ENCRYPTION ALGORITHMS FOR DATABASES

Doina FUSARU, Prof. Ph.D.*
Mariuța ȘERBAN, Assist. Prof. Ph.D. student*
*Faculty of Financial-Accounting Management
Spiru Haret University

Abstract

For most cases, people use an ecrypted mode when sending personal information to a server, via an electronic form. Whenever shopping is done online, the browser uses cryptographic methods to send to the server the credit card number and private information. Thanks to the surprising development of the Internet, and not to the structural models (OSI and TCP/IP) this technology is based on, the electronic commerce requires quality, security, reliability and, above all, the possibility of implementing all such concepts.

It is interesting that none of the widely used cryptographic systems is mathematically demonstrated to be safe. As a matter of fact, the entire technology of cryptography is based on mathematical problems that are still unanswered to.

Looking at the above, the study of the cryptographic and security methods, as well as finding strong crypto-systems is still a pivotal issue.

Key-words: *encryption algorithms, database security, cryptographically, cipher*

JEL Classification: L86

Introduction

Nowadays, the databases are essential constituents of the web applications, providing them the possibility to have a dynamic content. Due to the fact that the secret or confidential information is usually stored in a database, the protection of them must have a high priority.

In order to receive or send any information, one needs to connect to the database, to send a valid call, to receive the results and to close the connection.

One of the widely used query languages for such type of interaction is Structured Query Language (SQL).

For the database systems, data protection is twofold: security and integrity. Data security and integrity denotes the fact that the access to data requires a legit and controlled authorization, which is the task of the database administrator, via the Databases Administration Systems (D.A.S.). To this purpose, D.A.S. allows the authorization and control of access to data, the use of visions, special procedures, data encryption.

Encryption algorithms for databases

Cryptography is a set of standards and protocols for encoding the data and messages, so that they might be stored and sent, in a safer manner. This science underlies many Internet-based security services and mechanisms, by using mathematical methods to change the data, aiming to hide their content or to protect them against alteration. Cryptography provides help in having a safer communication, even when the transmission environment is not trustworthy. Likewise, it may be used to encrypting the sensitive files, so that the number of intruders is kept to a minimum. Also, it may contribute to providing data integrity, as well as keeping them secret. Cryptography assists in checking the data and messages source, by the digital signatures and certificates. When cryptographic methods are being used, the cryptographic keys must remain secret – the algorithms, the keys size and the files formats may be made public, without bringing any harm to the security issue. The modern cryptography has to include the following features:

Confidentiality – the guarantee that no one can read the message, except for the meant recipient.

Data integrity – it achieves data protection against alteration or unauthorized handling. Data handling means insertions, delays or substitutions.

Authentification – involves the possibility to identify the information source and the entity, where the latter may be a person, a computer terminal, a credit card.

Non-repudiation – prevents the negation of some previous actions.

In other words, cryptography need to appropriately account for directions, both theoretically and practically. It needs to prevent and detect stealing and other ilegal action, as it is one of the information security assurance techniques.

There are two types of *cryptographic* systems:

• *symmetric (secret key)* that use the same key, for both encrypting and decrypting the messages;

• *asymmetric (public key)* that use distinct keys for encrypting and decrypting (connected to each other).

From the algorithm perspective and of the field of use, cryptography may be divided into four cryptographic primitives:

- secret-key cryptographic algorithms;
- public-key cryptographic algorithms;
- digital signature;
- hash functions.

In order to build a cryptographic system that will solve the issues of the computer-based security, surely and efficiently, there is a need to use the cryptographic primitives as a group, upon requirements.

A *cryptosystem* is made of:

• M-clear text;

• C-ciphered text;

• 2 inverse functions, *E()* and *D();*

• an algorithm that generates the keys *Ke* and *K*i

Secret-key cryptographic algorithms (symmetrical)

To assure confidentiality of date stored in computers or transmitted via networks, *secret-key cryptographic algorithms (symmetric)* are being used. They are characterized by the fact that both users share the same key, both for encrypting and decrypting. The encrypting key should be kept secret from the unauthorized users, since the one who knows it will be able to access the secret information. The symmetric cryptographic algorithms have a high encrypting speed, compared to the asymmetric ones, and are very comfortable in encrypting large blocks of information.The security of such type of algorithm largely depends on the key length and the possibility of keeping it secret.The main issue that emerges during the attempt to create secret communications among the various users is the key management; for n users, n(n-1)/s bi-directional connections are possible (and we need the same number of keys). Generally speaking, this involves difficulty in generating, distributing and memorizing the keys. The use of the electronic computers has allowed the usage of larger keys, therefore increasing the resistance to cryptoanalitic attacks. When the secret key has a reasonable size and is frequently enough changed, it is practically impossible to break the code, even if the encrypting algorithm is known.

The symmetric encryption security greatly depends on the *encryption key protection*. As a result, their management is an essential factor and refers to:

– *keys generation*, i.e. the (pseudo)random means of creating the key octets (bits) succession;

– *keys distribution,* meaning the way how the keys are distributed and made known to all the users who have access to the encrypted information;

– *keys memorization,* i.e. their safe storage on a magnetic support or a card, usually encrypted under a different keys encryption key, called *master key.*

The fundamental issue of using the encryption in networks is the one to find a method of *safe* and regular *distribution* of the encryption keys, as it is necessary to change them as often as possible.The internet and other networks that involve working with databases are also using the network services, via specific protocols or public key systems, the so-called *digital envelopes.*

The most known symmetric encryption algorithms are:

1) block ciphers:

- DES (Data Encryption Standard), Triple DES;
- DEA (International Data Encryption Algorithm);
- AES (Advanced Encryption Standard).

2) sequential ciphers:

- RC4.

Public-key cryptographic algorithms (asymmetrical)

A new perspective upon the encryption systems is provided by the public-key cryptographic algorithms (asymmetrical). These algorithms are characterized by the fact that different keys are used during encrypting and decrypting, keys that are connected between them by a mathematical relation. This type of relation has such nature, that, whether a key is known, the other one is very difficult to find. Thus, if we encrypt with one of them, we will only be able to decrypt with the other one and vice versa.

Two directions of using the asymmetrical crypto-systems:

– ***confidentiality***, the public key will be made public and the person who wishes to send confidential data to the public key owner will only encrypt via this key, being aware that the owner solely can decrypt them;

– ***authentification of both the transmitter and data,*** where the transmitter encrypts the data via his secret key, and the person who wishes to authenticate the data will use the pair key for decryption (the public one).

Even if the asymmetric cryptosystem is strong enough, you need to have the key length of at least 2,304 bits, in order to provide a security level, comparable with the one from a 128 bit-key in the symmetric cryptosystem. The asymmetric cryptosystems are much slower in encrypting and decrypting and are not as good at encrypting large volumes of information, i.e. the computer-based databases. The symmetric cryptosystems are circa 1,000 times faster than the asymmetric ones, hence the latter ones are more often used for the basic purposes:

– ***key distribution***, used for the encrypting symmetric algorithms;

– ***the digital signature,*** an attribute of a user, for the purpose of his identification.

The most widely used asymmetric encryption systems are as below:

– RSA (Rivest-Shamir-Adleman);

– EG (El Gamal);

– ECC (Elliptical Curve Cryptography).

Hash functions

The hash function applies to a message of a certain length M and transmits a value of a fixed length h: h=H(M), where h has the m length. There are plenty of such functions, but the hash ones have also extra properties, which make them unidirectional:

- knowing M is easy to calculate h;
- knowing H and h it is difficult to calculate M, for which H(M) = h;
- Knowing M, is it difficult to find another M1 message, for which H(M)=H(M1).

The hash functions play an important role in the authentication of a message content, transmitted to the systems including databases. Their role is not to reveal the secret of transmission, but to create a value h=H(M), also called *digest,* essential in the digital signature procedure, very hard to forge. One of the fundamental requirements for such function is that, by changing a bit upon entrance, this will trigger an avalanche of changing bits at the exit.

There are more calculation plans for the digest of a message, where the most widely used are:

– MD5 – an algorithm that receives a message of an arbitrary length at the entrance and generates a digest of 128 bits at the exit;

– SHA1 – NIST, along with NSA designed an algorithm for the calculation of hash function, called *Secure Hash Algorithm (SHA),* where the standard is *SHS.* It is meant to be used together with the digital signature system, *DSS*. *SHA* generates a digest of 160 bits, higher than *MD5.*

Therefore, the issue of an efficient definition of a encryption infrastructure emerges.

The infrastructure of the hybrid encryption algorithms systems for databases

The definition of the infrastructure of the encryption algorithms systems is done via the exchange of confidential information by a vulnerable network, like Internet or Intranet. The information should not be read by anyone else than the recipient, who will be able to identify the information source, in order to detect a possible information alteration. The components required for the above are as follows:

- a symmetric encryption algorithm (for eg AES), to encrypt the information flow;
- the infrastructure of the keys in use (creation, organization, storage, distribution, maintenance):
- session keys, used by the symmetric encryption algorithms;
- terminal keys, used to encrypt the session keys (the infrastructure of the public keys is utilized – PKI);
- master keys, necessary for encrypting the terminal secret keys;
- asymmetric encryption algorithm (for changing the session keys);
- hash functions, to validate the integrity of data or to authenticate their content;
- a digital signature, to authenticate the data source.

The drafting of the hybrid encryption system of an instant files transfer into databases

The system above-mentioned refers to creating encrypted communication systems on Internet or Intranet, which allows the instant transfer of files and/or messages. This system may be implemented into various systems of databases, information-distributed. A proposition has been made to create a hybrid encryption system at the application level in the TCP/IP protocols series. The application defines a port for communication and uses the TCP protocol to transport data within the relations among the databases.

This encryption system is seen as a solution to the computer-based security issues of databases and may be done both for the calculation systems that are connected to the Internet by a constant IP address and for the ones where the IP is generated upon connection.

The hybrid encryption system includes the following components:

- the symmetric system of data flow encryption;
- the asymmetric algorithm for a regular change of the session keys;
- the digital signature to authenticate the entities during communication.

The general architecture of the system includes the following applications:

- a server application, with functions of generating, signing and administration of the certificates for each user;

o user applications, which communicate among them and with the server application.

The server application provides the following services:

o the generating of digital certificates for each user;

o keeping the record of the users who received certificates;

o the signing of the certificates, to check their validity;

o the control and follow-up on the certificates that have been voided or are expired.

The certificate that contains the secret key will be encrypted via a phrase introduced by the user, and the public-key certificate will be stored in a public database, so that each user has access to it.

The public key certificate has the following structure:

– the identification series of each certificate;

– user (personal data);

– the certificate expiration date;

– the public key of the user;

– the signature of the server application.

The certificate that keeps the secret key will have the same structure, except that this secret key is encrypted with a symmetric encryption algorithm (the hash function of the user phrase will be used as a key).

The database user application has the following possibilities:

1. Connecting to another user:

o at first, the user identity will be checked via the key phrase;

o the creation of a connection request to the desired application.

2. The request includes the following data:

o the request subject;

o the public key certificate;

o the session key, encrypted with the recipient public key;

o the source digital signature.

3. The acceptance of a connection from another user:

– the source authenticity will be checked;

– the session key will be extracted and decrypted.

4. The encryption/decryption of the data flow via the session key.

5. The loading of the public keys digital certificates from the server application.

In order to achieve a encryption system, the most efficient platforms need to be used, to create the applications: the *Java* platform at Sun or the *Microsoft .Net Framework* platform.

Three extensions are being suggested, as a integrant part in the SDK package, which gives a new perspective upon security. The three extensions are: JCE (*Java Cryptography Extension*), JSSE (*Java Secure Socket Extension*) and JAAS (*Java Authentication and Authorization Service).*

JCE represents the frame where the below are implemented:

– encryption algorithms, where the most known are DES, RC2, RC4, IDEA, 3DES, AES, RSA;

– algorithms to generate keys for the encryption algorithms;

– password encryption, PBE (Password Based Encryption).

JCE has the following services:

– key plants (keys may be generated for the algorithms types above-mentioned);

– the creation and management of the databases where the keys are stored;

– the building and management of the encryption algorithms parameters;

– certificates plants.

JSSE implements the protocols SSL V3 (*Secure Socket Layer*) and TLS 1.0 (*Transport Layer Security*). This extension also provides such a support for the HTTPS protocol and the RSA encryption algorithm.

The JAAS extension allows to the services on a server to authenticate themselves and provide the control upon the users of these services.

Microsoft suggests, in general, the same services via Microsoft CryptoAPI technology.

Future trends in the databases security

One of these trends is the *elliptical curves in Galois fields*. The complexity of solving an equation of an elliptical curve type is an issue much harder than the decomposition of a number in prime factors. The study is still in its first stages and a cipher using this scheme is expected soon.

Not long ago, the theoretical bases for the quantum computers have been drafted. They rely on the *polarization effect*. The measure unit of information for such systems is the *qbit*. In comparison to the classic bit, which may have only two values, these systems include a third one, depending on the system structure at one point. The great advantage of this system is that, once the information transmission starts between two terminal points, any attempt to capture the information without changing it is impossible, due to the polarization effect. This method operates perfectly in theory and hardware implementations are expected in the future.

Thanks to the surprising development of the Internet, and not to the structural models (OSI and TCP/IP) this technology is based on, the electronic commerce requires quality, security, reliability and, above all, the possibility of implementing all such concepts.

Consequently, the study of the encryption and security methods for the computer-based databases, as well as finding strong cryptosystems remains a permanent issue.

Conclusions

The computer-based security of databases is an issue that becomes more relevant and acute, along with the development of the calculation networks and systems industry. One of the basic methods in providing informational security is the encryption method. At the present moment, cryptography covers a set of protocols, encryption algorithms, infrastructures of handling the cryptographic keys, etc. In order to achieve a safe system of information protection, we need to foresee all the directions it may be attacked from, since it is useless to secure a side of the system when the attach might come from a more vulnerable point. A cryptographic system is effective when it maintains the balance between what is necessary and what is possible. To create such system, a good infrastructure is required, with the following components: symmetric cryptographic algorithms, asymmetric, of hash function, digital signature and an infrastructure of the necessary keys.

In practice, should the issue of implementing this system emerges, two ways can be taken: the selection of an existent system or the creation of a new one. Each of these two has its advantages and disadvantages. The existent solutions have been already studied by specialists and implemented, hence they are safer to use; the problem is that they cannot be included in our computer-based system all the time. Therefore, we need to create a system adjusted to our needs.

Such system is suggested for the secured data transfer among distributed information systems. This system is an effective solution for the companies that own a distributed information infrastructure. The advantages brought about are the flexibility, automatization at work and an increased security by implementing the encryption algorithms of the computer-based databases.

REFERENCES

- Scout Oaks, *Java Security*, O'Reilly USA, May, 2001.
- Roşca Ion Gh., *Comerţul electronic, concepte tehnologii şi aplicaţii,* Editura Economică, 2004.
- Barefoot Coy, *Revoluţia comerţului electronic*, Amaltea Publishing House, 2004.
- Menezes A., *Handbook of Applied Cryptography*, CRC Press Inc. Publishing House, 1997.
- Young Adam, *Malicious Cryptography*, John Wiley & Sons Inc. Publishing House, 2004.
- Patriciu Victor Valeriu, *Securitatea comerţului electronic*, All Publishing House, 2001.
- Mao Webno, *Modern Cryptography: Theory and Practice*, Prentice Hall Publishing House, 2003.
- Cobb Chey, *Cryptography for Dummies*, John Wiley & Sons Publishing House, Inc. 2004.
- Piper Fred, *Cryptography: A Very Short Introduction*, Oxford University Press Publishing House, 2005.
- MSDN Library Security.