ISSN XXXX XXXX © 2017 IJESC

Research Article Volume 7 Issue No.5

# Novel Countersign Vault

Adiba Firdous[1], Prof. Syeda Butool Fatima[2]
B.Tech Student[1], Assistant Professor[2]
Department of Computer Science Engineering[1], Department of Electronics and Communication[2]
JPNCE, Dharmapur, Mahboob Nagar, Telengana, India[1]
KBNCE, Gulbarga, karnataka, India[2]

**Abstract:**
Password managers are critical pieces of software relied upon by users to securely store valuable and sensitive information, from online banking passwords and login credentials to passport- and social security numbers. Surprisingly, there has been very little academic research on the security The purpose of this project ”COUNTERSIGN VAULT” is to deal with the security issues ,to make it more secure and reliable to save different lists of passwords along with their username for different accounts with a single master password. Here we use a reverse string technique to store the password. We provide a two level security, first at the time of login and next at the time of accessing the stored information so that no one can copy or steal or modify the passwords.

**Keywords:** Countersign vault, password manager, master password, user login, encryption

## 1. INTRODUCTION

The main aim of the proposed system is to provide a more secure and reliable way for handling of secret information like passwords for different accounts. Passwords are important aspect of computer security. All users with access to a single system , are responsible for taking appropriate steps, to select and their passwords. The scope of this is includes all personnel who have or are responsible for an account (or any form of access that supports or require password) on any system that resides at any facility, has access to or stores any non- public information. The objective of the project is to make password management efficient and effective. We can analyze the various lists of passwords along with their username for the specific website. Success of the project lies in aspects such as. How well to automate the system. How well the user feels ease with the system. How well the client-server interact with the system. Overall product after completion of this product should overcome the inabilities of the previous password managers and the final product should consist of The final product should display only the username and password for the required website. For each activity the user needs to login every time. More security should be present at the time of login. For every activity, user should provide master password once again. Exactly the same information as given by the user should not be stored. Implementation of the CONTERSUGN VAULT can help to provide more security in a reliable way to store and retrieve passwords along with their user names for different websites for different users. This system is a bit easy to implement as well as easily accessible with two level security. As the number of services offered on the Internet continues to increase, the number of passwords an average user is required to remember increases correspondingly, to the point where it is no longer feasible for most people to remember a new, strong password, for every account. Users typically solve this problem in one of two ways. A common solution is to reuse the same password on many different websites. This approach increases the potential damage if a password is stolen, cracked, or if a service that has access to it is compromised, since the attacker will be able to reuse it on all online services that share the password. Another approach is to use a password manager to store strong (random) passwords for each site.

## 2. LITERATURE REVIEW

The existing password manager uses normal web application, which will be suffered by numerous security attacks.
The main drawbacks in the existing system are

- It provides only one level authentication i. e, only at the time of login, only username and password will be matched with the present database.
- After logging in, at the time of accessing the information, no authentication will be done.
- If anyone will come to know the master password, they can easily access the whole information present inside the database.
- After logging in, whole list of password stored in the database will be displayed.
- It is less secure and prone to more security threats.

## 3. PROPOSE SYSTEM

The proposed system provides two level authentication and most of the draw backs in the existing system are eliminated. They are

- First level of authentication i.e, login includes match with username and password along with one of the security question needs to be answered.
- To add the information to the list, password along with user name and website will added only if you provide the master password.

At the time of retrieval of information, only the required password along with its username will be displayed for the given website if you provide the correct master password for the given user name.

## 4. SYSTEM DESIGN

This use case diagram consists of two actors i.e, user and system. User can interact with the system with the help of different actions like login, register, store password, retrieve password and log out. User first needs to register in order to

have the access to the application then user can either store or retrieve the passwords in the list by logging in to the system. After each activity the user will be logged out. If user needs to perform one more action then he needs to login once again. Figure 2 Implemented using four classes such as user, database, site, encryption

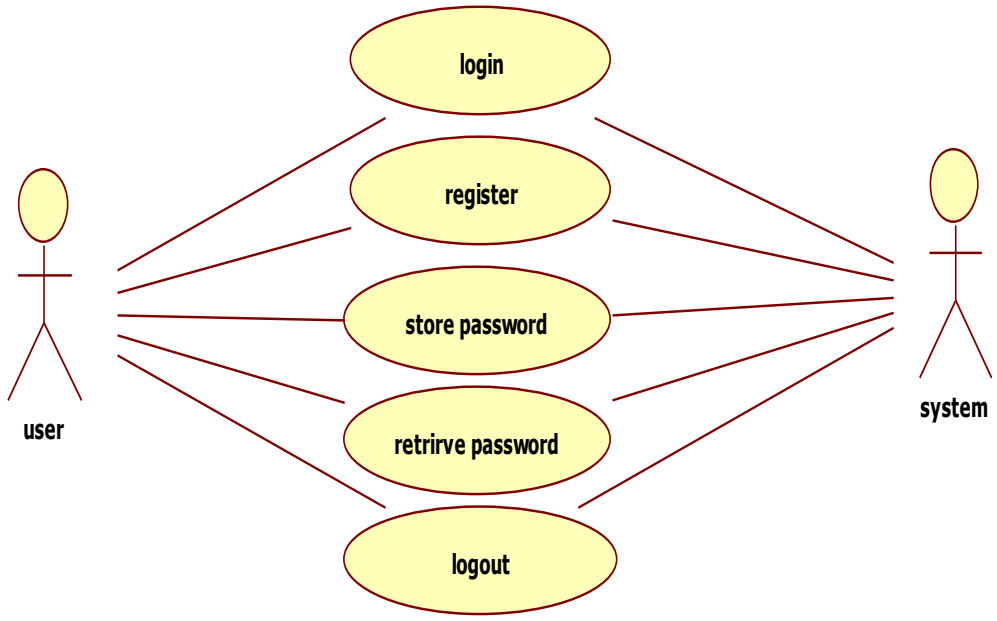


**Figure.1. case diagram**

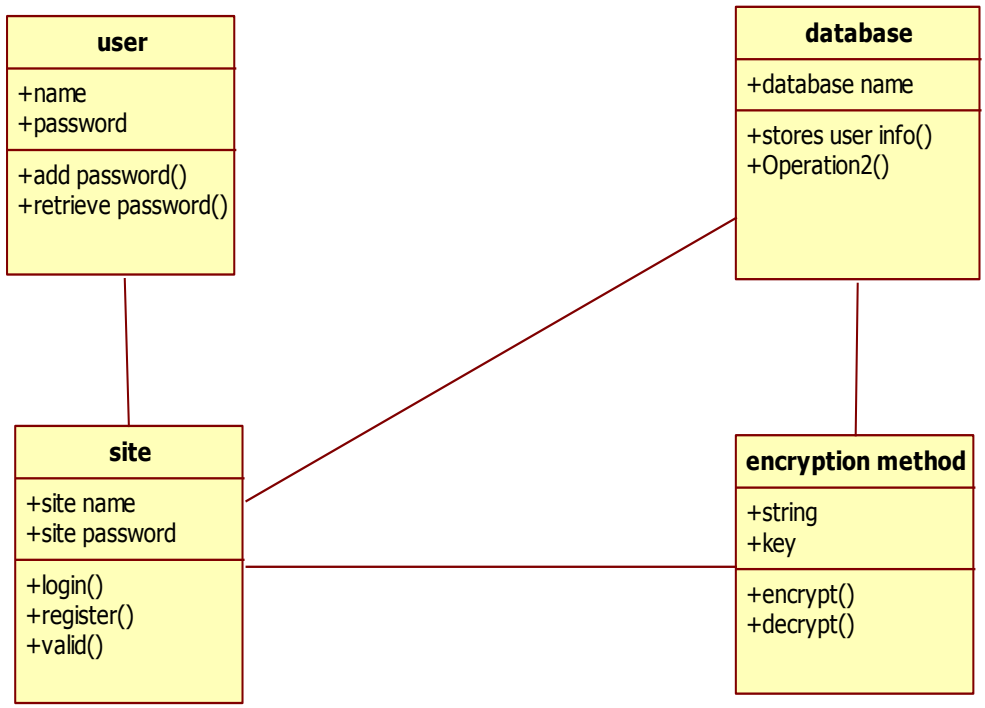


**Figure.2.class diagram**

Each class has its own operations and attributes. User class consists of attributes like user name and password. It can perform add password ( ) and retrieve password( ) operations. Database class consist of attributes like database name.. It can store user's information to the server and retrieve the required information to the system. Site class includes attributes like site name for which you want the user name and password, master password of this application. Site class can perform different operations like login( ), register( ) and valid( ) for validation of the user name and password. Encryption method is a class which includes the attributes like string and key which is to be encrypted and stored in the server. It can perform operations the data like encrypt ( ) and decrypt () this sequence diagram figure 2 consists of four objects and their interaction. User interact with the site with the help of register() then site stores the information to the database and returns to home page. From there user logs in through the function login(), which includes user name and password along with any one answer to the security question. Site compares these details with database, if matches then return to add

password or get password page. User enters the information to be stored, it goes from site to encryption method, where the data is encrypted. From there it gets stored in the database. At the time of retrieving the information, decryption takes place at the encryption method and returns back to the user.

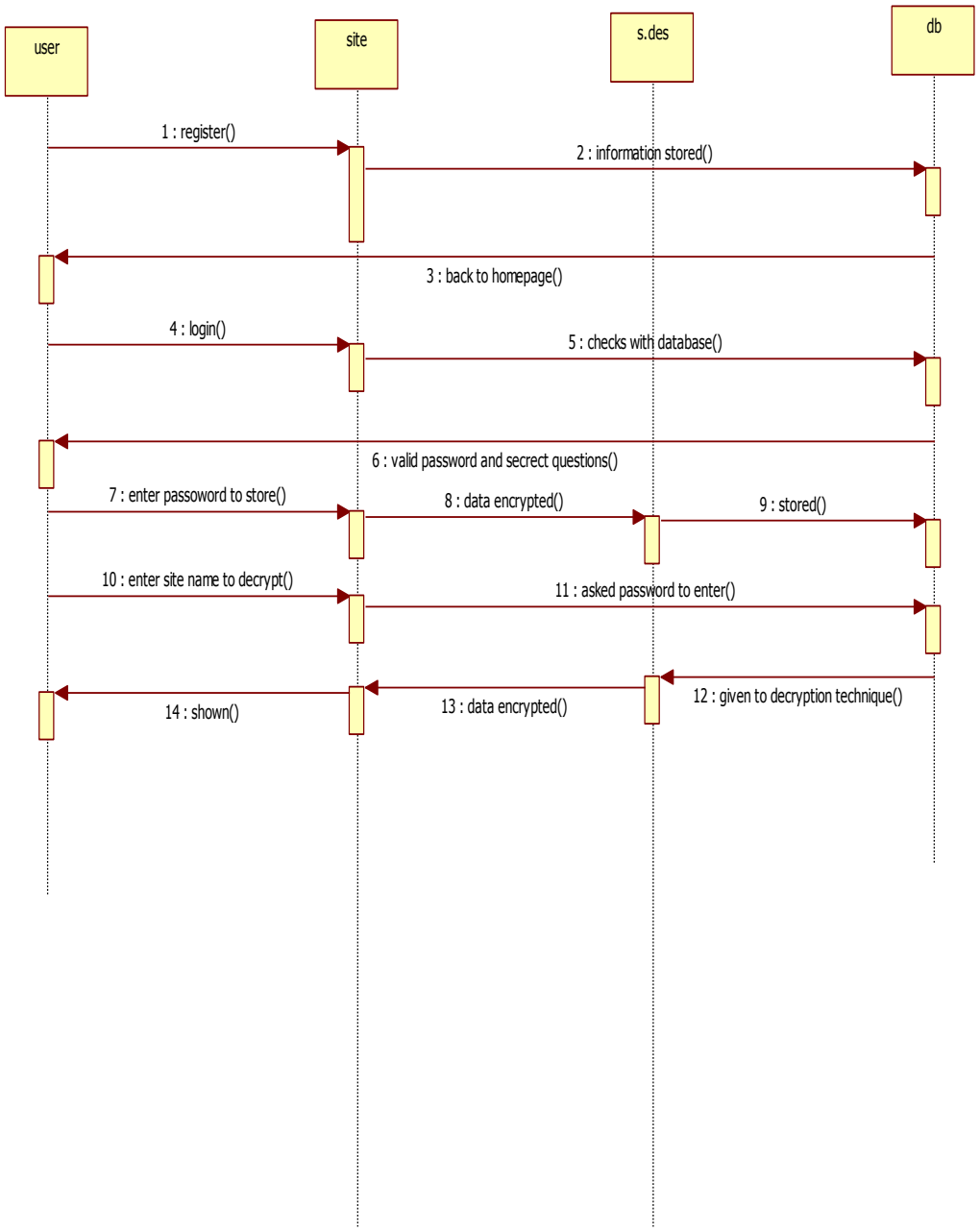


**Figure.3. sequence flow**

**Collaboration diagram figure 4**

1. Regitration
2. Stored
3. back to home page
4. Login
5. Checks with db
6. Verified
7. Enter pwd to store
8. Encryption
9. Stored in db
10. Retrieval
11. Ask for Pwd

**Results:**
Testing

Unique user name along with all the mandatory fields

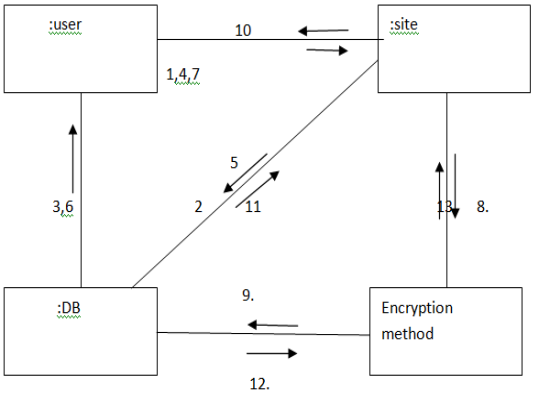


**Figure.4.collaboration diagram**

## 5. RESULTS

load source code to the wamp server and then run the wamp and project. First we get the login page, if you are new to the site, you need to go with the registration form first and after submitting , you can login with your username and password. Registration page consist of username and password along with 10 security questions, which have to be answered compulsory. In the login page, you need to enter user name and password along with one of the ten security questions. After validation, the next page will appear. In the next page, you will be given two options: either you can add a password in the list or you can retrieve the existing password from your list. If you go with "add a password in the list " ,then you need to enter website name, username, password and your master password. If you go with "get a password from the list", you need to enter the site name for the required password along with the master password, then you will get the user name and password for the given website.

**Registration:** Unique user name along with the entire mandatory field, If the user name already exist is the database

**Login page:** Login page with correct username, password and answer, Username is wrong while other two fields are correct, Username and password are correct and answer is wrong, Username and answer are correct while password is wrong

**Add page:** Master password not matches with the username logged in, Master password matches with username logged in

**Get page:** Master password not matches with the username logged, Master password matches with username logged

**Registration**

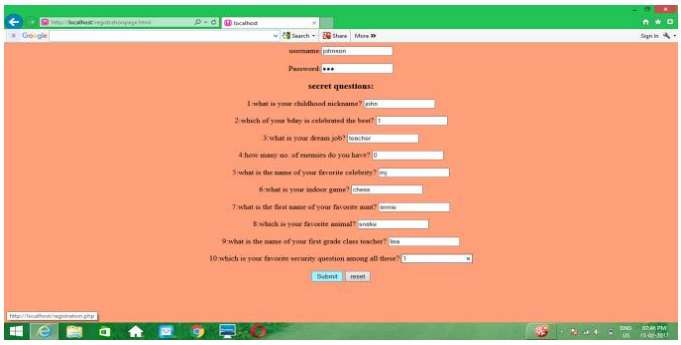


**Figure.5. Unique user name along with all the mandatory fields**

**login page:**

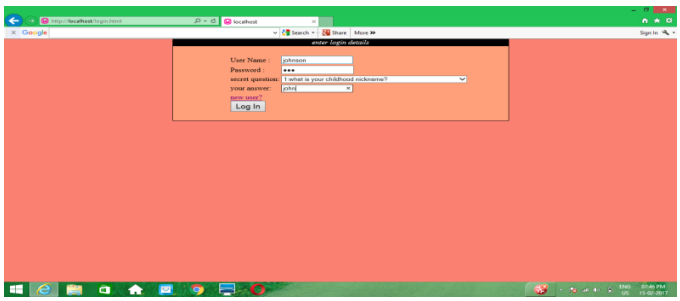


**Figure.6. login page with correct user name, password and answer**

**Add page:**

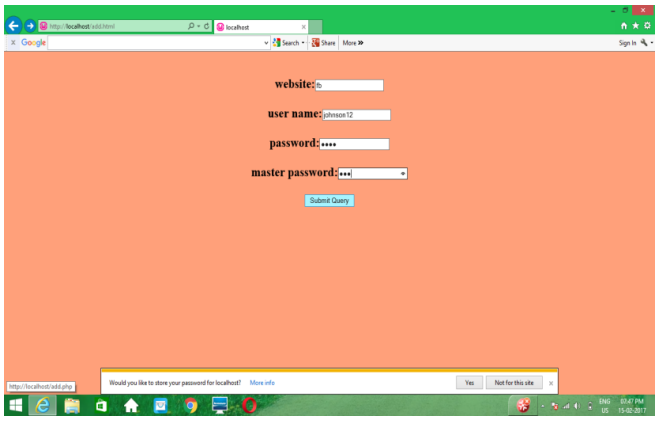


**Figure.7. master password matches with user**

**Get page:**

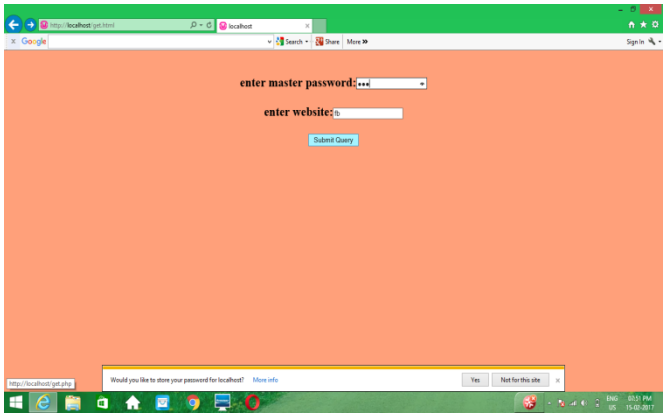


**Figure. 8. mater password matches user login**

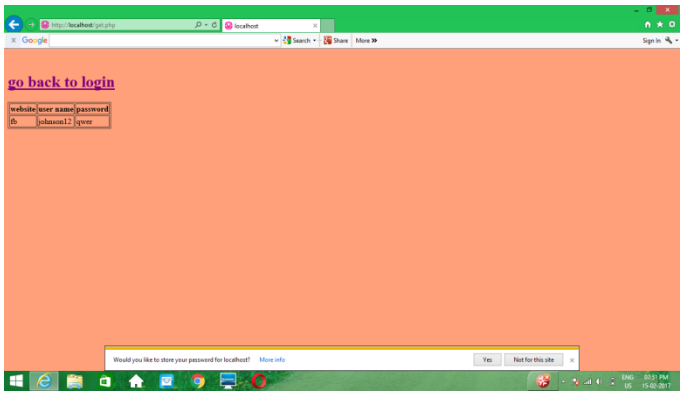


**Figure.9. final page.**

## 6. CONCLUSION

The proposed system easy to use, since it uses the GUI provided in the user dialog. User friendly screen are provided. The usage of software increases the efficiency, decreases the effort. It has been efficiently employed as a windows service .It also provides the user with variable options in storing passwords. It has been thoroughly tested and implemented .It performs that task of developing a web application, which provides asynchronous post backs by making use of PHP .Finally our work shows that it is indeed possible to construct a format that provides security, usability and low computation and storage overhead, using standard encryption methods. We

define realistic security models, design to represent the capabilities of real world attacks.

## 7. ACKNOWLEDGMENTS

I'm very grateful to all authors for using the valuable work as reference. I extent my thanks to all my teachers who help me though out my paper

## 8. REFERENCES

[1]. B. Pellin. Keepassdroid. http://www.keepassdroid.com.

[2]. A. Belenko and D. Sklyarov. \Secure Password Managers" and\Military-G rade Encryption" on Smart phones: Oh, Really? Technical report, Elcomsoft Co. Ltd., 2012. http:// www. elcomsoft.com/WP/BH-EU-2012-WP.pdf.

[3]. M. Bellare and C. Namprempre. Authenticated Encryption: Relations among Notions and Analysis of the Generic Composition Paradigm. J. Cryptology, 21(4),2008.

[4]. D. Bernstein. The Salsa20 Family of Stream Ciphers. In New Stream CipherDesigns - The eSTREAM Finalists. Springer- Verlag, 2008.

[5]. G. Blasko, C. Narayanaswami, and M. Raghunath. A Wristwatch-Computer Based

[6]. Password-Vault. Technical report, IBsM Research Division, 2005.

[7]. https://www.w3schools.com/php/default.asp