IMPLEMENTING CRYPTOGRAPHY IN DATABASE

KWAN SHONG HANN

A thesis submitted in fulfillment of the
requirements for the award of the degree of
Bachelor of Computer Science
(Computer Systems & Networking)

Faculty of Computer Systems & Software Engineering
Universiti Malaysia Pahang

APRIL 2010

ABSTRACT

With the increasing dependency of database for data storage, many sensitive data such as personal information data and credit card information data are being stored inside a database. These data is so valuable that it attracts unauthorized personnel to gain access for obtaining the data for further usage. Even though the network infrastructure nowadays is protected with different kind of security measures, there is none of them is able to block all the threats perfectly. Therefore, the data needs to be able to protect its confidentiality even though all measures have been failed. This is where the importance of cryptography in protecting the data confidentiality even the data is in the hands of the attacker. Cryptography can be implemented in different kind of methods to secure the database. The purpose of this study is to develop a cryptosystem that is able to implement cryptography to the data before storing them into the database. This implementation has been strengthened by introducing two approaches which are the classification of types of data using Key Family and the classification of data encryption key status according to the activation time. Key Family separates types of data such as personal information data and credit card information data. Both families use different data encryption key to encrypt and decrypt to limit the access of the attacker if one of the key is obtained. The state of data encryption key is determined by the activation date which the key which has the latest time will be activated and the old ones will be expired. This will prevent the key remains too long for encryption and decryption which poses risks for threats to break the key. This system is developed using Java programming language with the use of Java Cryptography Extension for the cryptography process. MySQL database is used as the protected database where all the data in the database is encrypted for protection. Finally, it is hope that this system can provide better security for data confidentiality and also become the last line defense of data towards the attacks.

ABSTRAK

Pengkalan data pada hari ini semakin diperlukan untuk menjadi tempat penyimpanan data utama kepada perisian computer yang dibangunkan. Ini menyebabkan lebih banyak data yang sensitif seperti data peribadi individual dan data credit card. Data tersebut adalah amat berharga sehingga ia mendorongkan pencerobohan terhadap pengkalan data. Walaupun sistem rangkaian pada hari ini telah dilengkapi dengan pelbagai jenis cara untuk mencegah pencerobohan terhadap rangkaian, tiada satu cara yang dapat mencegahnya dengan sempurna. Oleh itu, data hendaklah diubahsuai supaya ia dapat mempertahankan informasi yang terkandung dalamnya. Ini dapat dicapai dengan menggunakan teknik Cryptography terhadap data. Tujuan penyelidikan ini adalah untuk membangunkan satu sistem akan mengapplikasikan cryptography terhadap data sebelum ia disimpan dalam pengkalan data. Applikasi ini telah diperkuatkan dengan penggunakan Key Family yang akan mengumpulkan jenis data yang sama menjadikan keluarga yang sama. Ini adalah untuk mengelakkan penceroboh dapat mendapatkan semua data dengan hanya mendapatkan satu kunci tersebut. Kunci yang digunakan untuk encrypt akan diperkenalkan dengan menggunakan masa activasi untuk mengubah penggunaan kunci dari semasa ke semasa. Kunci yang lama akan diganti dengan kunci yang mempunyai masa yang terbaru. Ini adalah untuk mengurangkan masa kunci yang digunakan kerana masa adalah satu element yang penting bagi penceroboh untuk mencapai kunci tersebut. Sistem ini dibangunkan dengan menggunakan bahasa pengaturcara Java dengan applikasi Java Cryptographty Extension yang digunakan untuk process encrypt dan decrypt. Akhir sekali, diharap sistem ini dapat memberikan pertahanan yang lebih baik kepada penyimpanan data di dalam pengkalan data dan ia dapat menjadi satu pertahanan yang terakhir kepada data terhadap ancaman-ancaman.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

This chapter will provide a brief overview of the entire project including the objective of the project, scope and problem

1.1 Introduction

Nowadays, security issues among server and databases becomes popular as the dependencies of human on computer devices has been increased rapidly to store data and company's files. Therefore, its security vulnerability has become a concern to the company or user.

To help in securing the content of the database, the introduction of encryption technology to database will be the last line defense for the data after the firewall and network protection has been broken in by unauthorized personnel. It helps to secure the content of the database safely by transforming text or files into meaningless characters which will avoid the third party to read or altar the information in the database.

Besides the attack from external user, there is a possibility of internal database attack happens as database administrator has the right to read or write the content in the database. To curb this problem, encryption technology will be introduced individually according to the user in the database. The user manages to select his/her encryption method to secure data in the database. The user has his/her own password to encrypt personal details in the database without the acknowledgement of third party.

With this, the data will be secured not only from external attack to the database; it will also protect the privacy of the user of the database from immoral database administrator who violates the rights of the user in database.

1.2 Problem Statement

Firstly, the existing plain text file in database will become the target of network culprits to attack the database for either personal usage or as a source to earn profit illegally. The content maybe deleted or edited by whoever make it through the firewalls and network protection.

Conventional database allows one encryption method for all the data contained inside the database. Once the only encryption method is solved by the hackers, all the data will be translated into meaningful data to be edited or removed.

Thirdly, in a shared database which contained different users, database administrator have access to data which stored by the users. There is a possibility the database administrator will explore the data personally without the permission from the files owner in the database.

1.3 Objective

i. To introduce encryption technology into database to help to transform personal data in database into meaningless character in the database. This will prevent unauthorized personnel from reading or altering the content of the database.

ii. The encryption method will be a standalone application which will be a middle tier application to encrypt the data. This will reduce the risk of having the same encryption at all the database. This will help to protect internal database administrator from violating the privacy of the users in the database.

iii. To limit database administrator's access to the users' private files. Database administrator can only access to the database management files and the users' personal details given by the users. This will prevent the database administrator from accessing the data without the permission from the user.

1.4 Scope

Basically, this system contains several scopes which it can carry out its operation.

i. This database encryption system will focus on one to one basis which means an application with a database that it connects to.

ii. Data type for encryption will be focus on plain text alphabet and numerical data.

iii. Type of application using this system is application that saves confidential data such as credit card numbers in the database.

CHAPTER 2

LITERATURE REVIEW

This chapter briefly explains about the studies of the background of the project to understand the technology and process involves in the project studied.

2.1 Introduction to Database

The introduction of databases in human is to help human to save the work of documentation and filing. A database offers a mass amount of space for us to save our data in a proper and neat method. Its designed is to offer an organized mechanism for storing, managing and retrieving information. Database uses table which consists of columns and rows just like how the Microsoft Excel spreadsheet. The column in the table hold attributes which will determine the value to be inserted into. For the row, it contains a single record which contributes values to the different columns in the table [1].

prodID	category	product	date
1	Helmets	Caberg Justissimo	2007-07-23
2	Helmets	Caberg rhyno	2007-07-03
3	Helmets	Nolan N102	2007-07-01
4	Helmets	Shoei Z-One Black	2007-07-04
5	Jackets	Clover Jacket Talon	2007-07-02
6	Jackets	Clover Jacket Hydro	2007-07-13
7	Jackets	Clover Jacket Askja	2007-07-08
8	Luggages	Givi Installation Plate XLV 650	2007-07-11
9	Luggages	Luggage Set Givi E41N	2007-07-23
10	Luggages	Luggage Givi E52 Maxia 52Lt	2007-07-05
11	Luggages	Zarges Aluminum Side Case	2007-07-09
12	Locks	Abus Granit Power 58HB140/310	2007-07-23
13	Locks	Abus Mini Black 39	2007-07-06
14	Locks	Abus Steel-O-Flex 900/170	2007-07-16
15	Locks	Disclock Kryptonite K-Disc	2007-07-22
16	Locks	Disclock Kryptonite Cryptodisco	2007-07-01

Figure 2.1: Example of database

2.1.1 Types of Databases

There are two (2) main types of databases which are the Flat file database and Relational database [2].

A Flat file database contains a record per line which consists of plain text or mixed text and binary file. The fields from the record will be separated by delimiters such as commas or a fixed length of space. There is no structural relationship between two different records in the database. An example will be a sheet of paper which contains a list of names, the owners' addresses and the owners' salary. This is a flat file database which usually can be done by a typewriter or word processor.

There is a problem using a flat file database because it is very prone to corruption if it is applied on a semi-active database. This type of database base has no inherent locking mechanism which will detect a file is being used or modified.

An example of flat file database:

```
Lname, FName, Age, Salary|Smith, John, 35, $280|Doe, Jane, 28, $325|Brown, Scott, 41, $265|Howard, Shemp, 48, $359|Taylor, Tom, 22, $250
```

Figure 2.2: Example of Flat File database

A relational database has its collection of data item organized as a set of table which has been formally-described. Its data can be accessed from users and reassembled back in different ways without having the database tables to be reorganized. The Structured Query Language (SQL) in a relational database is a standard user and application program interface which is used for interactive queries for information from a relational database and for gathering data for reports

A relational database is a set of tables which consists of predefined categories known as column. Each row contains a unique instance/value of data corresponding to the categories defined by the column. Relational database allows connection between two different tables with the condition that both of the tables has at least one common field which it able to relate any two files.

Relational database management system (DBMS) is a management system for database with the ability to access the data organized in the tables of the database and relate common field from one record to another. It also has the capability to recombine the record from different tables which makes it a powerful tool for data usage.

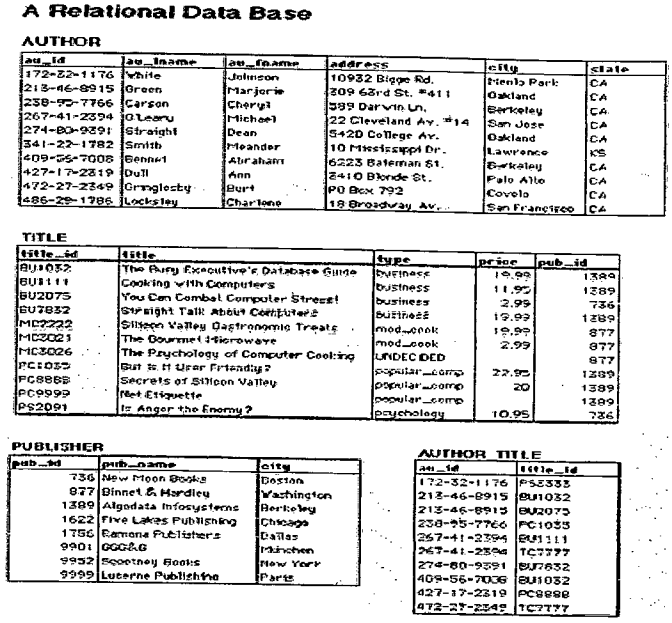

Figure 2.3: Example of relational database

2.1.2 Overview of Existing Database Security Technology

There are several relational databases technologies nowadays to support different types of usage and demand in the market. They are built with different architecture and tools. Here are the overviews of different databases.

Microsoft SQL Server 2005

MS SQL Server is a database management system (DBMS) developed and marketed by Microsoft. This system is the most important part of Microsoft.NET technology. MS SQL Server runs exclusively under Microsoft operating systems Windows 2000, Windows Server 2003, and Window XP [3].

SQL Server secures data with hierarchical encryption layers and a key management infrastructure. Each layer secures the layer beneath it, using a combination of certificates, asymmetric keys.

There are two (2) layers:

i. Service Master Key
ii. Database Master Key

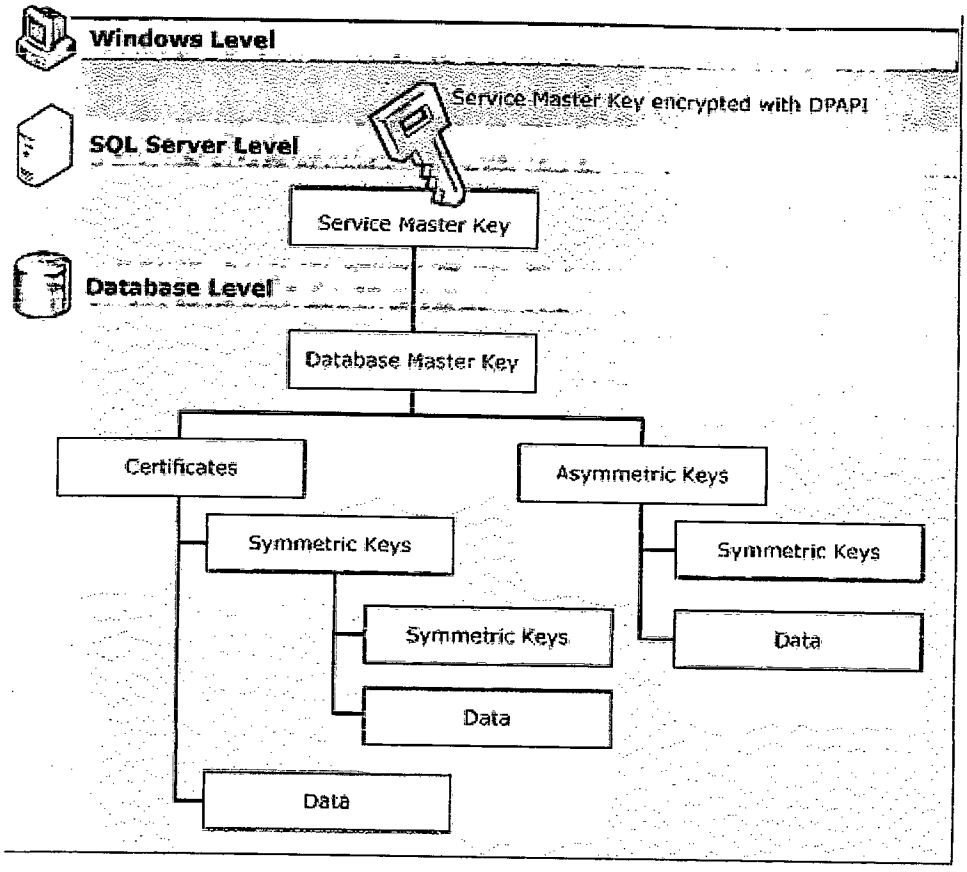

Figure 2.4: Architecture of Security in Microsoft SQL Server 2005

The top layer, Service Master Key, is encrypted using the Windows data protection API. Database Master Key depends on the encryption mechanism which provided by Microsoft.

PostgreSQL

PostgreSQL is an open source, Client/Server, relational database. PostgreSQL offers a unique mix of features that compare well to the major commercial databases such as Sybase, Oracle and DB2. The major advantage of PostgreSQL is that it is open source – the source code is readable by anyone. It is not owned by a single company because it is developed, maintained, broken and fixed by a group of volunteer developers around the world [4].

Important aspects in PostgreSQL:

i. Postgre SQL offers inheritance in the database
ii. Postgre SQL enables user to add new fundamental data types.
iii. Postgre SQL includes support for geometric data types such as point, line segment, box, polygon, and circle.

Three aspects in PostgreSQL security:

i. Securing the PostgreSQL data files
ii. Securing client access
iii. Granting and denying access to specific tables and specific users

Oracle Databases

The Oracle Database consists of a relational database management system (RDBMS) produced and marketed by Oracle Corporation. As of 2009, Oracle remains a major presence in database computing.

In order to meet the demands of encryption in database, Oracle8i introduced a PL/SQL package to encrypt and decrypt stored data.

The package, DBMS_OBFUSCATION_TOOLKIT, is provided in both standard edition and Enterprise Edition Oracle9i [5]. The package currently support bulk data encryption using Data Encryption Standard (DES) algorithm, and includes procedures to encrypt (DESEncrypt) and decrypt (DESDecrypt) using DES.

Besides that, Oracle has added support for the triple DES (3DES) encryption in Oracle8i. Furthermore, it also added support for cryptographic checksumming using the MD5 algorithm. Cryptography checksums can ensure data intergrity.

2.1.3 Database Security

There are different types of attack on the database which are classified as external and internal attack.

External attack as its name described, its source of attack is from unknown personnel outside the usage of database such as internet user or hackers from all around the world. The administrator takes several measures to curb this problem by setting up firewalls, intrusion detection systems and authorization before entering the database. This made it harder for the internet culprit to break in with all these network security tools [6].

The second attack will be the database internal attack, which has the advantage of being within the perimeter firewall and having access to information inside the territory. Many organizations have more worries about external attack while neglecting the risk of having attack internally. The internal threat is always underestimated as its manipulator often is the database administrator. This is because database administrator has access to everything in the database. They have the ability to read or edit the content in the database without leaving a trace behind [6].

Besides that, the basic of attack to the database can also be categorized into one-offs and persistent. One-off attack is an attack when the attacker achieves his goals inside the database and vanishes in the air without returning to the database again. However, persistent attack is an attack where the attacker will return to the compromised database frequently and launch additional attack to the database.