Lucrările Seminarului de
CREATIVITATE MATEMATICĂ
Vol. 11(2002), 147 - 154

CRIPTAREA UNUI FIŞIER CU CHEIE SECRETĂ ÎN VISUAL BASIC 6.0

Marieta GÂTA

1. Noţiuni generale.

a. Criptare

Ştiinţa care studiază metodele de criptare şi decriptare se numeşte criptologie. Ea are în compunere criptografia, cea care se ocupă cu mijloacele de cifrare/descifrare legale şi criptoanaliza care studiază metode de spargere a cifrurilor, adică de determinare a mesajelor în clar sau a cheilor din criptograme. Datele criptate se numesc cipher iar împreună cu datele necriptate constituie criptograma. Cuvântul "criptografie" vine de la grecul "krypte", care înseamnă "ascuns" şi de la grecul "grafik" care înseamnă "scriere". Practic, orice cipher este rezultatul aplicării unui algoritm care este invariabil şi a unei chei care este variabilă. Puterea de criptare este dată de timpul şi de resursele necesare decriptării textului criptat fără să se ştie cheia de criptare. Teoretic nici un algoritm de criptare nu este perfect, decriptarea datelor fiind doar o chestiune de timp, implicit de viteza de procesare a datelor, uneori. Posibilitatea "spargerii" codului creşte cu înaintarea în timp (datorita apariţiei calculatoarelor din ce în ce mai performante şi implicit a folosirii unor chei de dimensiuni mai mari) şi cu apariţia unor criptoanalişti din ce în ce mai bine determinaţi. Mult timp eforturile criptografilor au fost dirijate spre întărirea cifrurilor prin complicarea algoritmului combinând substituţii şi transpoziţii asupra unor simboluri sau asupra unor blocuri (grupe de simboluri).

Dezvoltarea metodelor criptografice a fost mult influenţată de apariţia internetului, dorinţa utilizatorilor acestuia fiind păstrarea siguranţei asupra poştei electronice private, a transferului electronic de documente şi a altor aplicaţii. Datorită aplicaţiilor ei în securitatea Internet-ului, criptografia a devenit azi unul dintre cele mai dinamice domenii de cercetare ştiinţifică academică, ea fiind mult timp ascunsă sub "voalul" utilizărilor militare şi diplomatice. Peste 99% din aplicaţiile criptografice utilizate în lume nu protejează secrete militare ci se întâlnesc în bănci, taxe de drum, acces la terminale, contoare de electricitate, carduri etc. Rolul criptografici în aceste aplicaţii este de a împiedica furtul datelor importante. În cele mai multe dintre aceste aplicaţii nu s-a utilizat eficient criptografia, atacurile reuşite neavând nimic în comun cu criptoanaliza. Chiar şi NSA (National Security Agency) a admis că marea parte a erorilor, apărute în activităţile sale, provin din erorile de implementare şi nu din algoritmi sau protocoale. În aceste condiţii nu contează cât de bună a fost criptografia, atacurile criptoanalitice reuşite anulând acest lucru şi speculând erorile de implementare.

b. Sisteme criptografice simetrice şi cu chei publice

Un algoritm criptografic este o funcţie matematică folosită în procesul de criptare şi decriptare. Algoritmul criptografic lucrează cu o cheie (care poate fi un cuvânt, un număr, o frază) pentru a cripta textul. Acelaşi algoritm criptează diferit cu diferite chei. Securitatea datelor criptate depinde de două lucruri: puterea algoritmului şi secretul cu care este ţinută cheia.

Algoritmii criptografici (cifruri) permit o transformare a datelor numită cifrare sau criptare. Intrarea unui cifru este textul-clar iar ieşirea este textul-cifrat. La recepţie textul-clar se obţine printr-un proces de decriptare care primeşte la intrare textul-cifrat. Procesele descrise sunt controlate de chei care reprezintă un parametru al transformării. Un sistem criptografic (criptosistem) se compune din: M-text (mesaj) clar; C-text (mesaj) cifrat sau criptogramă; două funcţii inverse $E(\ )$ şi $D(\ )$; un algoritm care produce cheile de cifrare Ke şi de descifrare Kd astfel încât: mesajul cifrat C se obţine din mesajul clar M, folosind funcţia de cifrare E şi cheia de cifrare Ke: C=EKe(M) şi mesajul clar se obţine din criptograma C folosind funcţia de descifrare D şi cheia de descifrare Kd: M=DKd(C).

Există două tipuri de sisteme criptografice:

- simetrice (cu cheie secretă) care folosesc aceeaşi cheie atât la cifrare cât şi la descifrarea mesajelor, adică Ke=Kd.
- asimetrice (cu cheie publică) care folosesc chei distincte de cifrare şi descifrare (legate una de alta). Una din chei Kd este ţinută secretă şi este cunoscută doar de proprietarul ei. A doua cheie (perechea ei) Ke, este făcută publică.

c. Cifruri simetrice (cu cheie secretă)

Aceste sisteme de criptare folosesc aceeaşi cheie K atât la criptarea cât şi la decriptarea mesajelor. Cheia este ţinută secretă şi este folosită în comun de către emiţător (cel care cifrează mesajul M) şi de către receptor (cel care descifrează criptograma C). Sistemele simetrice sunt bine cunoscute, conduc la performanţe bune şi sunt folosite pentru protecţia datelor utilizatorilor. Avantajul acestui tip de criptare este cel ar rapidităţii iar dintre dezavantaje cel al transmiterii cheii secrete către receptor ar fi demn de remarcat. Se pot aminti aici criptosisteme simetrice cunoscute, cum ar fi standardul american de cifrare DES (Data Encryption Standard) sau sistemul IDEA (International Data Encryption Algorithm). Securitatea criptării simetrice depinde de protecţia cheii, administrarea acestora este un factor vital în securitatea datelor şi cuprinde următoarele aspecte: generarea cheilor, distribuţia cheilor şi memorarea cheilor. Problema fundamentală este cea a găsirii unor modalităţi de distribuţie sigură, periodică, a cheilor criptografice.

d. Cifruri asimetrice (cu chei publice)

Problema distribuirii cheii secrete, este rezolvată de acest tip de criptare. Sistemele criptografice având chei publice reprezintă un moment crucial în evoluţia criptografiei moderne. Whitfield Diffie şi Martin Hellman, de la Universitatea Stanford din California, au pus bazele unui principiu diferit de cel al cifrării

simetrice (1976). (Există însă dovezi ca acest tip de criptare a fost descoperit, înaintea celor doi de către Serviciul Secret Britanic, dar a fost ținut secret și nu s-a implementat). În locul unei singure chei secrete criptografia asimetrică folosește două chei diferite una pentru cifrare alta pentru descifrare. Una dintre chei -cheia privată (PRIV)- este ținută secretă și este cunoscută doar de proprietarul ei. A doua cheie (perechea ei, dedusă matematic printr-o funcție greu inversabilă din prima) -cheie publică (PUB)- este făcută publică. Ambele chei sunt de fapt niște fișiere, furnizate de un program capabil să genereze aceste perechi. Dacă cheia publică se poate face cunoscută oricui, cea privată se va păstra în siguranță. Pentru a se asigura confidențialitatea unui mesaj, datele sunt cifrate la emisie cu cheia publică a receptorului. Ele pot fi descifrate doar de către destinatarul autentic cu cheia privată pe care o deține doar el. Dacă se dorește semnarea digitală (electronică) a datelor în vederea verificării autenticității, datele sunt cifrate cu cheia privată a emițătorului iar verificarea poate fi făcută numai cu cheia sa publică, deci semnătura nu poate fi falsificată, întrucât doar emitentul autentic este în posesia cheii sale private. Algoritmi care folosesc astfel de chei sunt: Elgamal (numit după inventatorul său Taher Elgamal), RSA (inventatori: Ron Rivest, Adi Shamir și Leonard Adleman), Diffie-Hellman, DSA - Digital Signature Algorithm (inventat de David Kravitz). În practică, de cele mai multe ori, sunt folosiți în combinație atât algoritmii simetrici cât și cei nesimetrici.

2. Program de criptare a unui fișier realizat în limbajul Visual Basic 6.0 folosind tehnica criptării unui sistem cu cheie secretă.

a. Concepte algoritmice ale programului

Realizarea programului a fost făcută în Visual Basic 6.0, prezentarea lui fiind făcută în continuare. Programul folosește o tehnică de cifrare cu cheie secretă în locul celei cu cheie publică care este mai complexă. În vederea criptării unui fișier ce va fi transmis apoi prin intermediul e-mail-ului, cei doi utilizatori – expeditorul și destinatarul – vor cădea de acord asupra unei parole care va avea o lungime nu foarte de mare. Aceasta parolă, care în cazul programului va fi reprezentată printr-un șir, va reprezenta cheia secretă. Ea va fi convertită printr-un algoritm de distribuire într-o configurație de 3 octeți. Convertirea prin distribuire a unui șir se face printr-un calcul în sens unic, asemănător unei sume de control. Acest calcul poate fi repetabil, dar nu și ușor reversibil. Aceeași parolă poate conduce prin distribuire la același rezultat, dar având rezultatul operației de distribuire nu se va putea determina ușor parola. Octeții pseudoaleatori vor fi generați folosind generatorul de numere aleatoare al Visual Basic-ului. Tehnica folosită va adăuga un octet ce va conține 8 caractere generate aleator, de salt, în antet și apoi distribuirea acestei combinații de caractere cu parola, generând astfel restul caracterelor din antet. Cei 2 octeți din antet (16 caractere) vor putea fi folosiți pentru a verifica rapid dacă parola introdusă este corectă, la decriptarea fișierului. Și tot aceste caractere din antet, în număr de 16, vor fi apoi utilizate pentru a cripta fișierul, fișierul rezultat prin aceasta metodă de criptare fiind de fiecare dată altul.

Fișierele folosite de program sunt clasa Cifru.cls și formele View.frm și

Secret.frm. Funcțiile, procedurile, metodele și evenimentele formei Secret: Amestec (funcție), Criptare (procedură), Decriptare (procedură), File_Change (eveniment), Iesire_Click (eveniment), View_Click (eveniment), Criptare_Click (eveniment), Decriptare_Click (eveniment), Browse_Click (eveniment), iar ale clasei Cifru: Inițializare (procedură), Contractă (procedură), Dilată (procedură), DoXor (metodă), Let Text (proprietate), Get Text (proprietate), KeyString (proprietate). Transformarea unui șir într-o secvență repetabilă dar aleatoare, imprevizibilă, de 8 caractere s-a făcut cu ajutorul unei funcții numită Amestec. Această funcție va returna o valoare care va fi utilizată pentru a verifica parola introdusă de utilizatorul ce dorește decriptarea fișierului.

S-a folosit o clasă Cifru care a definit un obiect care va cripta un singur șir la un moment dat. Acest obiect are o proprietate write-only numită KeyString care va configura șirul folosit pentru a defini o cheie unică pentru criptare și decriptare. După ce va fi configurată această proprietate, generatorul de numere aleatoare al Visual Basic-ului va fi configurat ca valoarea de start a lui să fie unică, bazată pe caracterele din cheie și pe ordinea în care acestea apar în cheie.
Funcția rnd a Visual Basic-ului se va folosi cu un parametru negativ pentru a inițializa generatorul de numere aleatoare astfel încât să repete secvența dată. Apoi se va apela instrucțiunea Randomize cu o valoare.

O altă proprietate numită Text va memora textul ce se va cripta/decripta. Se va introduce în această proprietate un text ce va putea fi citit, se vor apela metodele pentru criptare și compactare a șirului și se va salva valoarea rezultată a acestei proprietăți într-un fișier read-only, unde el va apărea într-o formă codificată, imposibil de descifrat. Apoi se va introduce în această proprietate șirul criptat din fișier, se va configura proprietatea KeyString la valoarea folosită pentru criptarea șirului, se vor apela metodele pentru decompactare și decriptare a șirului și se va afișa apoi conținutul proprietății text.

Metoda numită DoXor va acționa asupra șirului ce este conținut de proprietatea text și va aplica un operator Sau-Exclusiv pe fiecare octet al șirului cu următorul octet pseudoaleator din secvența ce a fost definită de proprietatea Keystring. Acest proces este unul reversibil, operația ce va folosi aceeași secvență de octeți pseudoaleatori va reface șirul la valoare lui inițială.

Metoda numită Dilata, convertește orice șir, la un șir ce poate fi afișat pe ecran sau tipărit la imprimantă. Astfel, orice șir, va fi convertit la unul ce va fi mai lung, dar în care toate caracterele sunt tipăribile și afișabile pe ecran. Astfel dacă s-ar folosi un Sau-Exclusiv, care s-ar aplica asupra octeților unui șir ce ar avea Blank-uri, Tab-uri sau caractere grafice neobișnuite, unii octeți ar putea avea valoarea Tab, sau Blank, sau caractere grafice. Metoda aceasta ia biții de la fiecare grup de trei caractere și formează al patrulea caracter, apoi, cele patru caractere sunt transferate într-un domeniu de caractere care sunt toate tipăribile și afișabile pe ecran.

Metoda numită Contracta va face operația inversă asupra șirului modificat de metoda Dilata, și anume va converti șirul care conține numai caractere ce se pot tipări și afișa pe ecran într-un șir ce poate conține oricare din cele 256 caractere

ASCII. Algoritmul folosit a evitat folosirea caracterului spațiu în transferarea caracterelor din șirul extins.

Codul sursă din Visual Basic al procedurilor Criptare, DoXor, Dilata și al funcției Amestec este prezentat în continuare, iar procedurile Decriptare și funcția Contracta se realizează exact invers.

```
Sub Criptare()
Dim sh As String
Dim sn As String
Dim s1 As String
Dim Cifru2 As New Cifru
Dim n As Long
Open txtFile.Text For Binary As #1
s1 = Space$(LOF(1))
Get #1, , s1
Close #1
sn = Amestec(Date)
sh="[Marieta]" & sn & Amestec (sn & txtPassword1.Text)
Cifru2.KeyString = sh
Cifru2.Text = s1
Cifru2.DoXor
Cifru2.Dilata
s1 = Cifru2.Text
Open txtFile.Text For Output As #1
Print #1, sh
n = 1
Do
Print #1, Mid(s1, n, 70)
n = n + 70
Loop Until n > Len(s1)
Close #1
End Sub

Function Amestec (s1 As String) As String
Dim Cifru1 As New Cifru
Cifru1.KeyString = s1 & "abcdef"
Cifru1.Text = s1 & "abcdef"
Cifru1.DoXor
Cifru1.Dilata
Cifru1.KeyString = Cifru1.Text
Cifru1.Text = "abcdef"
Cifru1.DoXor
Cifru1.Dilata
Amestec = Cifru1.Text
End Function

Public Sub Dilata()
Dim ic As Integer
Dim li As Long
Dim lj As Long
Dim ik As Integer
Dim la As Long
Dim sb As String
la = Len(msText)
sb = Space(la + (la + 2) \ 3)
For li = 1 To la
ic = Asc(Mid(msText, li, 1))
lj = lj + 1
Mid(sb, lj, 1)=Chr((ic And 63) +59)
Select Case li Mod 3
Case 1
ik = ik Or ((ic \ 64) * 16)
Case 2
ik = ik Or ((ic \ 64) * 4)
Case 0
ik = ik Or (ic \ 64)
lj = lj + 1
Mid(sb, lj, 1) = Chr(ik + 59)
ik = 0
End Select
Next li
If la Mod 3 Then
lj = lj + 1
Mid(sb, lj, 1) = Chr(ik + 59)
End If
msText = sb
End Sub
Public Sub DoXor()
Dim ic As Integer
Dim ih As Integer
Dim li As Long
For li = 1 To Len(msText)
ic = Asc(Mid(msText, li, 1))
ih = Int(Rnd * 256)
Mid(msText, li, 1) = Chr(ic Xor ih)
Next li
End Sub
```

b. Mod de utilizare al programului

Se va selecta un fișier de orice tip pentru a fi criptat cu ajutorul butonului **Alege fisierul...** La începutul fișierului se va insera un mic antet format din stringul "Marieta", care ne va permite să sesizăm dacă nu cumva fișierul selectat este deja criptat. Se va apăsa butonul **Criptare** dacă dorim să criptăm sau butonul **Decriptare** dacă dorim să decriptăm. Se va introduce parola. În cazul decriptării va

fi suficient să se introducă o singură dată parola dar în cazul criptării parola va fi obligatoriu a se introduce de două ori, pentru a nu se strecura erori la tastare. Dacă vor apărea erori la introducerea parolelor vom fi atenționați printr-un mesaj că am introdus parole diferite. Fișierul deja criptat va fi salvat în format ASCII folosind numai caractere ce pot fi tipărite, pentru a putea fi transmis prin e-mail, știut fiind că dacă fișierul ar conține date binare necorespunzătoare acestor formate vor apărea erori la transmitere. Dacă vom încerca să criptăm același fișier de mai multe ori, rezultatul va fi diferit de fiecare dată chiar dacă vom folosi aceeași parolă. Apăsând butonul **Ieșire** se va ieși din program.

c. Rezultate obținute din funcționarea programului

Fereastra principală a programului este prezentă în Figura 1. Se observă că butoanele **Vizualizare conținut fișier**, **Criptare** și **Decriptare** sunt inactive până la apăsarea butonului **Alege fișierul...**. Pentru criptare, după ce se va apăsa butonului **Criptare** acest buton devine inactiv. La operația inversă, aceea de decriptare, după apăsarea butonului **Decriptare** acesta devine și el inactiv. Interfața programului ce se profilează în această figură este aceea care apare după apăsarea butonului **Alege fișierul...** (va apare o fereastră în care se va alege fișierul ce va fi criptat). Prezentarea (în fereastra ce se deschide după apăsarea butonului **Vizualizare conținut fișier**) conținutului fișierului t1.txt ales înainte de a fi criptat s-a făcut în Figura 2, iar în Figura 3 este vizibil conținutul aceluiași fișier t1.txt după ce a fost criptat (apăsând pe butonul **Criptare**) sau conținutul aceluiași fișier obținut după decriptare (apăsând pe butonul **Decriptare**). Toate aceste fișiere pot fi vizualizate în mod Read-Only . Pentru alte doua fișiere (t2.txt și t3.txt) rezultatele vor fi cele așteptate și anume de fiecare dată la revenirea după decriptare (deci operațiile vor fi: criptare-decriptare) se va ajunge la același fișier de la care s-a plecat inițial. Dacă păstrăm fișierul dar modificăm parola vom obține un alt fișier criptat. Dimensiunile fișierelor t1.txt, t2.txt, t3.txt (folosite pentru a exemplifica funcționarea programului) înainte de criptare sunt: 5KB, 3KB, 2KB, iar dimensiunile acelorași trei fișiere după criptare sunt: 6KB, 4KB, 2KB, deci dimensiunea lor nu crește foarte mult dacă nu se criptează fișierul apăsând de mai multe ori pe butonul criptare, caz în care dimensiunea fișierului crește proporțional cu numărul de criptări succesive. Dacă dimensiunea inițială a fișierului este mare (de ordinul MB) atunci și dimensiunea fișierului criptat crește. De asemenea, timpul după care se va obține fișierul criptat e mai lung în cazul unui fișier de ordinul MB decât în cazul unui fișier de câțiva KB.

d. Aplicații practice ale programului

Aplicația se poate folosi cu succes în cazul mesajelor de poștă electronică, dar ea nu oferă un nivel ridicat de securitate neputând fi folosită în situații în care confidențialitatea e un factor important, de exemplu aplicațiile care transferă informații financiare sau alte informații particulare cu grad ridicat de confidențialitate, un hacker bine motivat putând să "spargă" cu ușurință mesajele.

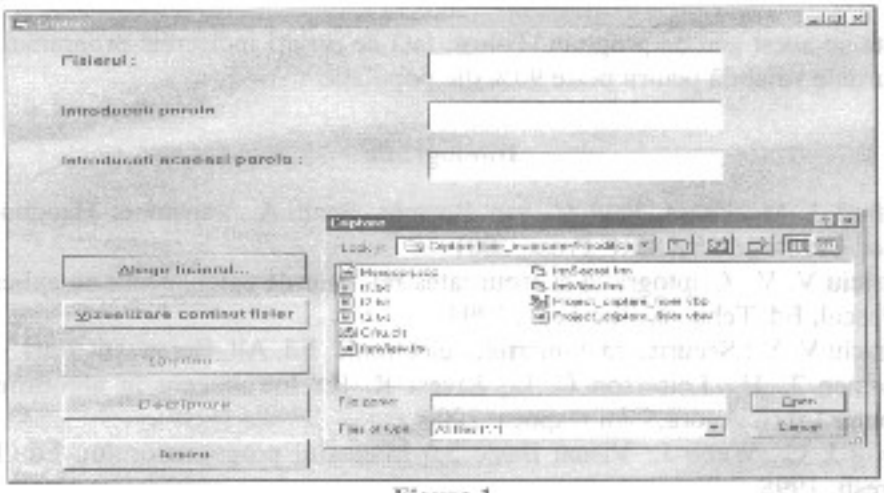

Figura 1

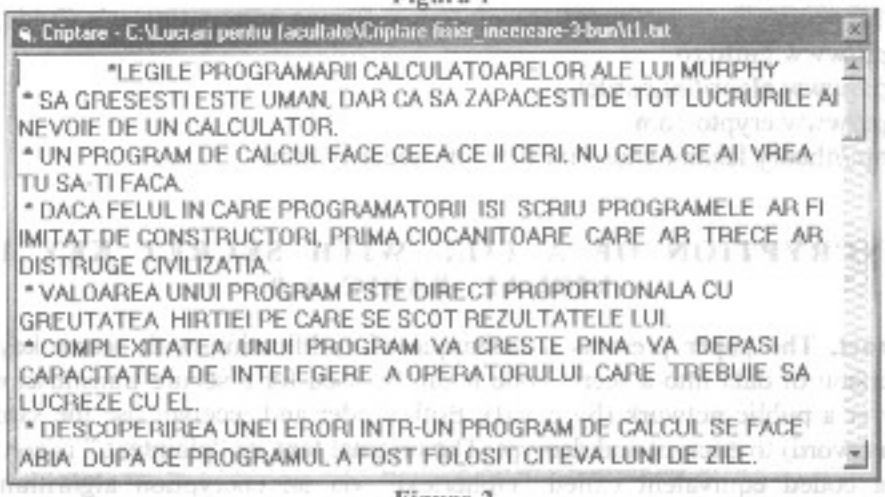

Figura 2

Figura 3

criptate cu acest gen de program. Totuşi, faţă de priviri indiscrete programul oferă o securitate valabilă pentru peste 90% din populaţie.

Bibliografie

1. Alfred J. Meckness, Paul C. van Borscht, Scott A. Vanstone: Handbook of Applied Cryptography, CRC Press, 2001
2. Patriciu V. V.: Criptografia şi securitatea reţelelor de calculatoare cu aplicaţii în C şi Pascal, Ed. Tehnică, Bucureşti, 1994
3. Patriciu V. V.: Securitatea comerţului electronic, Ed. All, Bucureşti, 2001
4. Cormen T. H., Leiserson C. E., Rivest R. R.: Introducere în algoritmi, Ed. Computer Libris Agora, Cluj Napoca, 2000
5. Craig J. C., Webb J.: Visual Basic 5.0 Manualul programatorului, Ed. Teora, Bucureşti, 1998
6.***Microsoft Corporation: Microsoft Visual Basic4.0 Programmer's Guide, 1995
7. http://www.ginfo.ro
8. http://www.algoritmi.pardel.ro
9. http://www.crypto.com
10. http://theory.lcs.mit.edu/~rivest/crypto-security.html

ENCRYPTION OF A FILE WITH SECRET KEY IN VISUAL BASIC 6.0

Abstract. The paper presents a technique of codification with secret key. The conversion of data into a secret code it can be used for a secure transmission of a file over a public network (by email). Both sender and receiver use the same key (or password) to encrypt and decrypt. The original text, or "plaintext," is converted into a coded equivalent called "ciphertext" via an encryption algorithm. The ciphertext is decoded (decrypted) at the receiving end and turned back into plaintext. The program used a modality of graphic presentation, through modeling and design. Also this program applies the techniques object oriented programming offered by Visual Basic.

Primit la redacţie: 27.09.2002

Marieta Gâta
Universitatea de Nord Baia Mare
Departamentul de Matematică şi Informatică
4800 Baia Mare, Str. Victoriei nr. 76
ROMANIA
E-mail: marietag@ubm.ro