ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 4, Issue 2, February 2017

Security of Critical Data In Database – An Overview

[1] Naulesh Kumar
Principal, Vidya Memorial Institute of Technology, Tupudana, Ranchi

Abstract: -- **Today, it is imperative for enterprises to know which specific threats they are trying to protect against and take stringent measures to address those threats. We have developed a proposed Networks to bring complete data privacy to the enterprise. With this Network Data Secure Platforms, an organizations can protect critical data from both internal and external threats, and ensure compliance with legislative and policy mandates for security. Data Secure features a dedicated security appliance and specialized software that enables organizations to encrypt data in applications and databases.**

File level encryption within databases provides protection largely against media theft and loss. Column level encryption provides protection against media theft along with a broad range of logical threats. While protecting against identified threats, enterprises must consider the type of encryption to deploy to protect sensitive data in databases. Effectively addressing all these factors will ensure enterprises can protect sensitive data, while minimizing the overall cost to the business.

The process to migrate plaintext data to encrypted format is quite simple when we use this Network Connector; furthermore the process can be completely automated. The most important is that the integration is completely transparent to applications that interface with our sensitive data. Before deploying Data Secure, our sensitive data is encrypted, and applications can continue interacting with sensitive data using the same SQL statements; however, instead of interacting directly with that sensitive data, the application servers are actually interacting with a view of the data.

In this thesis, I have an algorithm and try to implement in Java. The result can be shown in the encrypted form on command Prompt Scree. With this any data, either Numeric, Alphanumeric or Strings can be encrypted. .

The percentage by which encrypted data is larger than plaintext data varies depending on the encryption algorithm that is used to perform the encrypt operation. In this thesis I used DESede cipher in CBC mode with PKCS5 padding, as such, a nine digit number Expands to 16 bytes of binary data and the results is shown in byte code. Its advantage is much faster than public key system and easy to implement in both hardware and software.

Using this technique an organizations can protect critical data from both internal and external threats, and ensure compliance with legislative and policy mandates for security. Secure key management, backup, and administration are a few of the core elements for achieving true data privacy. I have built from the ground up with security in mind, implementing these policies and procedures is fast and easy, utilizing our unique management interface. We encourage administrators to focus on developing key management and administrative policies for their organizations that will provide maximum security and hope that this thesis will serve as a guide in this effort.

I. INTRODUCTION

Today, it is most important for enterprises to know which specific threats they are trying to protect against and take stringent measures to address those threats. KUMARNET Networks brings complete data privacy to the enterprise. With this Data Secure Platforms, organizations can protect critical data from both internal and external threats, and ensure compliance with legislative and policy mandates for security. Data Secure features a dedicated security appliance and specialized software that enables organizations to encrypt critical data in applications and databases.

File level encryption within databases provides protection largely against media theft and loss. Column level encryption provides protection against media theft along with a broad range of logical threats. While protecting against identified threats, enterprises must consider the type of encryption to deploy to protect sensitive data in databases, while minimizing performance impact, ensuring strong and key management, and minimizing the impact of batch type processes. Effectively addressing all these factors will ensure enterprises can protect sensitive data, while Sminimizing the overall cost to the business.

Storage level encryption within databases provides protection largely against media theft and loss. Column level encryption provides protection against media theft along with a broad range of logical threats. While protecting against identified threats enterprises must consider the type of encryption to deploy to protect databases, performance impact, key management, and impact to batch type processes. These considerations

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 4, Issue 2, February 2017

along with execution on best practices for key management and security will ensure enterprises that their sensitive data is well protected and controlled within their environment.

II. INTRODUCTION OF CRYPTOGRAPHY

2.1

In current scenario number of problem arises with the security of documents, files and important data. So, there is a need to have a technique to protect the documents and which avoids the unauthorized access of data in an unsecure communication environment. There are various techniques which are used to keep the data confidential from hackers. Some of these are passwords, cryptography and biometrics. Passwords are not so good for this task due to their low entropy. Biometrics technique produces harmful effects on the human beings and it is too costly. For these above problems cryptography is the best solution for security. A plaintext is a message to be communicated in a secret way. Encryption is the process of creating a ciphertext (hidden data) from a plaintext and decryption is the reverse process of encryption, where cipher text is converted into plaintext.

The study of encryption and decryption is called cryptology and cryptography is the application of them. For encode a plaintext changes the plaintext into a series of bits or numbers alpha-numeric may be used which include A to Z and 0 to 9 values . The most common method of encoding a message these days is to replace it with its ASCII value, which is an 8 bit representation for each symbol. The process of decoding turns bits or numbers back into plaintext is called decryption. Stream cipher operates on a message symbol by symbol or bit by bit. A block cipher operates on blocks of symbols. A transposition cipher is used to rearranges the letters symbols or bits in a plaintext. A substitution cipher is used to replaces the letters symbols or bits into plaintext with others or without changing the order. A product cipher alternates the transposition and substitution. The concept of stream cipher versus block cipher only applies to substitution and product ciphers not transposition ciphers.

2.2 Cryptography

Cryptography is best method to protect data and Important files from unauthorized parties. It is the science of writing the data in secret code and about the design and analysis of mathematical techniques that is enables secure communication in the presence of millions adversaries. Cryptography is the art of secret writing. The basic service provided by cryptography is the ability to send information between participants in a way that prevents others from reading it [4]. Cryptographic systems involve both an algorithm and a secret value. The secret value is known as the key. The reason for having a key in addition to an algorithm is that it is difficult to keep devising new algorithms and it will allow reversible scrambling of information's.

2.3 Encryption

For achieving the data security encryption is the most effective way everywhere and everyplace where security is need. The process of hiding the contents of a message in such a way that the original information is recovered only through a decryption process is called encryption. The purpose of Encryption is to prevent unauthorized parties from viewing crucial information. An encryption occurs when the data is passed through some substitute technique like shifting technique, table references or mathematical operations. A different form of data is generated through these processes. The unencrypted data is called plaintext and the encrypted data is called cipher text. This represents the original data in a difference form. Encryption key is use to encrypt the original message which depend upon key based algorithms.

There are two general categories for key based Encryption algorithm first one is called Symmetric Encryption (SE) which uses a single key to encrypt the message and decrypt the message. Second is Asymmetric Encryption (AE) which uses two different keys a public key to encrypt the message, and a private key to decrypt the message . There are several different types of key based Encryption algorithms such as DES, RSA, PGP, Elliptic curve but all of these algorithms depend on high mathematical manipulations.

1.4 Keys

A key is a value that works with a cryptographic algorithm to produce a specific ciphertext.

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 4, Issue 2, February 2017

For encryption, key is used called encryption key and for decryption, it is called decryption key. Key size is measured in bits, larger the key size more secure communication. Key can be used by two ways, private or publically. In Symmetric Encryption, key is used as private key or it is also called single key for encryption and decryption. In Asymmetric Encryption, key is used as private or public key one for encryption and second for decryption

III. DATABASE INTEGRATION PROCESS

As the incidence and severity of security breaches continues to grow, it is increasingly incumbent upon organizations to begin encrypting data inside the enterprise. KUMARNET offers breakthrough solutions that make it practical to encrypt critical data, and ensure it's secured throughout an organization. With this DataSecure Platforms, organizations can better ensure that they are compliant with legislative and policy mandates for security, and eliminate the risks of a breach. KUMARNET DataSecure Platforms deliver comprehensive security capabilities:

- Encrypt critical data in Web servers, application servers, and databases.
- .Ensure all access to critical data is carefully managed, logged, and controlled.
- Administer keys and policies in a secure, centralized fashion.
- Ensure security processing is highly scalable and reliable.

DataSecure works seamlessly with leading databases—including IBM DB2, Microsoft SQL Server, and Oracle—delivering capabilities for securely and efficiently managing encryption. DataSecure encrypts data in the database at the column level, and can be used to secure such information as credit card numbers, social security numbers, passwords, account balances, and email addresses. DataSecure significantly streamlines the administrative tasks involved in database encryption—it automates much of the configuration and implementation process and it can be deployed without any disruption to the applications tied to the database. The KUMARNET DataSecure Platform is comprised of three components:

- The Data Secure appliance, a dedicated hardware system,
- The Network-Attached Encryption™ (NAE) Server, which runs on the DataSecure appliance, and
- The KUMARNET NAE Connector, software that is installed on the Web, application, or database server.

The NAE Connector features standards-based cryptographic interfaces that allow the protection of user-defined data through integration of security functions at the business logic layer. These small software components are installed on each database that has a need to interface with the Ingrian appliance, and they initiate encrypt and decrypt operations..

3.1 How it Works

Integrating the DataSecure platform in your existing database infrastructure is a straightforward, automated process. The NAE Connector component outlined above consists of complete code to manage a seamless interaction between the database and the DataSecure platform. The DataSecure appliance features a secure, Web-based user interface that walks administrators through a step-by-step configuration process and, once parameters have been set, even automates the installation of the required NAE Connector software on the database. Once installed and configured, the NAE Connector dynamically generates all the necessary stored procedures and functions to:

- Encrypt and decrypt data on demand from inside the database.
- Migrate data from plaintext to ciphertext and change the database schema to accommodate encrypted columns.
- Rotate cryptographic keys.
- Automate subsequent encrypt and decrypt operations.
- Authenticate users so that only authorized users are able to access sensitive data.

This transparent integration means that you can continue using your existing SQL statements without having to modify them. And, more importantly, you do not have to write any of the logic to perform encrypt or decrypt operations from your database. Following is a high-level diagram outlining how the solution is deployed.

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 4, Issue 2, February 2017

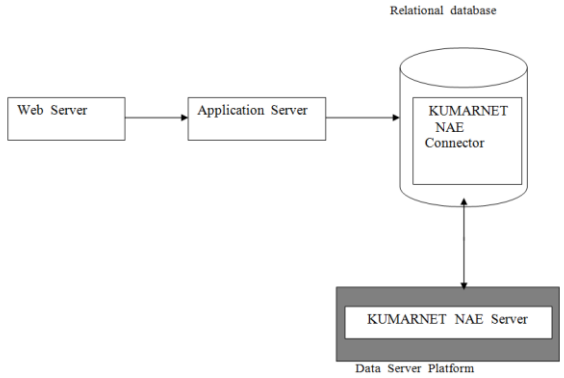


Figure 2 : High Level View of Implementation with KUMARNET Data Secure Platforms

The diagram above shows a Web server accessing an application server, which is making a call to a database to access sensitive data. The NAE Connector is installed in the database and the sensitive data is stored in encrypted format within the database. The user who requests the sensitive data must have permission within the database to make a request to the NAE Connector. That user must also have access to the requested key on the NAE Server. If either of the conditions above is not met, then the user is not given access to the sensitive data. If the user is authorized to use the requested key for decryption, then the NAE Server performs the decrypt operation on the sensitive data. The decrypted data is then passed back to the database. The DataSecure appliance is the physical device where cryptographic operations and key management operations are performed. Different hardware platforms provide varying capabilities for performance and FIPS compliance. The NAE Server and all cryptographic keys reside on this hardened security system.

3.2. How Does Data Get Encrypted?

Before implementing DataSecure in your enterprise, your sensitive data is most likely being stored as plaintext, so the logical question is: how do you migrate plaintext data into encrypted format? The process is straightforward, and, as mentioned above, KUMARNET automates this process through the DataSecure appliance's GUI. To illustrate the simplicity of the process, take social security numbers as an example. If you have a table called CUSTOMER that stores sensitive customer data like names, addresses, and social security numbers, you might want to encrypt the social security numbers. Your original CUSTOMER table might look like this:

SQL> select * from CUSTOMER;

NAME	SSN	ADDRESS	CITY	STATE	ZIP
Ranjan	123456789	Ranchi	Ranchi	Jharkhand	835220
Nitin Mohan	ABC246809	Delhi	Delhi	Uttar Pradesh	110030
N.Kumar	BIT1018-04	Belly Rd.	Patna	Bihar	800001
Raj kiran	HCL04CS16	Kakinara	Andhra	Andhra	630032

Table 2.1: Sensitive Data Is Stored In the Column SSN.

The first step in the process of securing your sensitive data is to identify what data you want to secure and where that data resides. In this example, social security numbers are stored in a column called SSN. Once you have identified the sensitive data, you can configure KUMARNET to automate this data protection process. During the first step, KUMARNET renames the table in which the sensitive data resides; the table must be renamed so that a view can be created later with the same name as the original table. Notice in Figure 3 that the table is renamed to CUSTOMER_ENC, but the SSN column is not yet changed.

SQL> select * from CUSTOMER;

NAME	SSN	ADDRESS	CITY	STATE	ZIP
Ranjan	123456789	Ranchi	Ranchi	Jharkhand	835220
Nitin Mohan	ABC246809	Delhi	Delhi	Uttar Pradesh	110030
N. Kumar	BIT1018-04	Belly Rd.	Patna	Bihar	800001
Rajkiran	HCL04CS16	Kakinara	Andhra	Andhra	630032

SQL> select *from CUSTOMER_ENC;

NAME	SSN	ADDRESS	CITY	STATE	ZIP
Ranjan	123456789	Ranchi	Ranchi	Jharkhand	835220
Nitin Mohan	ABC246809	Delhi	Delhi	Uttar Pradesh	110030
N.Kumar	BIT1018-04	Belly Rd.	Patna	Bihar	800001
Rajkiran	HCL04CS16	kakinara	Andhra	Andhra	630032

Table 2.2 : Rename Customer Table

In the next step, kumarnet creates a temporary table and exports the sensitive data to that temporary table. Notice in figure 4 below that ssn is the only column exported to the temporary table from the original table. The row_id

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 4, Issue 2, February 2017

column is added automatically and used later when returning the encrypted data back to the original table. The values in the column that held the sensitive data in the customer_enc table (remember—the customer table was renamed) are set to null to avoid any data conversion issues that might arise when changing the data type in a later step.

SQL> select *from CUSTOMER_ENC;

NAME	SSN	ADDRESS	CITY	STATE	ZIP
Ranjan	NULL	Ranchi	Ranchi	Jharkhand	835220
Nitin Mohan	NULL	Delhi	Delhi	Uttar Pradesh	110030
N.Kumar	NULL	Belly Rd.	Patna	Bihar	800001
Rajkiran	NULL	Kakinara	Andhra	Andhra	630032

SQL> select *from CUSTOMER_TEMP;

SSN	ROW_ID
123456789	1
ABC246809	2
BIT1018-04	3
HCL04CS16	4

Table 2..3: Export Sensitive Data to Temporary Table

Before Ingrian can populate the original column with encrypted data, it must modify the column size and data type because encrypted data is predictably larger than plaintext data, and most likely, your social security numbers are stored as some sort of integer or character data type. Although KUMARNET gives you the option to store your encrypted data in Base64 encoded format, it is recommended that you store your encrypted data in binary (the default choice during configuration) because binary data is smaller than Base64 encoded data and there is less overhead with binary because the system does not have to encode and decode data with every encrypt or decrypt operation.

SQL> desc CUSTOMER;

Name	Null?	Type
NAME		VARCHAR2(5)
SSN		CHAR(9)
ADDRESS		VARCHAR2(10)
CITY		CHAR(6)
STATE		CHAR(5)
ZIP		NUMBER(6)

SQL> desc CUSTOMER_ENC;

Name	Null?	Type
NAME		VARCHAR2(5)
SSN		VARCHAR2(16)
ADDRESS		VARCHAR2(10)
CITY		CHAR(6)
STATE		CHAR(5)
ZIP		NUMBER(6)

Table 2. 4: Modify Data Type and Column Size of the Encrypted Table

Once the column is modified, KUMARNET can migrate the sensitive data back into the CUSTOMER_ENC table. Before the data is imported back into the CUSTOMER_ENC table, however, the NAE Connector sends the data to the NAE Server, where the data is encrypted. The NAE Server returns the encrypted data to the NAE Connector, which then inserts the encrypted data into the CUSTOMER_ENC table.

SQL> select *from CUSTOMER_TEMP;

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 4, Issue 2, February 2017

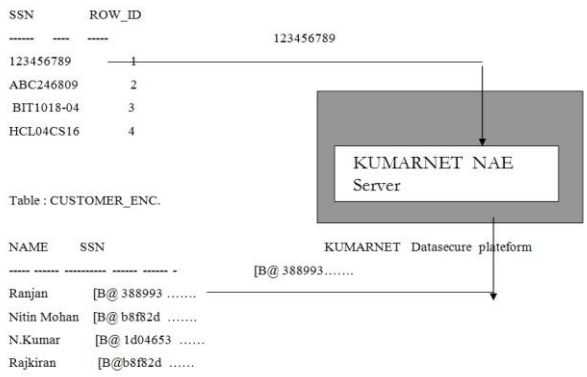


Table 1.5 : Encrypt Data and Insert into CUSTOMER_ENC Table

After encryption, the CUSTOMER_TEMP table is dropped and the CUSTOMER_ENC table might look something like this:

SQL> select *from CUSTOMER_ENC;

NAME	SSN	ADDRESS	CITY	STATE	ZIP
Ranjan	[B@ 388993	Ranchi	Ranchi	Jharkhand	835220
NitinMohan	[B@ b8f82d	Delhi	Delhi	Uttar Pradesh	110030
N.Kumar	[B@ 1d04653	Belly Rd.	Patna	Bihar	800001
Rajkiran	[B@b8f82d	kakinara	Andhra	Andhra	630032

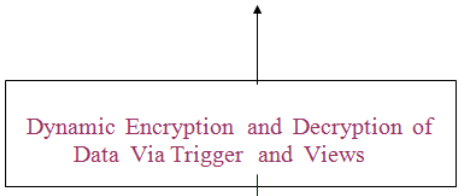
Table 2.6 : Sensitive Data Is Stored in Encrypted Format

1.3 How Do You Automate Subsequent Updates and Inserts?

Once your table and column are able to accommodate encrypted data, KUMARNET automates encryption and decryption of data by creating a view, triggers, and stored procedures that are generated during configuration to work with the NAE Connector. In this way, properly authenticated applications outside the database can continue to query and update the same database tables as before. The NAE Connector remains transparent to outside applications, and, more importantly, the amount of code changes necessary to integrate the NAE Connector is minimal.

CUSTOMER (View)

NAME	SSN	ADDRESS	CITY	STATE	ZIP
Ranjan	123456789	Ranchi	Ranchi	Jharkhand	835220
Nitin Mohan	ABC246809	Delhi	Delhi	Uttar Pradesh	110030
N.Kumar	BIT1018-04	Belly Rd.	Patna	Bihar	800001
Rajkiran	HCL04CS16	Kakinara	Andhra	Andhra	630032



CUSTOMER_ENC (Table)

NAME	SSN	ADDRESS	CITY	STATE	ZIP
Ranjan	[B@ 388993	Ranchi	Ranchi	Jharkhand	835220
Nitinmohan	[B@ b8f82d	Delhi	Delhi	Uttar Pradesh	110030
N.Kumar	[B@ 1d04653	Belly Rd.	Patna	Bihar	800001
Rajkiran	[B@b8f82d	kakinara	Andhra	Andhra	630032

Table 2.7 : View is Automatically Instantiated.

As you can see from Figure 8 above, when sensitive data is accessed, the view is instantiated by the database and populated with decrypted data from the CUSTOMER_ENC table. Because the view has the same name as the original table, all SQL statements that reference the encrypted data can function regularly without modification. Likewise, triggers trap all the inserts and updates executed on the view. If an insert statement is detected, a new insert statement is generated based on the original insert values. The social security number in this case is encrypted before insertion into the base table. Similarly, when an update statement is executed, a new update statement is generated to update the base table (CUSTOMER_ENC) as opposed to the view referenced in the call from the outside application (CUSTOMER).

As was mentioned above, the process to migrate plaintext data to encrypted format is quite simple when using the NAE Connector; furthermore the process can be completely automated through the use of triggers,

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 4, Issue 2, February 2017

views, and stored procedures, all of which are created and
Installed by the DataSecure appliance during configuration. What's most important is that the integration is completely transparent to applications that interface with your sensitive data. Before deploying DataSecure, your sensitive data sits in the clear in your databases.

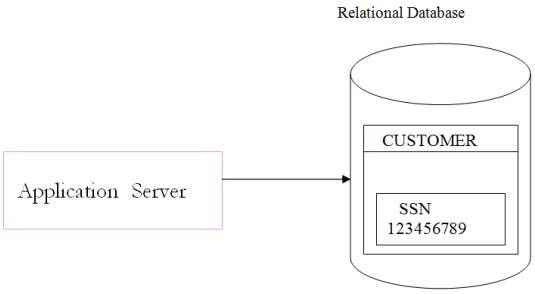


Figure2.2: App Server Interacts with Plaintext Data Before Deploying DataSecure

After deploying DataSecure, your sensitive data is encrypted, and applications can continue interacting with sensitive data using the same SQL statements; however, instead of interacting directly with that sensitive data, the application servers are actually interacting with a view of the data.

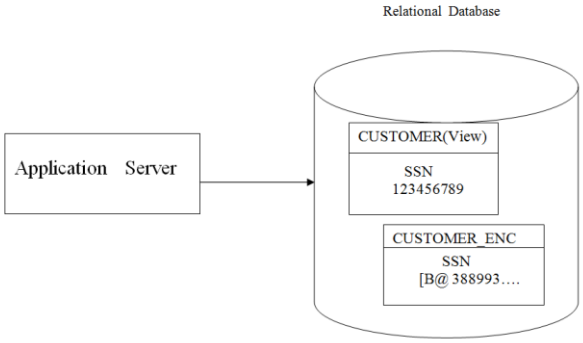


Figure 2..3 : App Server Interacts with Plaintext View of Encrypted Data After Deploying Data Secure.

1.4 Output of Java Programm.

Output of the java program is shown bellow in Command Prompt which is written in Appendix B and Appendix C for Encryption and Decryption of any data.

Output of the Column SSN , Encrypted by DESede Encryptin Algorithm in CBC mode in shown on the Command Prompt screen .

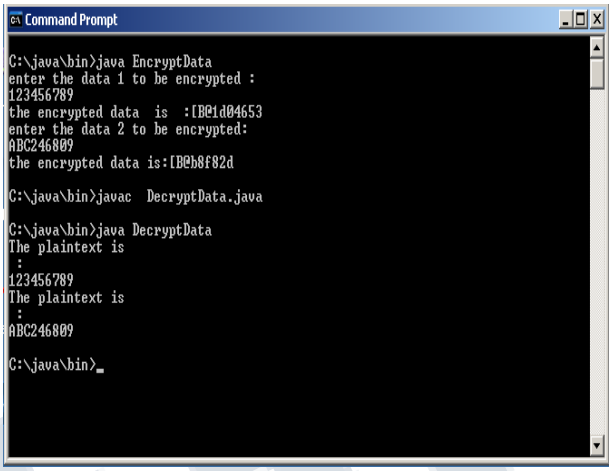


Figure 1.4 a : Output of the data (SSN) in Encrypted and Decrypted form.

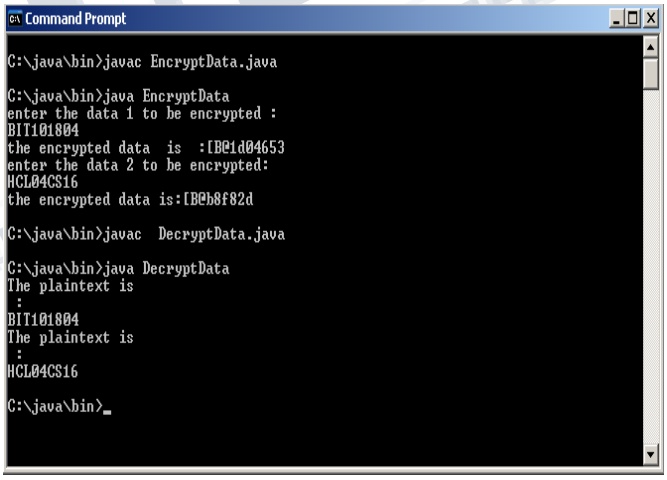


Figure 1.4 b : Output of the data (SSN) in Encrypted and Decrypted form.

Applications and future Work

Cryptography is utilized in various applications and environments. The specific utilization of encryption and the implementation of TDEAwill be based on many factors particular to the computer system and its associated components. In general, cryptography is used to protect data while it is being communicated between

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 4, Issue 2, February 2017

two points or while it is stored in a medium vulnerable to physical theft or technical intrusion (e.g., hacker attacks). In the first case, the key must be available at the transmitter and receiver simultaneously during communication. In the second case, the key must be maintained and accessible for the duration of the storage period. This I Networks brings complete data privacy to the enterprise. DataSecure features a dedicated security appliance and specialized software that enables organizations to encrypt critical data in applications and databases.

The Encryption technique (TDEA) that we used in this thesis can also be implemented in other modes like OFB and CFB mode using PKCS5 padding or No padding in Java or Some other language like C and C++. With This DataSecure Platforms, organizations can protect critical data from both internal and external threats, and ensure compliance with legislative and policy mandates for security. Secure key management, backup, and administration are a few of the core elements for achieving true data privacy. Because this DataSecure Platforms have been built from the ground up with security in mind, implementing these policies and procedures is fast and easy, utilizing our unique management interface. We encourage administrators to focus on developing key management and administrative policies for their organizations that will provide maximum security and hope that this thesis will serve as a guide in this effort.

REFERENCES:

Books and Publications.

1."Cryptography Decrypted " By H.X Mel & Doris Baker, Publication Addison – Wesley.

2. "Fundamentals of Network Seccurity."By Eric Maiwald, Dreamtech publication Edition –2004.

3. "Crytography and Network Security Principle and Practice" By Stalling , William, 2ed edition , Pretice Hall ,1999 ,ISBN)-13-869017-0. Practical Discussion of Cryptographic principle.

4. "Computer Networks" By Tanenbaum , Andrews, Parentice Hall of India, New Delhi- 1987

5. "Data Communications and Networking". By Behrouz A. Forouzans

Tata McGraw-Hill Publishing Company Limited, Edition- 2000.

6 "Cryptography in C & C++" By Michael WelschenbBach, translated by David Kramer.

7. "SQL,PL/SQL Progamming Language of oracle." By – Ivan Bayross.

8. " Starting Out with Oracle covering Database. " By John Day craig Van Slyke

9. "Beginning Oracle Programming" By Sean Dillon , Christopher Beck, Thomas Kyte,

& Joel Kallman , Shroff Publishers & Distributors Pvt. Ltd.

10." Internet & JAVA Programming" By R.Krishnamoorthy, S.Prabhu, New Age International(p)Limited, Publishers, New Delhi – 2003.

11. " Starting Out With JAVA" , By Tony Gaddis. , Published by Dreamtech , New Delhi-2004

12. "ICSA Guide to Crytography",By Nicholos, Randy , McGraw – Hill, !999. Comprehensive Discussion of Historical to Modern –day Cryptograpgy.

Internet Resources.

http://ieee.org
http://csrc.nist.gov/encryption/aes/
http://www.ingrian.com
http://www.cs.org
http://www.airdefense.net.
http://www.google.com,
http//www.bitpipe.com/rlist/datasecurity.html.
http://www.esat.kuleuven.ac.be/~rijmen/rijndael/

http://www.cis.comell.edu,
http://isaac.cs.berkeley.edu/mobicom.pdf.
http://www.computeruser.com/resources/dictionary/dictionary.html (reference for technical terms) .
http://www.sciencedirect.com
http://www.scirus.com
http://wwwspringerlink.com
http://www.computerworld.com (provides white papers, surveys, and reports) .
http://www.informationweek.com (provides information on wireless networks, wireless communications, and security solutions in the form of articles and other documents).
http://www.infosecuritymagazine.com (provides white papers, surveys, and reports).

http://www.isaac.cs.berkeley.edu/isaac/wep-faq.html(Universityof California at Berkeley.
RSA Laboratories. PKCS#11 v2.11: Cryptographic Token Interface Standard, Revision 1, November 2001. ftp://ftp.rsasecurity.com/pub/pkcs/ pkcs-11/v211/pkcs-11v2-11r1.pdf Mark Russinovich. Inside Encrypting File System. Windows & .NET Magazine, June -July 1999.
Sun Corporation. The Java™ Cryptography Architecture API Speciation & Reference. http://java.sun.com/products/jdk/1.2/docs/guide/ security/CryptoSpec.html (valid as of September 2002), December 1999.

Articles and White paper Material.
http://www.ieeexplre.ieee.org,
http://www.ingrian.com
http://www.airdedense.com,
http:// www.itsecurity.com.
http://www.infosecuritymagazine.com