MAT JOURNALS
Journal of Web Development and Web Designing
Volume 3 Issue 2

Web Application Shielding

[1]R. Chithra Devi, [2]G. Divya, [2]M. Priyadharshini
[1]Assistant Professor, IT Department, Dr.Sivanthi Aditanar College of Engineering, Tiruchendur - 628215
[2]Department of Information Technology, Dr.Sivanthi Aditanar College of Engineering, Tiruchendur – 628215

Abstract
In the olden days computer networks were used for sending emails so there was no issue of security but now a days people who are using internet as sharing tool are hacking the financial products like credit cards, debit cards by hacking the pin numbers and passwords and are misusing the accounts. There are several threats for the online applications such as hacking, intrusion and so on. Nowadays, application security is rapidly being recognized as a top priority. The systems store and retrieve knowledge and it'll shield the information from unauthorized users, disclosure, modification or destruction. Systems can make sure that the users have the authority to access the information, load new knowledge, or update existing knowledge. It is a very huge and complex task to provide security for a web application. So to avoid such problems a Web Application Shielding with the help of encryption techniques can be developed. This prevents hackers from exploiting vulnerabilities. This provides a higher level of security.

Keywords: *Authentication, Encryption &Decryption*

INTRODUCTION

Web application security is that the data Security and it deals with the safety of internet sites, net applications and net services.

Application security includes the measures taken to enhance the safety of an application cherish finding, fixing and preventing security vulnerabilities. Totally different techniques are accustomed surface such security vulnerabilities at different stages of an applications lifecycle such style, development, deployment, upgrade, maintenance. A continuously evolving however mostly consistent set of common security flaws are seen across completely different applications.

Asset
A resource of value such as the data in a database, money in an account, file on the file system or any system resource.

Vulnerability
A weakness or gap in security program that can be exploited by threats to gain unauthorized access to an asset.

Attack (or exploit)
An action taken to harm an asset.

Threat
Anything that can exploit vulnerability and obtain, damage, or destroy an asset.

Common technologies used for identifying application vulnerabilities include

Static Application Security Testing (SAST) is used as a Source Code Analysis tool. The method analyses source code for security vulnerabilities before the launch of an application and is employed to strengthen code. This method produces fewer false positives but requires access to an application's source code.

Dynamic Application Security Testing (DAST) is a technology, that is in a position to search out visible vulnerabilities by feeding a url into an automatic scanner. This methodology is

extremely ascendable, simply integrated and fast. DAST's drawbacks consists the necessity for professional configuration and therefore the high chance of false positives and negatives.

Interactive Application Security Testing (IAST) is a resolution that assesses applications from among using code instrumentation this method permits IAST to mix the strengths of SAST and DAST strategies also as providing access to code, hypertext transfer protocol traffic, library data, backend connections and configuration data. Some IAST products require the application to be attacked, while others can be used during normal quality assurance testing.

OBJECTIVE

To create a secure web application

Web application is trending in this digital period of life and its security can be more optimized as well as easy solution for web based issues, thus we create a web application shielding to achieve this.

To implement encryption algorithm

The application will be more secure when we provide an option of adding public key cryptosystems such as RSA and Caesar cipher techniques.

To obtain secure information exchange

The information to be exchanged is send privately as cipher text using encryption techniques.

BACKGROUND

The terminologies and technologies used in this project are described as follows:

Access key using PHP

Access key is generated for logging in into the website. It is obtained from the user credentials coded using PHP.

HTML

The Hyper Text Mark-up Language is used to develop the web application for which security is required.

Encryption &Decryption techniques

The data to be transferred must be kept private. It can be done by changing the data into cipher text by using encryption technique and vice versa.

ANALYSIS AND DESIGN

Database Module

The details that are needed for the application are available which include user credentials. The information are fetched for Authentication and key generation.

Key generation Module

Access key and Random keys are generated with the information provided in the database. The access key is used for logging in and random keys are used by the RSA and Caesar cipher public-key cryptosystems.

Login Module

Using the Access key and date of birth one can login to the website. Users are restricted to use the website if the detail does not match with the details in database.

Authentication Module

For authentication purpose unique details of the individuals are used. The further process is carried out only if the verification is completed.

Encryption Module

After verification is done the entered data by user gets encrypted twice with the help of random keys which is stored in the database. The encryption algorithms used are RSA and Caesar cipher public key cryptosystems.

Decryption Module

The cipher text is decrypted into original data using the same algorithm.

ARCHITECTURE

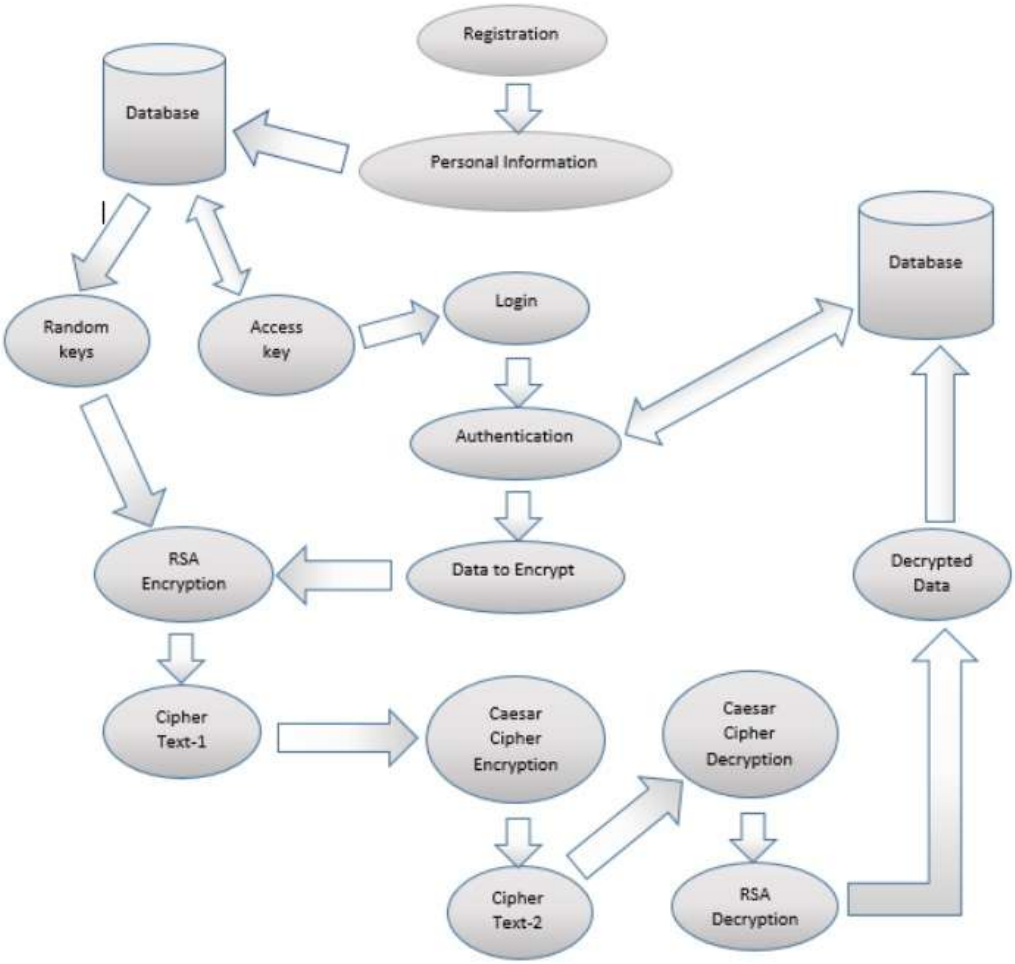
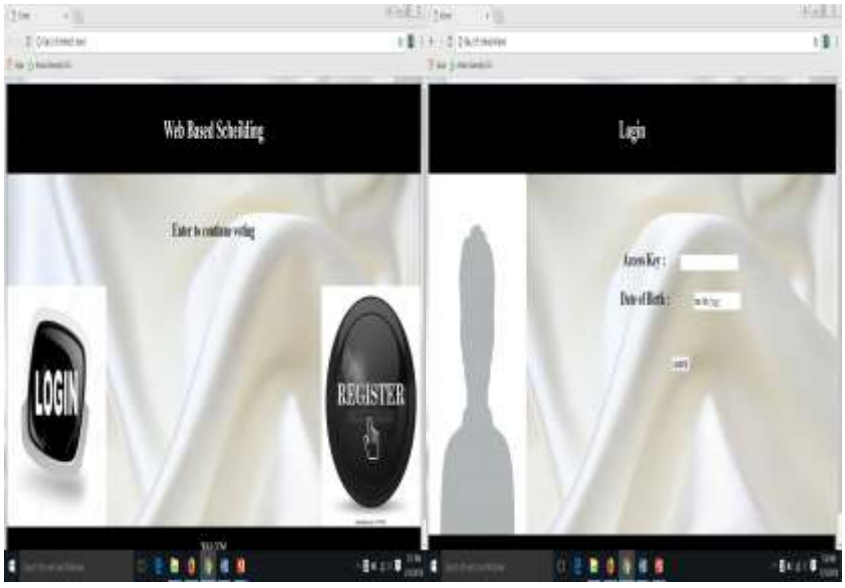


Fig. 1: *Architecture*

SCREENSHOTS



(a) **(b)**

(c) **(d)** **(e)**

Fig. 2: *(a-e): Screenshots*

IMPLEMENTATION

Module-1

The user credentials will be stored in the database with the help of registration. The entered details will be used for the key generation.

Module-2

The keys such as Access key and random keys are generated with the help of user credentials. The access key is used for the login purpose and the random keys are used for the encryption algorithms.

Module-3

After login, Verification is done with the help of unique details of the individuals. After the verification gets completed the vote has to be casted.

Module-4

The entered data is encrypted twice with the help of RSA and Caesar cipher public key cryptosystems. The votes are counted then the result is published to the users.

IMPLEMENTAION RESULTS

The user can create this application for the security purposes. This keeps the users confidential data in a secure way than the normal application

FUTURE WORK

This can be used for several security based application. It can be added with several unique details of the individuals.

CONCLUSION

Thus if the application is implemented it can provide a better way of implementing web security. It provides a secure way of using web applications. It gives confidentiality and protects against intrusions. Authorization can be achieved. We can achieve data Security.

ACKNOWLEDGEMENT

The authors thank the almighty lord, parents and the faculty members of Dr.Sivanthi Aditanar College of Engineering for recognition of paper in presentation and department achievements.

REFERENCES

1. Karthik, Chinnasamy, Deepalakshmi," Hybrid cryptographic technique using OTP: RSA" Intelligent Techniques in Control, Optimization and Signal Processing (INCOS), 2017 IEEE International Conference
2. [online].available:https://en.wikipedia.org/wiki/RSA_(cryptosystem)
3. [online].available: https://en.wikipedia.org/wiki/Public-key_cryptography

4. [online].available: https://www.w3schools.com/
5. Musa Bala Shuaibu, Ruqayyat Ahmad Ibrahim "Web application development model with security concern in the entire life-cycle" Engineering Technologies and Applied Sciences (ICETAS), 2017 4th IEEE International Conference.