hexa 50 4B), ceea ce înseamnă că este vorba despre o arhivă (PK reprezintă inițialele lui Phil Katz).



Prin urmare schimbăm extensia și obținem `hinding.zip`. Deschizând această arhivă găsim alte imagini, una dintre ele (care atrage atenția în mod deosebit) fiind `look_at_`

me.gif. Pentru a putea vedea această imagine însă, avem nevoie de parola obținută la problema 1.

Găsim în final și informația pe care o căutam, și anume `hollenger.dll`.

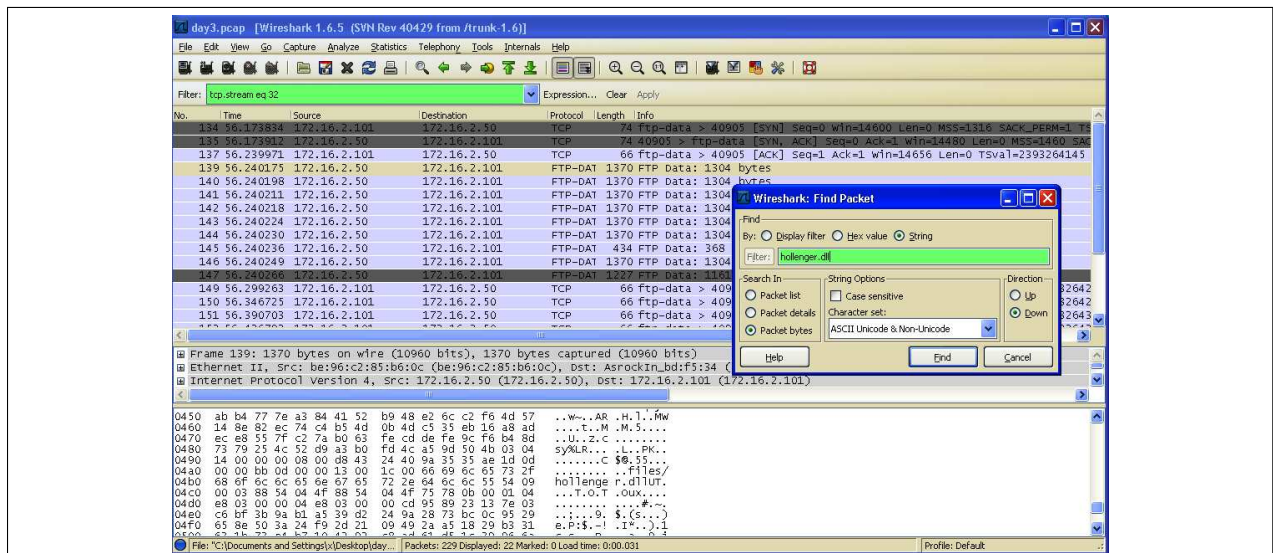


**Problema 3.** Cea de-a treia problemă este legată de analiza traficului de date.

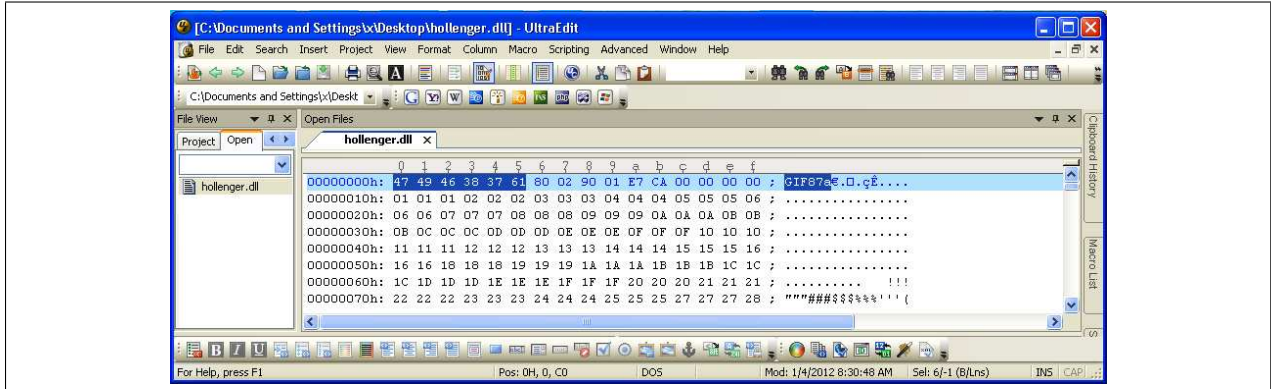
Presupunem că avem la dispoziție o captură de trafic de date, `day3.pcap`.

Se cere să se găsească, cu ajutorul răspunsului de la problema anterioară, fișierul transferat din calculatorul personal către o sursă necunoscută. Răspunsul pentru această problemă îl va constitui informația ascunsă în fișierul respectiv. Știm că începe cu "P", se termină cu "k" și mărimea fiecărei litere este importantă.

Pentru a putea deschide `day3.pcap` vom folosi **Wireshark**. În continuare căutăm `hollenger.dll` astfel: *Edit*  $\rightsquigarrow$  *Find Packet*  $\rightsquigarrow$  *Filter* : `hollenger.dll` (selectăm *Packet bytes* și *String*)  $\rightsquigarrow$  *Find*, iar apoi *Follow TCP stream*.



Observăm din nou PK și folosim opțiunea *Save as* pentru a obține `day3.zip`. Arhiva conține mai multe fișiere, printre care și `hollenger.dll`. Deschidem `hollenger.dll` cu UltraEdit și observăm "numărul magic" GIF87a (în format hexa 47 49 46 38 37 61), ceea ce înseamnă că este vorba de o imagine.



Schimbând deci extensia obținem `hollenger.gif`, aceasta fiind o imagine care conține următoarea frază : "The Root Password is **Pengu1nsR0ck**".

**Problema 4.** Obiectivul acestei probleme este recuperarea unei chei private ECDSA care a fost folosită pentru semnarea a două mesaje diferite.

Înainte însă de a continua prezentarea acestei ultime probleme, reamintim algoritmul de semnătură ECDSA:

Parametrii publici în acest caz sunt: un număr prim  $p$ , o curbă eliptică  $\mathcal{E}(\mathbb{F}_p)$  și un punct  $G \in \mathcal{E}(\mathbb{F}_p)$  cu  $\text{ord}G = q$ ,  $q$  prim.

Cheia publică (de verificare)  $V \in \mathcal{E}(\mathbb{F}_p)$  se construiește cu ajutorul cheii private (de semnare)  $1 \leq s \leq q - 1$  astfel:  $V = sG$ .

Semnătura mesajului  $m \pmod{q}$ , calculată cu ajutorul unei chei efemere  $e \pmod{q}$ , este definită ca fiind perechea  $(s_1, s_2) = (x_{eG} \pmod{q}, (m + ss_1)e^{-1} \pmod{q})$ , unde prin  $x_{eG}$  înțelegem coordonata  $x$  a punctului  $eG \in \mathcal{E}(\mathbb{F}_p)$ .

Semnătura  $(s_1, s_2)$  a mesajului  $m$  este verificată dacă are loc următoarea egalitate (în care  $v_1 = ds_2^{-1} \pmod{q}$  și  $v_2 = s_1s_2^{-1} \pmod{q}$ ) :  $x_{v_1G+v_2V} \pmod{q} = s_1$ .

Revenim acum la problema noastră. Datele care ne sunt puse la dispoziție se află în trei fișiere: `signatures.txt`, `parameters.der` și `public.oct`.

Primul fișier conține valorile hash-urilor și semnăturile pentru cele două mesaje (în format hexa):

```

m1=DE37B3145DB7359A0ACC13F0A4AFBD67EB496903
s11=ACB2C1F5898E7578A8A861BDF1CA39E7EF41EAC0B6AAA49468DD70E2
s12=BE4FA99C9D261C5F387A3ACE025702F6FB7884DD07CE18CAD48654B8
m2=28469B02BF0D2CFC86FF43CB612EE8FC05A5DBAA
s21=ACB2C1F5898E7578A8A861BDF1CA39E7EF41EAC0B6AAA49468DD70E2
s22=D3540E2B13E51605F5FEB8C87EE8E176E59213F31EA8B8FFDAD077E2

```



Observația importantă pe care se bazează însă întreaga rezolvare este aceea că valorile  $s_{11}$  și  $s_{21}$  sunt egale. În acest caz, dacă notăm cu  $e_1$ , respectiv  $e_2$  cheile efemere folosite pentru semnarea mesajelor  $m_1$ , respectiv  $m_2$ , rezultă fie că  $e_1 = e_2 = e$ , fie că  $e_1 + e_2 = q$ .

Vom arăta cum putem afla cheia privată  $s$  dacă presupunem că este vorba de primul caz, anume că pentru semnarea celor două mesaje diferite  $m_1$  și  $m_2$  s-a folosit aceeași cheie efemeră  $e$ . Notând cu  $r$  valoare comună  $s_{11} = s_{21}$ , avem următoarele două relații:

$$s_{21} = (m_1 + sr)e^{-1} \bmod q = r_1 \quad \text{și} \quad s_{22} = (m_2 + sr)e^{-1} \bmod q = r_2$$

de unde putem afla cheia privată  $s$  astfel:

$$r_1 r_2^{-1} = (m_1 + sr)(m_2 + sr)^{-1} \bmod q \Rightarrow s = (m_2 r_1 - m_1 r_2)[r(r_2 - r_1)]^{-1} \bmod q$$

În continuare vom lucra în **PARI/GP**, prin urmare transformăm mai întâi toate valorile de care avem nevoie din baza 16 în baza 10. O metodă de a face acest lucru poate fi următoarea:

```
gp> n=length(w);
gp> for(i=1,n,if(w[i]==A,w[i]=10,if(w[i]==B,w[i]=11,if(w[i]==C,w[i]=12,
    if(w[i]==D,w[i]=13,if(w[i]==E,w[i]=14,if(w[i]==F,w[i]=15))))));
gp> W=sum(i=1,n,16^(i-1)*w[n+1-i]);
```

Aflăm acum, în ipoteza că s-a folosit aceeași cheie efemeră  $e$ , cheia privată  $s$ :

```
gp> q=26959946667150639794667015087019640346510327083120074548994958668279;
gp> m1=1268638092138210163260758055822429538066610350339;
gp> m2=229934186335685840756719395324394646288453721002;
gp> r=18187250800097972010521080073937585100154901858571130778437166133474;
gp> r1=20042106687643588872389242180506526832832251371631259823173622191288;
gp> r2=22255471905305126694378074733040389009439136736542793238977855911906;
gp> s=((m2*r1-m1*r2)%q)*(bezout((r*(r2-r1))%q,q)[1])%q
15010575815029851772642085218329323233091815558722670713086641180071
```

Verificăm că aceasta este corectă, adică vrem să vedem dacă într-adevăr are loc egalitatea  $V = sG$ . Pentru aceasta inițializăm curba eliptică  $\mathcal{E}$  peste care vrem să lucrăm, iar apoi calculăm punctul  $sG$ :

```
gp> p=2695994666715063979466701508701963067363714442254057248109931527511;
gp> E=ellinit([0,0,0,0,5]*Mod(1,p));
gp> xG=16983810465656793445178183341822322175883642221536626637512293983324;
gp> yG=13272896753306862154536785447615077600479862871316829862783613755813;
gp> G=[xG,yG];
gp> ellpow(E,G,s);
```

Obținem că:

$$x_{sG} = 14091661710852556870833728605751404033863675975464814254659297347139$$

$$y_{sG} = 9333722541138719487032926806284603775374491724501611657294489976354$$

Aceste valori sunt egale cu  $x_V$ , respectiv  $y_V$ , prin urmare, cheia privată  $s$  pe care am găsit-o este bună.

Deoarece problema cerea cheia privată  $s$  în format hexa, facem în final și transformarea numărului  $s$  din baza 10 în baza 16:



```

gp> v=vector(60);
gp> v[1]=divrem(s,16)[1];
gp> for(i=2,60,v[i]=divrem(v[i-1],16)[1]);
gp> w=vector(60);
gp> w[1]=divrem(s,16)[2];
gp> for(i=2,60,w[i]=divrem(v[i-1],16)[2]);
gp> S=vector(60,i,w[61-i]);
gp> for(i=1,60,if(S[i]==10,S[i]=A,if(S[i]==11,S[i]=B,if(S[i]==12,S[i]=C,
    if(S[i]==13,S[i]=D,if(S[i]==14,S[i]=E,if(S[i]==15,S[i]=F))))));
Obținem că S=8E88B0433C87D1269173487795C81553AD819A1123AE54854B3C0DA7.

```





# Capitolul 31

## Probleme de sinteză

### 31.1 Enunțuri

1. Completați: Scopul cifrării este de a asigura ..... unei comunicații.
  - (a) autenticitatea
  - (b) confidentialitatea
  - (c) integritatea
  - (d) nerepudierea
2. Următorul text a fost obținut utilizând sistemul de cifrare Cezar (au fost eliminate accentele, spațiile și semnele de punctuație): MHPEUDVVHPRQULYDOPDLVFHVWS-RXUOHWRXIIHU. Care este decriptarea sa?
  - (a) Chacun semble des yeux approuver mon courroux.
  - (b) Ma bouche mille fois lui jura le contraire.
  - (c) J'embrasse mon rival mais c'est pour l'étouffer.
  - (d) De grâce, apprenez-moi, Seigneur, mes attentats.
3. Cifrați textul "Attaque à l'aube " cu ajutorul algoritmului de substituție precizat mai jos.

A	B	C	D	E	F	G	H	I	J	K	L	M
J	G	F	K	P	R	M	T	S	V	Z	D	Q

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

Care este textul cifrat obținut?

- (a) JOOJCXPJDJXGP
  - (b) SHHSMYVSWSYVPV
  - (c) JOOJCXPJBJXGP
  - (d) SHHSMYVSZSYVPV
4. Cifrul Vigenère reprezintă o modalitate de cifrare îmbunătățită a sistemelor de cifrare cu substituție simplă. În ce constă acesta?
- (a) în aplicarea succesivă a mai multor substituții alfabetice pe același text.
  - (b) în aplicarea de substituții alfabetice care nu cifrează niciodată o literă în ea însăși.
  - (c) în cifrarea literelor care apar cel mai frecvent (cum ar fi e) în mai multe simboluri diferite.
  - (d) în alegerea mai multor alfabete de substituție independente și schimbarea alfabetului folosit, la fiecare literă, în mod ciclic.
5. Reprezentarea în baza 2 a numărului 1729 este:
- (a) 10010110100
  - (b) 11011000001
  - (c) 11001100011
  - (d) 6C1
6. Propunem următorul algoritm de cifrare: Alice și Bob doresc să schimbe un mesaj  $m$  care reprezintă un număr întreg între 0 și  $N - 1$ . Pentru aceasta, ei partajează o cheie secretă comună  $k$  extrasă aleator între 0 și  $N - 1$ . Mesajul cifrat se obține ca  $c = m + k \bmod N$ . Ce părere aveți despre securitatea sistemului?
- (a) Proastă: sistemul reprezintă o variantă a sistemului lui Cezar.
  - (b) Bună, dacă adversarul nu cunoaște algoritmul de cifrare.
  - (c) Foarte bună, cu condiția să nu utilizeze cheia  $k$  decât o singură dată.
  - (d) Excelentă: sistemul reprezintă o variantă a algoritmului RSA.
7. Alice îi trimite lui Bob un mesaj cifrat  $c$  obținut cu ajutorul algoritmului precedent. Cum determină Bob mesajul original  $m$ ?
- (a)  $m = c + k \bmod N$
  - (b)  $m = c - k \bmod N$
  - (c)  $m = c \times k \bmod N$
  - (d)  $m = c^k \bmod N$

8. Care dintre acronimele următoare desemnează un algoritm de cifrare de tip bloc?
- (a) AES
  - (b) HMAC
  - (c) SHA-1
  - (d) NIST
9. Inversul lui 17 modulo 100:
- (a) este 83.
  - (b) este 53.
  - (c) este  $1/17$ .
  - (d) nu există.
10. Am în posesia mea un mesaj  $m$  pe care nu vreau încă să îl divulg, dar doresc să pot dovedi peste câțiva ani că îl cunoșteam deja în 2010 (conform ampretei de timp). Pentru aceasta, este suficient să public astăzi:
- (a) un text cifrat corespunzător lui  $m$  cu o cheie cunoscută numai de mine.
  - (b) un text cifrat corespunzător lui  $m$  cu o cheie cunoscută de toată lumea.
  - (c) imaginea lui  $m$  printr-o funcție de dispersie (funcție hash).
  - (d) imaginea lui  $m$  printr-un MAC folosind o cheie aleatoare.
11. Funcția de dispersie (hash) SHA-512 întoarce valori între 0 și  $2^{512} - 1$ . Se calculează imagini prin această funcție în mod aleator. Care este ordinul de mărime al numerelor pentru care trebuie calculate valorile prin aceasta funcție pentru a găsi 2 valori care să aibă primii 20 de biți egali?
- (a) 20
  - (b) 1000
  - (c) 1000000
  - (d)  $2^{512}$
12. Construim un generator de numere pseudo-aleatoare care inițializează cu  $x_0$  cu o valoare între 0 și 999 și determină  $x_{n+1} = 500x_n + 789 \bmod 1000$ . În ce condiții ați utiliza acest generator?
- (a) Pentru a produce numere aleatoare între 0 și 999, dacă nu prezintă interes nivelul de securitate.
  - (b) Pentru generarea unei chei de tip *one-time pad*.
  - (c) Pentru construcția unei funcții de dispersie.

- (d) Niciodată.
13. Cum este obținută cheia secretă necesară pentru criptarea comunicației, la conectarea la un site web securizat?
- (a) Se obține din parola introdusă pentru conectare, printr-un algoritm de derivare a cheii precum PBKDF (Password Based Key Derivation Function).
  - (b) Provine din cheia publică a serverului, conținută într-un certificat.
  - (c) Provine din cheia privată a serverului, divulgată clientului după stabilirea conexiunii.
  - (d) Se obține în urma unui schimb de chei între client și server, precum schimbul de chei Diffie-Hellman.
14. Care este dificultatea de a factoriza un număr prim pe 1024 de biți astăzi?
- (a) Este simplu!
  - (b) Numărul poate fi factorizat cu ajutorul a câteva mii de calculatoare actuale care să ruleze între 1 și 2 ani.
  - (c) Nimeni nu poate face asta momentan, dar poate se va reuși de către agenții precum NSA.
  - (d) Acest lucru nu va fi posibil timp de mai multe milenii.
15. Algoritmul RSA (fără padding) este un algoritm de cifrare:
- (a) simetric, tip bloc.
  - (b) simetric, tip fluid (debit).
  - (c) parțial homomorfic.
  - (d) bazat pe identitate.
16. Fie generatorul Geffe descris de trei registre de deplasare **LFSR**<sub>*i*</sub> (ale căror polinoame de feedback sunt primitive de grad 19, 21 și respectiv 24) iar ieșirea de formula:  $y(t) = a_1(t) \cdot a_3(t) \oplus \bar{a}_1(t) \cdot a_2(t)$ . Care este complexitatea **LC** și perioada **P** a acestui generator?

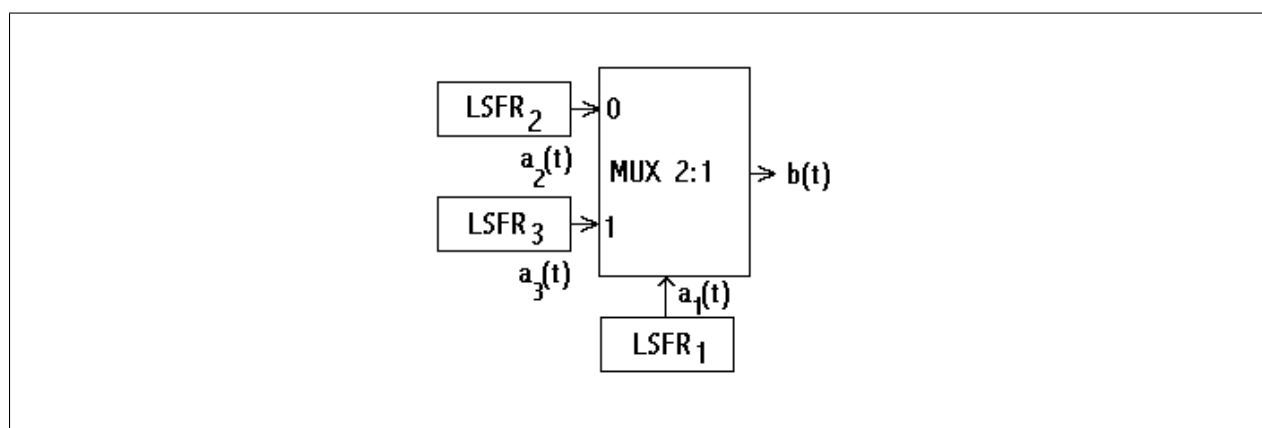


Figura 31.1: Generatorul Geffe.

- (a)  $LC = 640$ ,  $P = 2^{64}$ .
- (b)  $LC = 64$ ,  $P = (2^{19} - 1)(2^{21} - 1)(2^{24} - 1)$ .
- (c)  $LC = 876$ ,  $P = (2^{19} - 1)(2^{21} - 1)(2^{24} - 1)$ .
- (d) Niciunul dintre răspunsuri nu este corect.
17. Fie secvența dată de reprezentarea binară (scrisă pe 8 biți) a numărului  $i$ ,  $i = 0, \dots, 255$  :
- 00000000 00000001 00000010 00000011 00000100 ... 11111111
- Care este statistica testului frecvenței aplicată acestei secvențe binare? Este secvența aleatoare, relativ la testul frecvenței, la riscul de ordinul 1 de 5%?
- (a)  $f_{tf} = 256$ , șirul nu este aleatoriu.
- (b)  $f_{tf} = 1$ , șirul este aleatoriu.
- (c)  $f_{tf} = 0$ , șirul este aleatoriu.
- (d) niciunul dintre răspunsuri nu este corect.
18. Care dintre următoarele afirmații sunt adevărate:
- (a) Atac reușit asupra a două preimagini ale unei funcții hash implică reușita atacului de generare de coliziuni.
- (b) Atac reușit de generare de coliziuni asupra unei funcții hash implică reușita atacului asupra a două preimagini a aceleiași funcții hash.
19. Care dintre următoarele afirmații sunt adevărate:
- (a) Un registru de deplasare de lungime  $n$  are perioada de  $2^n - 1$ .
- (b) Un registru de deplasare de lungime  $n$  are perioada maximă de  $2^n - 1$ .

- (c) Un registru de deplasare de lungime  $n$ , cu polinomul caracteristic primitiv, are perioada de  $2^n - 1$ .
20. Probabilitatea de coliziune a două mesaje de lungime  $n$  biți procesate de aceeași funcție hash ideală, ce are ieșirea pe  $m$  biți, este:
- (a)  $2^{-m}$ .
  - (b)  $2^{-n}$ .
  - (c)  $2^{-mn}$ .
  - (d)  $2^{m-n}$ .
  - (e)  $2^{n-m}$ .
  - (f) Niciuna din valorile de mai sus.
21. Fie extensia Galois  $\text{GF}(3^2)$  generată de rădăcina polinomului  $X^2 - X - 1$ . În această extensie valoarea  $\log_{2\alpha+1}(1 + \alpha)$  este:
- (a) 8.
  - (b) 4.
  - (c) 2.
  - (d) 5.
  - (e) 6.
  - (f) Niciuna din valorile de mai sus.
22. Simbolul lui Jacobi  $\left(\frac{6278}{9975}\right)$  este:
- (a)  $-1$ .
  - (b) 0.
  - (c) 1.
  - (d) Niciuna din valorile de mai sus.
23. În cadrul unui acțiuni judiciare urmează a fi desemnat unul dintre cei doi judecătorii de serviciu. Deoarece niciunul dintre cei doi nu dorește să facă acest lucru în mod benevol, se propune modalitatea de decizie bazată pe rezultatul obținut din aruncarea unei monede. Astfel, judecătorul  $A$  alege "stema" sau "banul" iar judecătorul  $B$  aruncă moneda, decizia fiind luată în urma rezultatului obținut. Având în vedere faptul că  $A$  și  $B$  în locații fizice diferite se propune, de către criptograf, următorul protocol.
- PASUL 1.** Participantul  $A$  alege  $x = 0$  ("stema") sau  $x = 1$  ("banul") și o cheie aleatoare  $k$ . Se cifrează cu ajutorul algoritmului DES valoarea  $x$ :  $y = \text{DES}(x; k)$ .

**PASUL 2.**  $A$  transmite  $y$  către  $B$ .

**PASUL 3.**  $B$  aruncă o monedă și comunică lui  $A$  rezultatul obținut.

**PASUL 4.**  $A$  comunică lui  $B$  cheia  $k$ .

**PASUL 5.**  $B$  descifrează  $y$ , cu ajutorul algoritmului DES și obține ceea ce a ales  $A$ . Criptograful afirmă faptul că ”participantul  $A$  nu își poate schimba opțiunea” datorită valorii transmise  $y$ . Arătați următoarele:

- a) Utilizând ”birthday attack” utilizatorul  $A$  poate trișa;
- b) Care este complexitatea atacului de la punctul a)?
- c) Care este cerința primitivei criptografice ce asigură valabilitatea afirmației ”participantul  $A$  nu își poate schimba opțiunea”;
- d) Corectați protocolul astfel încât să nu mai fie posibil atacul de la punctul a).

24. Fie  $p$  un număr prim și  $G$  mulțimea tuturor elementelor  $x \in \mathbf{Z}_{p^2}$  care satisfac relația  $x \equiv 1 \pmod{p}$ . Arătați faptul că:

- a)  $G$  este grup multiplicativ;
- b)  $|G| = p$ ;
- c)  $L : G \rightarrow \mathbf{Z}_p$  definit de  $L(x) = (x - 1)p^{-1} \pmod{p}$  este un izomorfism de grupuri;
- d)  $p + 1$  este un generator al lui  $G$  și izomorfismul este logaritmul în baza  $p + 1$  a lui  $G$ . Cu alte cuvinte avem:  $(p + 1)^{L(x)} \pmod{p^2} \equiv x$  pentru orice  $x$ .

25. Să considerăm algoritmul de semnare DSS cu parametrii  $p, q, g$ , o funcție hash  $H$  și o cheie secretă  $x$ . În cadrul implementării se precalculează perechea  $(k, r)$  ce satisface relația  $r = (g^k \pmod{p}) \pmod{q}$ , aceasta fiind utilizată pentru generarea semnăturilor. Recuperați cheia privată de semnare.

26. Protocolul Wired Equivalent Privacy (WEP) utilizat în standardul IEEE 802.11 este utilizat pentru a proteja datele în cadrul transmisiilor wireless. Protocolul WEP are o cheie  $K$  de 40 de biți, partajată între entitățile ce comunică și este utilizată pentru protecția fiecărui ”frame”<sup>1</sup> transmis. În cadrul acestui exercițiu vom presupune faptul că cheia  $K$  este fixă și nu își schimbă valoarea. Pentru ca utilizatorul  $A$  să transmită un ”frame” la  $B$  va proceda după cum urmează:

**PASUL 1.** Codificarea CRC: Dându-se un mesaj de  $n$ -biți  $M$  ( $n$  este constant),  $A$  calculează o sumă de control de 32 de biți  $L(M)$ , unde  $L$  este o funcție liniară<sup>2</sup> ce nu depinde de  $K$ . Textul clar, de lungime  $(n + 32)$  biți, este  $P = M || L(M)$ .

**PASUL 2.**  $A$  cifrează  $P$  cu algoritmului RC4, cheia  $K$  și vectorul  $IV$  de 24 de biți specific fiecărui ”frame” transmis. Textul cifrat va fi  $C = P \oplus RC4(IV, K)$ .

<sup>1</sup>pachet de date.

<sup>2</sup> $L(X \oplus Y) = L(X) \oplus L(Y)$ .



**PASUL 3.**  $A$  transmite pe canalul radio  $(IV, C)$  către  $B$ .

*Întrebări:*

- a) Anunți producători specifică faptul că protocolul WEP are o securitate de  $40+24=64$  biți de cheie. Ce părere aveți de acest fapt. Justificați răspunsul.
- b) Care este modalitatea prin care  $B$  extrage mesajul original  $M$ ?
- c) În cadrul unor implementări, vectorul  $IV$  de 24 de biți, este ales aleatoriu la fiecare "frame" transmis. Arătați că acest lucru conduce la probleme de securitate atunci când traficul de date este mare. Propuneți o modalitate de remediere a problemei apărute.
- d) Să examinăm o altă problemă de securitate a protocolului WEP. Vom presupune faptul că atacatorul interceptează datele  $(IV, C)$  transmise de  $A$ . Arătați faptul că adversarul, chiar dacă nu cunoaște cheia  $K$ , poate calcula ușor un text cifrat  $C^*$  ( $C^* \neq C$ ) și retransmite  $(IV, C^*)$  fără ca  $B$  să poată detecta acest lucru. Câte posibilități de alegere avem pentru  $C^*$ ? Ce proprietate a securității este violată?

27. Descifrați, cu ajutorul algoritmului RSA-CRT, indicând semnificațiile elementelor algoritmului, mesajul:

$C = 9686\ 9613\ 7546\ 2206\ 1477\ 1409\ 2225\ 4355\ 8829\ 0575\ 9991\ 1245\ 7431\ 9874\ 6951\ 2093\ 0816\ 2982\ 2514\ 5708\ 3569\ 3147\ 6622\ 8839\ 8962\ 8013\ 3919\ 9055\ 1829\ 9451\ 5781\ 5154.$

Textul clar este în limba engleză.

Parametrii algoritmului sunt următorii:

- a) exponentul de cifrare este  $e = 9007$ ,
- b)  $p = 3490\ 5295\ 1084\ 7650\ 9491\ 4784\ 9619\ 9038\ 9813\ 3417\ 7646\ 3849\ 3387\ 8439\ 9082\ 0577$ ,
- c)  $q = 0003\ 2769\ 1329\ 9326\ 6709\ 5499\ 6198\ 8190\ 8344\ 6141\ 3177\ 6429\ 6799\ 2942\ 5397\ 9828\ 8533$ .

28. Fie numerele prime  $q = 7541$  și  $p = 2q + 1$ . Fie  $\alpha = 604$  și  $\beta = 3791$ .

- a) Arătați că  $\text{ord}(\alpha) = \text{ord}(\beta) = q$  în  $\mathbb{Z}_q$ . Mai mult, arătați că  $\alpha$  și  $\beta$  generează același subgrup  $G$  în  $\mathbb{Z}_p^*$ .
- b) Definim funcția hash  $h : \mathbb{Z}_q \times \mathbb{Z}_q \rightarrow G$  prin  $h(x_1, x_2) = x_1^\alpha x_2^\beta$ . Calculați  $h(7431, 5564)$  și  $h(1459, 954)$ .
- c) La punctul precedent ați obținut o coliziune pentru  $h$ . Folosiți-o pentru a calcula logaritmul discret  $d\log_\alpha \beta$ .
- d) Folosind logaritmul discret calculat, determinați și alte coliziuni pentru  $h$ .

## 31.2 Răspunsuri

1. *Răspuns:* (b). Pentru autenticitate, se folosesc MAC sau semnăturile electronice. Pentru integritate, în funcție de nivelul de exigență, se pot utiliza sume de control, funcții hash, MAC, etc.
2. *Răspuns:* (c). Vă puteți ajuta de poziția literelor dublate. Întrebare suplimentară: de unde provin aceste versuri?
3. *Răspuns:* (a). Literele de pe a doua linie sunt imaginile celor din prima linie, și nu invers.
4. *Răspuns:* (d). Metoda (a) este doar o substituție normală (compunerea a 2 permutări este tot o permutare). Metoda (b) este mai slabă decât prima întrucât expune mai multe informații despre textul clar. Metoda (c) se numește substituție polialfabetică.
5. *Răspuns:* (b). Este de ajuns să se calculeze restul împărțirii lui 1729 la 4 pentru a elimina (a) și (c). (d) este 1729 în hexazecimal (i.e. în baza 16).
6. *Răspuns:* (c). Algoritmul este o variantă a *one-time pad*. Oferă securitate perfectă dacă nu se utilizează cheia de criptare decât o singură dată. Poate fi de asemenea considerat o variantă a cifrului lui Cezar, dar aplicat unei singure litere și cu un decalaj ales aleator. Utilizat în acest fel, cifrul lui Cezar ar fi sigur. Sistemul nu are nicio legătură cu RSA. Răspunsul (b) nu ar satisface principiul lui Kerckhoff: un sistem de criptare trebuie să rămână sigur când adversarul cunoaște tot despre acesta, mai puțin cheia utilizată.
7. *Răspuns:* (b). Operația inversă adunării cu  $k \bmod N$  este scăderea cu  $k \bmod N$ .
8. *Răspuns:* (a). HMAC este MAC, SHA-1 este o funcție de dispersie și NIST este o agenție americană de standardizare.
9. *Răspuns:* (b).  $53 \times 17 = 1 \bmod 100$
10. *Răspuns:* (c). La momentul divulgării mesajului, toată lumea va putea verifica faptul că hash-ul este corect și că se cunoștea mesajul  $m$  la momentul calculării acestui hash. Metoda nu permite dezvăluirea mesajului  $m$ .  
O cifrare a lui  $m$  cu o cheie cunoscută doar de cel care face criptarea nu garantează nimic: se poate de asemenea publica un cuvânt aleator pentru ca ulterior să se aleagă cheia care să corespundă unei criptări corecte. Aceeași problemă apare în cazul MAC. O cheie cunoscută de toată lumea ar conduce la determinare textului clar  $m$ , ceea ce ar fi echivalent cu divulgarea mesajului  $m$ .
11. *Răspuns:* (b). Conform paradoxului nașterilor, pentru obținerea unei coliziuni pe primii 20 de biți ai funcției de dispersie, este necesar să se calculeze valoare funcției hash pentru  $\sqrt{2^{20}}$ , adică aproximativ 1000 numere.

12. *Răspuns:* (d). Valoarea lui  $x_n$  este constantă, egală cu 289, începând cu al treilea termen. Deci nu este vorba despre apariții aleatoare.
13. *Răspuns:* (d). Cheia de sesiune este determinată printr-un schimb de chei.
14. *Răspuns:* (a). Factorizarea unui număr prim este imediată.
15. *Răspuns:* (c). Proprietatea de homomorfism este aceea că cifrarea RSA a produsului a 2 mesaje (modulo  $N$ ) este produsul cifrărilor corespunzătoare celor 2 numere. Restul variantelor sunt eronate, fiindcă RSA este un cifru cu cheie publică, deci asimetric.
16. *Răspuns:* (c). Se aplică proprietățile generatorului Geffe.
17. *Răspuns:* (c). În această situație secvența supusă testării este ideală, numărul de biți de 0 este egal cu numărul de biți de 1 și anume 1024.
18. *Răspuns:* (a).
19. *Răspuns:* (b), (c). Un registru de deplasare de lungime  $n$  are  $2^n - 1$  stări posibile (starea nulă este exclusă). În situația în care polinomul caracteristic este primitiv atunci el generează toate stările posibile.
20. *Răspuns:* (a). Numărul de ieșiri posibile, ale unei funcții hash ideale cu ieșirea pe  $m$  biți, este  $2^m$ .
21. *Răspuns:* (e).
22. *Răspuns:* (a).
23. *Răspuns:* a)  $A$  va determina două chei  $k$  și  $k^*$  astfel încât:  

$$DES("banul"; k) = DES("stema", k^*).$$
 Pentru acest lucru procedează după cum urmează:  
 i)  $A$  va construi două liste  $(DES("banul"; k), k)$  și  $(DES("stema"; k^*), k^*)$ , pentru toate cheile  $k$  respectiv  $k^*$ . Listele sunt sortate în raport cu primul câmp al fiecărei intrări (i.e.  $DES("banul"; k)$  respectiv  $DES("stema"; k^*)$ ).  
 ii)  $A$  va căuta coliziuni în cadrul acestor liste și va obține  $k, k^*$  astfel încât:  

$$DES("banul"; k) = DES("stema"; k^*).$$
 iii) După ce se aruncă moneda  $A$  comunică lui  $B$  cheia  $k$  sau  $k^*$  după caz.  
 b) Complexitatea atacului anterior este reprezentată de căutarea coliziunilor în cadrul celor două liste, pe 64 de biți,  $DES("banul"; k)$  și  $DES("stema"; k^*)$ . Conform "birthday attack" este nevoie numai de  $2^{32}$  evaluări ale algoritmului DES pentru a determina o coliziune.

c) Cerința primitivei criptografice este ca funcțiile:

$$k \rightarrow DES(\text{"banul"}; k) \text{ și } k \rightarrow DES(\text{"stema"}; k)$$

să fie rezistente la coliziuni.

d) Se poate utiliza un algoritm de cifrare bloc pe 128 de biți, spre exemplu AES (în acest caz "birthday attack" are nevoie de  $2^{64}$  evaluări ale AES). Ca o alternativă se poate utiliza o funcție hash  $h$  rezistentă la coliziuni. Participantul  $A$  alege  $x \in \{\text{"stema"}, \text{"banul"}\}$ , o valoare aleatoare  $r$  și calculează  $y = h(x||r)$ . După ce  $B$  face alegerea,  $A$  poate dezvălui  $x$  și  $r$ .

24. *Răspuns:* a) Vom arăta faptul că  $G = \{x \in \mathbf{Z}_{p^2} | x \equiv 1 \pmod{p}\}$  în raport cu multiplicarea modul  $p^2$  este grup. Pentru aceasta se vor verifica următoarele: operația este parte stabilă, asociativitatea, elementul neutru și elementul simetrizabil.

b) Orice element  $a$  din  $\mathbf{Z}^{p^2}$  se poate scrie în mod unic  $a = a_1 + a_2 p$ , unde  $a_1$  și  $a_2$  sunt numere întregi ce satisfac relația  $0 \leq a_1, a_2 \leq p-1$ . Orice element  $a$  din  $\mathbf{Z}^{p^2}$  este în  $G$  dacă și numai dacă elementul corespunzător  $a_1$  este egal cu 1, de aici rezultă faptul că  $|G| = p$ .

c) Fie  $a = 1 + kp$ ,  $0 \leq k < p$  și  $b = 1 + lp$ ,  $0 \leq l < p$  elemente din  $G$ . Se verifică faptul că  $L$  este homomorfism:  $L(a \cdot b) = k + l \pmod{p}$  și  $L(a) + L(b) = k + l \pmod{p}$ , deci  $L(a \cdot b) = L(a) + L(b)$ . Direct se verifică injectivitatea și surjectivitatea lui  $L$ , deci  $L$  este izomorfism de grupuri.

d) Avem de arătat faptul că orice element  $a \in G$  poate fi scris ca o putere a lui  $p+1$ . Din binomul lui Newton rezultă:

$$(p+1)^2 \pmod{p^2} = \sum_{i=0}^n \binom{n}{i} p^i \pmod{p^2} = 1 + np.$$

Deci,  $p+1$  generează  $G$ . Pentru orice  $y \in G$  avem:  $y = \log_{p+1}(x)$  dacă și numai dacă  $x = (p+1)^y \pmod{p^2}$ .

Deoarece  $(p+1)^y \pmod{p^2} = 1 + py$ , obținem:

$$y = \frac{x-1}{p} \pmod{p} = L(x).$$

Această funcție logaritm stă la baza algoritmului criptografic Okamoto-Uchiyama.

25. *Răspuns:* Să considerăm semnăturile pentru mesajele  $m$  și  $m^*$ . Semnăturile sunt  $(r, s)$  și  $(r, s^*)$ . Avem:

$$s = \frac{H(m) + xr}{k} \pmod{q}$$

$$s^* = \frac{H(m^*) + xr}{k} \bmod q.$$

Deducem

$$k = \frac{H(m) - H(m^*)}{s - s^*} \bmod q.$$

Vom calcula apoi  $r = (g^k \bmod p) \bmod q$  și în final vom recupera  $x$  prin formula:

$$x = \frac{ks - H(m)}{r} \bmod q.$$

26. *Răspuns:* a) Nu este corect să se calculeze dimensiunea cheii prin sumarea dimensiunii celor două intrări în algoritm deoarece numai una este secretă. Deci dimensiunea cheii este de 40 de biți nu de 64 de biți.

b) Mai întâi  $B$  reconstruiește textul clar  $P^* = C \oplus RC4(IV, K)$ . Ulterior  $P^*$  este împărțit în două părți  $P^* = M^* || Q^*$ , unde  $M^*$  este de  $n$  biți iar  $Q^*$  de 32 de biți.  $B$  calculează  $L(M^*)$  și compară cu  $Q^*$ .  $B$  acceptă mesajul  $M^*$  dacă și numai dacă  $L(M^*) = Q$ , altfel va respinge mesajul  $M^*$ .

c) Conform "birthday paradox" alegând  $IV$  aleatoriu la fiecare "frame" rezultă că la fiecare  $2^{\frac{24}{2}} \approx 5000$  "frame"-uri există o coliziune pentru două  $IV$  din cele 5000 transmise de la/către același utilizator. În această situație avem o coliziune în șirurile cheie, ceea ce poate conduce la informație despre textul clar ([9]). O alternativă este de a incrementa  $IV$ .

d) Fie  $M^* = M \oplus \Delta$  un nou mesaj, unde  $\Delta$  este un șir de  $n$  biți. Vom calcula diferența dintre noul text cifrat  $C^*$  și  $C$ :

$$\begin{aligned} C^* \oplus C &= (P^* \oplus RC4(IV, K)) \oplus (P \oplus RC4(IV, K)) \\ &= P^* \oplus P \\ &= (M \oplus M^*) || (L(M) \oplus L(M^*)) \\ &= \Delta \oplus L(\Delta). \end{aligned}$$

Deci, pentru orice  $\Delta$  nenul, adversarul cunoaște faptul că  $C^* = C \oplus (\Delta || L(\Delta))$  care verifică CRC-ul. În concluzie acesta are  $(2^n - 1)$  posibilități de alegere pentru  $\Delta$  (și  $C^*$ ). Proprietatea violată este cea de *integritate a mesajului*. O concluzie ce se desprinde din acest exercițiu este aceea că CRC-urile (cu sau fără cheie) ne asigură protecția contra erorilor de transmisie nu și împotriva unui adversar malițios.

27. *Răspuns:* Prin calcule directe vom obține:  $d = e^{-1} = 0001\ 0669\ 8614\ 3685\ 7802\ 4442\ 8687\ 7132\ 8920\ 1547\ 8070\ 9906\ 6339\ 3786\ 2801\ 2262\ 2449\ 6631\ 0631\ 2591\ 1774\ 4708\ 7334\ 0168\ 5974\ 6230\ 6553\ 9685\ 4451\ 3277\ 1090\ 5360\ 6095 \bmod (p-1)(q-1)$ .

Apoi, prin calcul direct sau utilizând CRT:

$M = C^d = 20\ 0805\ 0013\ 0107\ 0903\ 0023\ 1518\ 0419\ 0001\ 1805\ 0019\ 1721\ 0501\ 1309$   
 $1908\ 0015\ 1919\ 0906\ 1801\ 0705 \bmod N, N = p \cdot q.$

Folosind codificarea spațiu = 00, A = 01, B = 02, ..., Z = 26 obținem textul clar:

"THE MAGIC WORDS ARE SQUEAMISH OSSIFRAGE".



# Bibliografie

- [1] **A. Atanasiu**, *Securitatea Informației, vol. 1, Criptografie*, ed. InfoData, Cluj, 2008.
- [2] **A. Atanasiu**, *Securitatea Informației, vol. 2, Protocoale de securitate*, ed. InfoData, Cluj, 2009.
- [3] **T. Baignères, P. Junod, Y. Lu, J. Monnerat, S. Vaudenay**, *A Classical Introduction to Cryptography Exercise Book*, Springer, ISBN 978-0-387-27934-3, 2006.
- [4] **A.J. Menezes**, *Handbook of Applied Cryptography*, CRC Press, 1997.
- [5] **E. Simion și Gh. Oprea**, *Elemente de Cercetări Operaționale și Criptologie*, Politehnica Press, ISBN 973-8449-006, 2002.
- [6] **E. Simion, V. Preda și A. Popescu**, *Criptanaliza. Rezultate și Tehnici Matematice*, Ed. Univ. Buc., ISBN 973575975-6, 2004.
- [7] **E. Simion**, *Enciclopedie Matematică*, Ediție coordonată de M. Iosifescu, O. Stănășilă și D. Ștefănoiu, Editura AGIR, ISBN 978-973-720-288-8, pp. 905-944, 2010.
- [8] **B. Schneier**, *Applied Cryptography*, Adison-Wesley, 1998.
- [9] **S. Vaudenay**, *A Classical Introduction to Cryptography: Applications for Communications Security*, Springer-Verlag, 2005.