IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

PREVENTION OF ATTACKS ON DATABASES USING AES

[1]Shaik Arshia Zainab, [2]Shaik Abdul Malik Khudus
[1]Undergraduate Student, [2]Undergraduate Student
[1]Department of Information Technology
[1]Gokaraju Rangaraju Institute of Technology and Engineering, Hyderabad, India

Abstract: The Internet is a source for thousands of web-based applications. For each web application, there is a database associated with storing information related to users. These databases often store confidential data like login credentials, user transaction details, addresses, so on and so forth. Intruders are in a constant race to be able to hack and exploit the sensitive data stored in such databases. SQL (structured query language) injection is a threat that attacks the databases used for storing data collected from various web-based applications. The intention behind such attacks is to mainly steal or exploit the data stored in a database, by illegally accessing and gaining ownership. Weakly protected databases are the ones that are vulnerable to such attacks. SQL injection is mainly about entering malicious queries in the input fields. In this paper, we propose a model wherein we focus on encrypting the data stored.

Index Terms: **AES, Database, Encryption, Intruder, SQL, Web Applications**

I. INTRODUCTION

SQL Injection is one of the main concerns that intervenes with database security. It attacks the database without the consciousness of the database administrator. It may tamper the data stored in various tables and databases, by deleting the full database or a few records or tables without the knowledge of the respective user or administrator. It is a technique used to exploit the database system through vulnerable web applications, mainly databases that are weakly protected. These attacks not only intend to breach the security and steal the entire content of the database but, at the same time, they try to make irrational changes to both the database schema and the information stored in the database. Detection of such attacks is a problem, as it cannot be detected early. The attack is identified, at a point where the information is already compromised. Often, in scenarios wherein the attacker aims to collect confidential data, the user is unaware that his confidentiality is compromised. Intruders often need a simple web browser to attack the database. In understanding how and where a possible attack occurs, the logical view of web applications consisting of three tiers needs to be understood.

1) User Interface Tier: This is a layer wherein the user directly interacts with the website. The user inputs decide what kind of interactions should take place between other layers. It forms the front end of the web application.

2) Business logic tier: This layer deals with server-side programming. Here, the requests from the users are processed. This layer acts as an intermediary layer between the user interface tier and the database tier.

3) Database tier: This layer involves the retrieval and storage of information. Hence, it deals with the database server.

II. PROPOSED SYSTEM

The system that we propose deals with encrypting the data present in the database using a very efficient encryption algorithm known as Advanced Encryption Standard (AES). This procedure ensures that even if the intruder breaches the security and gains access to the database, all he can access is encrypted data. Hence, there is no chance of misusing data. Also, we have used stored procedures to ensure that the intruder or attacker is not able to alter the table contents. AES when abbreviated stands for Advanced Standard Encryption. Encryption of electronic data is possible with the use of the AES algorithm. This algorithm is based on the structure of the substitution-permutation network (SPN). SPN is a network wherein the inputs are the plaintext or unencrypted text and a key. Then to the inputs, several rounds of substitution-boxes and permutation-boxes are applied to get a block of ciphered text. This whole process is iterative, and the key is produced at the end of each round. By reversing this entire process, we can decrypt the cipher block.

A. AES Algorithm:

In this algorithm, there are related operations that involve both substitution and permutation. Substitution deals with replacing bits, whereas, the permutation is shuffling one or more bits. AES algorithm uses a matrix arrangement for encryption and decryption. All the iterations of the algorithm use bytes instead of bits. For instance, the plain text of 128 bits is 16 bytes. Further, in this algorithm, the bytes are arranged in a 4x4 matrix, and it has a fixed block size of 128 bits with a variable key size of 128, 192, or 256 bits. Based on the key size, the number of iterative rounds to be performed are determined. 10, 12, and 14 rounds for 128, 192, and 256 bits respectively.

Process of Encryption:

For each round, the below steps repeat in the same sequence.

Step1: Byte Substitution - In this step, a sub byte replaces each of the 16 bytes using a substitution box. Step2:

Shifting Rows - All the rows of the matrix shifted to left. Any left out placed on the right side.

Step3: Mix Columns - Except for the last round, this step repeats for all iterations. In this step, a mathematical function is used to take four bits from each column and gives four new bits. This way, at the end of this step, 16 new bytes are produced.

Step4: Add round key - If this is the last round, then the output of this round is the ciphered block of text. In this step, the bytes (16 bytes) are to consider as 128 bits of the key for that particular round.

Decryption Process:

In this process, all we have to do is reverse the steps involved in encryption. Both the encryption and decryption algorithm implementations are separate.

III. RESULTS AND DISCUSSION

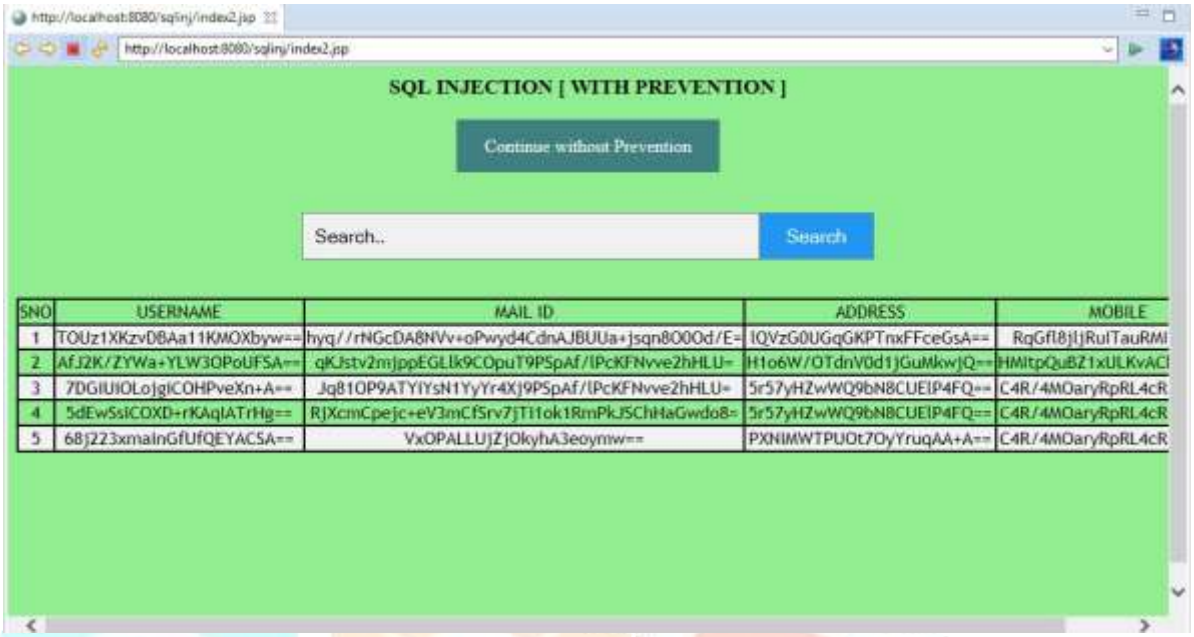
1. Weakly protected database with no encryption or protection:

In weakly protected databases, on entering malicious queries in the web text fields leads to data hacking. Attacker is able to access all the fields in the database.

Fig 1. Without AES

2. *Encryption with AES:*

On encrypting with the AES algorithm, even if the attacker succeeds in gaining access to the database, all he is able to retrieve is encrypted data.



SNO	USERNAME	MAIL ID	ADDRESS	MOBILE
1	TOUz1XKzvDBAa11KMOXbyw==	hyq//rNGcDA8NVv+oPwyd4CdnAJBUUa+jsqn8O0Od/E=	IQVzG0UGqGKPTnxFFceGsA==	RqGfl8jIjRulTauRMI
2	AfJ2K/ZYWa+YLW3OPoUFSA==	qKJstv2mjppEGLlk9COpuT9PSpAf/lPcKFNvve2hHLU=	H1o6W/OTdnV0d1jGuMkwJQ==	HMltpQuBZ1xULKvACl
3	7DGIUIOLojgICOHPveXn+A==	Jq81OP9ATYIYsN1YyYr4XJ9PSpAf/lPcKFNvve2hHLU=	5r57yHZwWQ9bN8CUEIP4FQ==	C4R/4MOaryRpRL4cR
4	5dEwSsICOXD+rKAqIATrHg==	RJXcmCpejc+eV3mCfSrv7jTi1ok1RmPkJ5ChHaGwdo8=	5r57yHZwWQ9bN8CUEIP4FQ==	C4R/4MOaryRpRL4cR
5	68j223xmaInGfUfQEYACSA==	VxOPALLUjZjOkyhA3eoymw==	PXNIMWTPUOt7OyYruqAA+A==	C4R/4MOaryRpRL4cR

Fig 2. With AES encryption.

IV. CONCLUSION

Almost all web applications require user input at some point, which attackers can use as an opportunity to steal confidential data. For this, our proposed system delivers an efficient technique to encrypt the data. Hence, even if a security breach goes unidentified, the attacker will not be able to collect confidential data.

V. FUTURE SCOPE

This method of encrypting the data can be introduced in mobile applications. The naive developers can utilize this to protect customer data, collected through mobile applications.

REFERENCES

[1] Zainab S. Alwan et al, "International Journal of Computer Science and Mobile Computing, Vol.6 Issue.8", August- 2017, pg. 5-17

[2] Mehar Sood, Smita Singh, "International Journal of Advanced Computational Engineering and Networking", ISSN: 2320-2106, Volume-5, Issue-7, Jul.-2017

[3] Alireza Hodjat, Ingrid Verbauwhede, “The Energy Cost of Secrets in Ad-hoc Networks (Short Paper)”.

[4] Ako Muhamad Abdullah, “Advanced Encryption Standard (AES) Algorithm to Encrypt and Decrypt Data”.