

## Secure Distributed Databases Using Cryptography

Prof.dr. Ion IVAN, asist. Cristian Valeriu TOMA  
Catedra de Informatică Economică, A.S.E. București

*The computational encryption is used intensively by different databases management systems for ensuring privacy and integrity of information that are physically stored in files. Also, the information is sent over network and is replicated on different distributed systems. It is proved that a satisfying level of security is achieved if the rows and columns of tables are encrypted independently of table or computer that sustains the data. Also, it is very important that the SQL - Structured Query Language query requests and responses to be encrypted over the network connection between the client and databases server. All this techniques and methods must be implemented by the databases administrators, designer and developers in a consistent security policy.*

**Keywords:** databases, encryption, cryptography, digital signature.

### 1 Introducere

În securitatea sistemelor distribuite în general și în securitatea bazelor de date în particular se urmărește securizarea mesajelor și a tranzacțiilor ce se efectuează în sistem. Principalele obiective urmărite în securizarea unui sistem distribuit sunt:

1. *Confidențialitatea* – protejează conținutul mesajelor (tranzacțiilor) împotriva citirii neautorizate, de către alte persoane decât receptorii specificați de emițător.

2. *Autentificarea* – permite receptorului unui mesaj să determine în mod sigur identitatea expeditorului.

3. *Integritatea datelor* – în rețea furnizează receptorului unei tranzacții siguranța că mesajul primit este identic cu mesajul emis de expeditor.

4. Prevenirea nerecunoașterii mesajului (tranzacției) de către expeditor (*Non-Repudierea*) – garantează integritatea și originea mesajelor (tranzacțiilor) din punctul de vedere al expeditorului și nu al destinatarului. Se împiedică astfel ca expeditorul unei tranzacții electronice să nege trimiterea ei.

5. *Aplicarea selectivă a unor servicii* – de multe ori este necesară acoperirea unor părți ale mesajelor (tranzacțiilor), de exemplu cele conținând numărul cărții de credit al unui client. Aceasta nu trebuie să fie în clar vânzătorului, care poate abuza de utilizarea ei.

Criptografia este o “unealtă” folosită în realizarea securității sistemului (nu este singura, se adoptă pachete de măsuri grupate în politici de securitate pentru a asigura securitatea sistemului).

Există două tipuri de sisteme criptografice: simetrice și asimetrice. *Sistemele criptografice simetrice* (cu cheie secretă) folosesc aceeași cheie, atât la criptarea cât și la decriptarea mesajelor. *Sistemele criptografice asimetrice* (cu cheie publică) folosesc chei distincte la criptare și decriptare (dar legate una de alta). Una din chei este ținută secretă și este cunoscută doar de proprietarul ei. A doua cheie (perechea ei) este făcută publică. Algoritmi criptografici (cifrurile) folosiți în sisteme criptografice simetrice se împart în cifruri flux (stream ciphers) și cifruri bloc (block ciphers). Cifrurile flux pot cripta un singur bit de text clar la un moment dat, pe când cifrurile bloc criptează mai mulți biți (64 sau 128 de biți) la un moment dat.

### 2. Sisteme criptografice și modalități de criptare

În acest articol se operează o serie de concepte din criptografia computațională cum ar fi: funcțiile hash, algoritmi criptografici simetrici, asimetrice și modalități de criptare.

#### 2.1. Funcții hash

Funcțiile hash (de dispersie) joacă un rol important în autentificarea conținutului unui mesaj transmis în comunicații. Obiectivul lor NU este de a asigura secretul transmisiilor, ci de a

crea o valoare  $h=H(m)$ , numita rezumat (digest), cu rol în procedura de semnătură digitală, valoare  $h$  foarte greu de falsificat. În procedura de semnare digitală sunt implicate 3 entități:

- $M$  = mesajul de "digerat";
- $h = H(M)$  rezumatul calculat prin hash;
- $S = \text{Sign}(H(M))$  semnătura digitală.

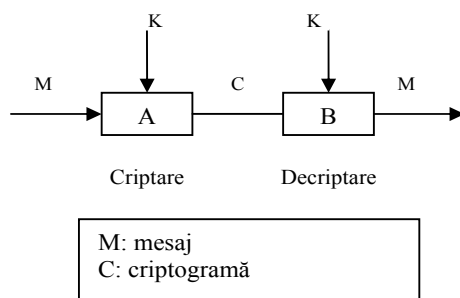
Funcțiile hash au câteva caracteristici comune:

- fiind dat  $M$ , este simplu să se calculeze  $h$ ;
- fiind dat  $h$ , este IMPOSIBIL să se calculeze  $M$ , astfel încât  $H(M) = h$ ;
- fiind dat  $M$ , este greu să se găsească alt mesaj  $M'$  (imposibil chiar), astfel încât  $H(M) = H(M')$ ;
- este imposibil să se găsească 2 mesaje aleatoare, astfel încât  $H(M) = H(M')$ , proprietate numită rezistență la coliziune.

Una din cerințele fundamentale pentru o astfel de funcție este ca, modificând un singur bit la intrare să producă o avalanșă de modificări în biții de la ieșire. Unele din cele mai utilizate funcții de dispersie sunt: **SHA-1**, **MD5**, MD2, RIMPED-160.

## 2.2. Algoritmi criptografici cu cheie simetrică

În cazul sistemelor criptografice simetrice (cu cheie secretă) se folosește aceeași cheie, atât la cifrarea cât și descifrarea mesajelor. Cheia este ținută secretă și folosită în comun de către emițător și receptor (figura 1). Sistemele simetrice sunt bine cunoscute, conduc la performanțe bune și sunt folosite pentru protecția datelor utilizatorilor. La baza cifrurilor simetrice stau cifrurile elementare bazate pe: transpoziție și substituție.



**Fig.1.** Cifrare simetrică.

Următorii algoritmi din „era 128 de biți” (și cheia și mesajul au minim 128 de biți) sunt cei mai cunoscuți: Rijndael (finalist și câștigător AES), Twofish (finalist AES), Serpent (finalist), RC6 – Rivest Cipher 6 (finalist), MARS (finalist), Blowfish, DEAL, SAFER+, FROG, CAST-256, Magenta, SkipJack, DFC – Decorrelated Fast Cipher.

## 2.3. Algoritmi criptografici cu cheie asimetrică

Cifrurile asimetrice (cu chei publice) folosesc chei distincte de cifrare și descifrare (dar dependente una de alta). Deoarece este imposibil deducerea unei chei din cealaltă, una din chei este făcută publică, fiind pusă la dispoziția oricui dorește să trimită un mesaj cifrat. Doar destinatarul, care deține cea de-a doua cheie, poate descifra și utiliza mesajul. Tehnica cheilor publice este folosită și în semnătura digitală (electronică), fapt ce a sporit popularitatea sistemelor criptografice cu chei publice. Modul de folosire a unui cripto-sistem asimetric este redat în figura 2.

În figura 2 sunt prezentate trei utilizări ale acestor tipuri de criptosisteme asimetrice. Exemplul 2 și 3 din figură detaliază procedura de semnătură digitală deoarece aceste exemple arată modul cum se execută o semnătură digitală asupra unui mesaj  $M$ . Exemplul 3 ilustrează procesul de semnătură electronică plus procesul confidențialitate.

## 2.4. Moduri de criptare

La nivelul modului de utilizare ai algoritmilor simetrici (indiferent de algoritmul folosit, iar uneori chiar și asimetrice) se întâlnesc în practică două tipuri de cifrări: cifrarea bloc (block ciphering) și cifrarea flux (secvențială) (stream ciphering).

*Cifrarea bloc* operează cu blocuri de date în clar și cifrate – de regulă 64 și 128 biți dar, uneori, și mai mari. Cele mai cunoscute moduri de acest tip sunt: ECB, CBC, PCBC, OFB/NLF. *Cifrarea secvențială* operează cu secvențe de date în clar și cifrate de mărime un bit sau octet dar, uneori și cu date de 32 de biți. Cele mai cunoscute moduri de acest tip sunt: cifrarea secvențială, cifrarea secvențială cu auto-sincronizare, cifrarea cu reacție, cifrarea sec-

vențială sincronă, cifrarea secvențială cu reacție la ieșire, cifrarea cu contor.

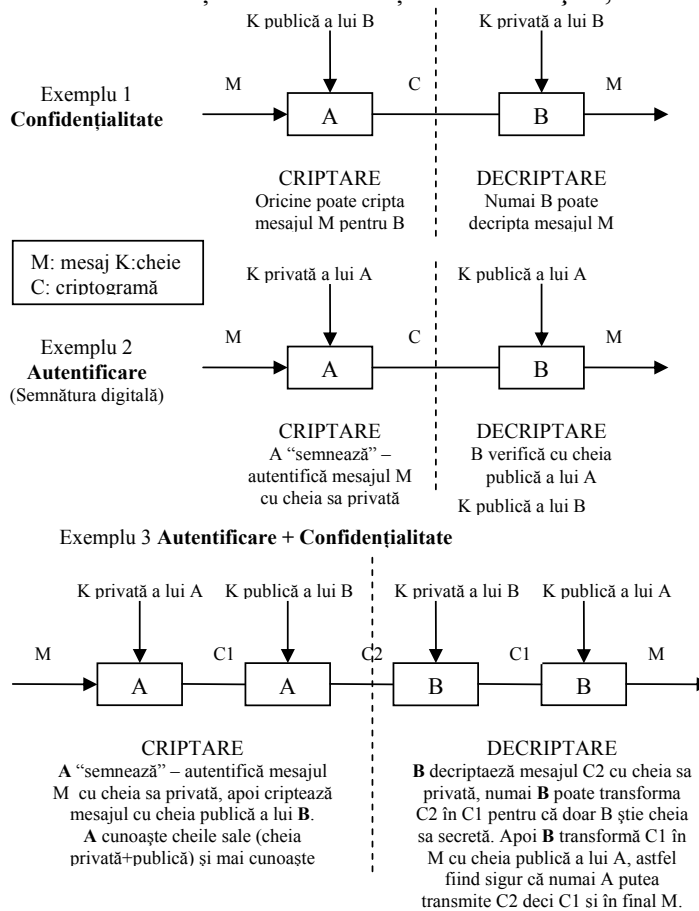


Fig.2. Sistemul de cifrare cu cheie publică folosit pentru confidențialitatea sau/și autentificare unui mesaj M între expeditorul A și destinatarul B.

### 3. Dimensiunile potențial securizabile ale bazelor de date

Există trei mari dimensiuni ce adeseori sunt analizate în ce privește criptarea în baze de date:

- Dimensiunea granularității datelor ce sunt criptate și decriptate. Câmpul, atributul sau tuplul sunt câteva alternative în ce privește criptarea datelor. In practică s-a dovedit că ce mai bună alegere este să fie criptat câmpul pentru că minimizează numărul de octeți ce trebuie criptați și decriptați.
- Dimensiunea implementării algoritmilor criptografici la nivel hardware versus software. Cele mai bune rezultate dar și cu cele mai mari investiții inițiale sunt cele furnizate de sisteme hibrid, apoi se consideră criptarea/decriptarea hardware și apoi cea software. Fiecare model, hibrid, hardware sau software are avantajele/dezavantajele lui, precum și costuri și performanțe operaționale diferite.

- Dimensiunea locației serviciului de criptare. Serviciul de criptare este fie local, fie apelabil prin proceduri la distanță (RPC – Remote Procedure Call), fie serviciu atașat rețelei. În mod practic se observă că dacă criptarea respectiv decriptarea are loc cât mai aproape de mediul unde sunt stocate datele, atunci gradul de securizare prin criptare crește. Criptarea executată de sistemele de gestiune a bazelor de date protejează datele ce aparțin tabelor din bazele de date existente și este o alegere a administratorilor, utilizatorilor și a dezvoltatorilor de aplicații de baze de date dacă trebuie criptată/decriptată și comunicația între baze de date și aplicații ce utilizează acele date.

### 4. Alegerea topologiei pentru implementarea mecanismelor criptografice

Se iau în considerare câteva combinații de criptare precum cea la nivel software și har-

dware și cele legate de granularitatea datelor. În practică analizele au fost întâi efectuate la nivel de câmp/atribut. S-au dezvoltat acceleratori de căutare peste câmpuri criptate, incluzând câmpurile cheie primară. La nivel hardware se preferă criptarea doar a cheilor de administrare și a stocării.

*Criptare/decriptare la nivel elementar software.* În testele efectuate pe bazele de date relaționale distribuite sau nu, s-au luat în considerare algoritmi criptografici cu cheie simetrică precum AES – Rijndael sau Blowfish și algoritmi cu cheie asimetrică precum RSA sau DSA. Testele au arătat ceea ce era de așteptat că algoritmi cu cheie simetrică sunt mult mai rapizi decât cei cu cheie asimetrică. Procedura este destul de simplă, se stochează algoritmul AES sau RSA ca fiind funcție definită de utilizator (stored procedure). Odată înregistrată această funcție poate fi apelată de un declanșator (trigger) ori de câte ori apare o comandă SQL de inserare date (INSERT) sau interogare date (SELECT).

*Criptare/decriptare la nivel hardware.* Unul din cele mai utilizate sisteme de criptare hardware este HSM FIPS-140-1. Cheia principală este creată și criptată de sistemul HSM. Ca și algoritm principal de criptare se utilizează sistemul criptografic cu cheie simetrică DES. Se preferă criptarea/decriptarea software pentru ca are cel mai bun raport calitate preț. Cel mai utilizat mod de criptare a datelor și structurilor de date în SGBD-uri este CBC.

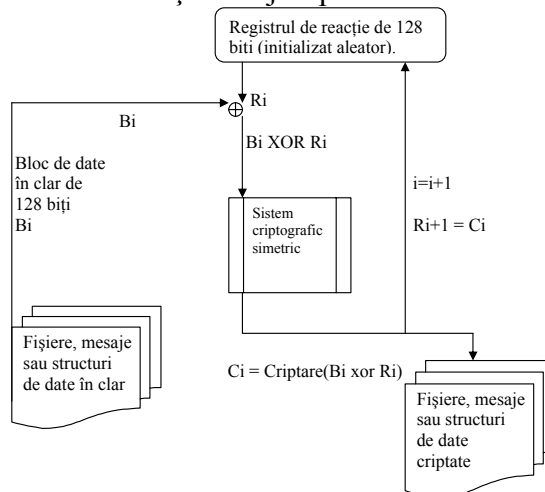
*Cifrarea bloc cu înlanțuire CBC* (Cipher Block Chaining) - Acest mod de cifrare bloc, adaugă mecanismului de criptare un bloc cu reacție. Rezultatul criptării unui bloc anterior revine prin buclă și intervine în criptarea blocului curent.

În acest mod, datele cifrate nu mai depind doar de datele în clar, ci și de modul de cifrare a blocului anterior. În CBC, datele în clar, înainte de a intra în blocul decriptare propriu-zis, sunt însumate modulo 2 (XOR) cu blocul de date cifrat anterior. Figura 3 reprezintă modul de criptare CBC.

Pașii pe scurt sunt următorii: se inițializează registrul de reacție cu o funcție hash de dispersie MD5, ce produce rezumatul unei paro-

le. Apoi pentru  $i$  (un contor) de la 0 la numărul de blocuri a fișierului sau structurii de date se execută XOR (sau exclusiv) între blocul citit din fișier și blocul de date din registrul de reacție. Se scrie blocul criptat în fișier și apoi se atribuie registrului de reacție blocul de biți criptat. Se incrementează  $i$  și apoi se reia procesul. Aceasta e o structură de date compusă ce implică două structuri de date de tip fișier și o structură de tip masiv sau după necesități poate fi una dinamică (listă de liste de octeți).

CBC face ca același bloc de date să se transforme în blocuri de date diferite, deoarece la diferite rulări valoarea de inițializare a registrului de reacție poate fi diferită. Dacă valoarea inițială a registrului de reacție rămâne neschimbată între rulări, atunci două mesaje identice folosind aceeași cheie se vor transforma în același mesaj criptat.



**Fig.3.** Cifrare CBC.

Vectorul de inițializare (valoare inițială a registrului de reacție) nu trebuie neapărat să fie secret (deși se poate genera printr-o funcție hash în urma unei parole, astfel încât să nu fie nevoie de transmiterea – valorii inițiale – prin rețea la receptor).

Chiar dacă acest lucru pare greșit (de a nu ține secret valoarea inițială), nu este deoarece, oricum prin canal (rețea) circulă blocurile criptate dar nu și cheia deci, cineva ce ar dori să spargă cifrul va trebui să cunoască ce structură de date s-a folosit, ce algoritmi și protocolul de transmisie a datelor, și apoi să spargă algoritmul.

## 5. Concluzii

În acest articol au fost elaborate și analizate o serie de concepte prezente în securizarea bazelor de date prin utilizarea criptografiei computaționale. Acest domeniu este o mică parte din ce înseamnă securitatea la nivelul bazelor de date unde se aplică politici de securitate consistente la nivel aplicație, utilizator și administrator de baze de date.

S-a observat că în practică datorită comodității și lipsei de interes a unor dezvoltatori și administratori de baze de date și de aplicații ce utilizează baze de date, apar breșe de securitate ce sunt uneori vulnerabilități greu de acceptat în contextul unei cerințe cât mai pronunțate de produse program și servicii cu grad ridicat de valoare adăugată și de securitate. Modul prin care poate fi combătut acest lucru este conștientizarea pericolelor ce se pot produce la nivel informații precum și comunicarea și dezbateră la nivelul comunității dezvoltatorilor și utilizatorilor a acestor probleme.

## Bibliografie

- [1] J. He and M. Wang. Encryption in relational database management systems. In Proc. Fourteenth Annual IFIP WG 11.3 Working Conference on Database Security (DBSec'00), Schoorl, The Netherlands, 2000.
- [2] R. Agrawal, J. Kiernan, R. Srikant, and Y. Xu. An XPath-based preference language for P3P. In Proc. of the 12th Int'l World Wide Web Conference, Budapest, Hungary, May 2003.
- [3] S. Muchnick. Advanced Compiler Design and Implementation. Morgan Kaufmann Publishers, 1997.
- [4] N. R. Adam and J. C. Wortman. Security-control methods for statistical databases. ACM Computing Surveys, 21(4):515– 556, Dec. 1989.
- [5] R. Agrawal and J. Kiernan. Watermarking relational databases. In 28th Int'l Conference on Very Large Databases, Hong Kong, China, August 2002.
- [6] R. Agrawal, J. Kiernan, R. Srikant, and Y. Xu. Hippocratic databases. In Proc. of the 28th Int'l Conference on Very Large Databases, Hong Kong, China, August 2002.
- [7] Agrawal, J. Kiernan, R. Srikant, and Y. Xu. Implementing P3P using database technology. In Proc. of the 19th Int'l Conference on Data Engineering, Bangalore, India, March 2003.