

U-Safe Using Bookmarklet

Minal Zope¹, Amar Wagh², Ganesh Mane³, Omkar Pisal⁴, Vinaysheel Wagh⁵

^{1, 2, 3, 4, 5} Department of Computer Engineering

^{1, 2, 3, 4, 5} AISSMS IOIT, Pune, India

Abstract- Now a days in government and private offices to register a service is very time consuming and vulnerable to fraud. As a result, government offices are not able to provide high-quality services to the citizens; while private offices are making very less productivity and profits from their services. The current automatic form filling systems are dedicated to only single websites or many of them are working with the static data handling so we propose this system which has following features. It is going to deal with automatic filling of application forms for universal websites with additional feature of uploading documents required by organization. The data which is going to be stored in database is in encrypted format for providing security of user's vulnerable data.

Keywords- Bookmarklet, automated form filling on website, document uploadation.

I. INTRODUCTION

Current practices in government and private offices to register a service are time consuming and prone to fraud. As a consequence, government offices are unable to provide high-quality services to citizen; while private offices make less productivity and profits from their services. The current auto form filling systems are dedicated to only single websites or many of them are working with the static data handling, so we propose this system which has following features. It is going to deal with automatic filling of application forms for any website with additional feature of uploading documents required by organization. The data which is going to be stored in database is in encrypted format and while retrieving the data from database it is decrypted and displayed back on the form this all procedure is for providing security of user's vulnerable data.

II. BACKGROUND

This section provides the background study used to develop U-SAFE project. It starts with the current practices that use paper based application forms and nowadays increase in number of the online forms. Then, the section presents briefly the use of the U-SAFE and its importance.

2.1. Paper Based and Online Application forms

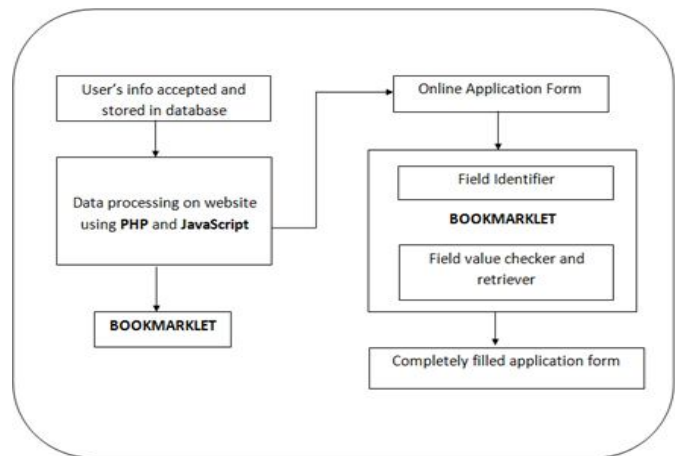
Paper based application forms are widely used in government sectors and the private offices. But nowadays the number of the online forms like college forms, Bank account opening form, etc. are increased. There are registration forms on many websites that are needed to be filled to use the resources of the website. Everything now is online so mostly all the people have to fill the forms each and every time. There is need of the automatic online form filling agent that will reduce the time and make the work easy. That's why U-SAFE is being proposed by us.

III. PROPOSED SYSTEM

The proposed system enables the service registration to be made quickly and securely. To realize this system, several steps have been followed and are described in this section. It begins with the diagram of Architecture of the proposed system.

3.1. Architectural Diagram

The given fig-3.1 is block diagram of our proposed system. As shown in given figure our system is divided into 2 parts in first part users whole information is accepted and bookmarklet is generated for particular form and in second part information which is required by application form is filled automatically by bookmarklet generated for that form.



3.2. System Description

The user first has to fill the registration form on the U-SAFE website. The user will fill all the information like his

personal information or we can say the static and dynamic data that is to be filled in online form. The user can save his personal information as well as he can upload the documents. After that he will be provided with the bookmarklet by us which will be used by the user for filling any online application forms automatically. The documents required by any application form can be provided by downloading already saved documents on U-SAFE website.

3.3. Algorithm of proposed system

The Steps are follows

Step1:

Register on the website.
Enter the Username.
Enter the Email.
Enter the password.

Step2:

Registration form

Step3:

Download the libraries for the Cryptojs.

Step4:

Get the values from form by their id's and store them in variables.
E.g. \$fn=document.getElementById('u_1').value;

Step5:

Store the iv(initialization vector)
Var iv =
CryptoJS.enc.Hex.parse('000102030405060708090A0B0C0D0E0F');

Step6:

Store the key for encryption
var key =
CryptoJS.enc.Hex.parse('2b7e151628aed2a6abf7158809cf4f3c');

Step7:

Pass the iv, key and the value to encrypt function and store encrypted value into variable.
var encrypted1 = CryptoJS.AES.encrypt(\$fn, key, {iv: iv});

Step8:

Pass encrypted data to the server
window.location.href =
"http://localhost/Document.php?name1="+encrypted1 +
"&name2=" +encrypted2+"&name3="

```
//+encrypted3+"&name4=" +encrypted4+"&name5="
+encrypted5+"&name6=" +encrypted6;
```

Step9:

Connect to the server and select database.
mysql_connect("\$host", "\$username", "\$password")or
die("cannot connect");

Step10:

Select the database
mysql_select_db("\$db_name")or die("cannot select DB");

Step11:

Get the values from the form
\$Firstname=\$_GET['name1'];
\$Middlename=\$_GET['name2'];
\$Lastname=\$_GET['name3'];

Step12:

Store the iv(initialization vector)
Var iv =
CryptoJS.enc.Hex.parse('000102030405060708090A0B0C0D0E0F');

Step13:

Store the key for encryption
var key =
CryptoJS.enc.Hex.parse('2b7e151628aed2a6abf7158809cf4f3c');

Step14:

Pass the iv, key and and the value to decrypt function and store decrypted value into variable.
var decrypted1 = CryptoJS.AES.decrypt(encrypted, key, {iv: iv});

Step15:

Insert into table
\$sql="insert into
\$tbl_name(id,Firstname,Middlename,Lastname)";
\$result=mysql_query(\$sql);
if(\$result){
echo "Successful";
echo "
";
echo "Back
to main page";
}
else {
echo "ERROR";
}
Retrieving from the database

Step16:

```

Connect to server and select database.
mysql_connect("$host", "$username", "$password")or
die("cannot connect");
mysql_select_db("$db_name")or die("cannot select DB");
$current_user =$_POST['email'];
$result1 = mysql_query("SELECT * FROM project where
Email='$current_user' ; ");
while ($row = mysql_fetch_array($result1)) {
$Firstname = $row['Firstname']."";
$Middlename = $row['Middlename']."";
$Lastname = $row['Lastname']."";

```

Step17:

```

Pass the variables to JavaScript file of websites and
generate bookmarklets.
$myfile = fopen("/var/www/html/facebook.js", "w") or
die("Unable to open file!");

```

```

$txt = "(function(){
document.getElementById('u_0_1').value = '$Firstname';
document.getElementById('u_0_3').value = '$Lastname';
document.getElementById('u_0_6').value = '$Email';
void(0)
})();";
fwrite($myfile, $txt);
fclose($myfile);
?>

```

IV. METHODOLOGY

4.1. Bookmarklet

Mr. Steve Kangas of bookmarklets.com have invented the word bookmarklet when he started to create scripts based on a suggestion in Netscape navigator's JavaScript guide. Before that Mr. Tantek Çelik called these scripts "favelets" and used that word as early as on 6 September 2001. Mr. Brendan Eich, who developed JavaScript at Netscape navigator, gave this account of the origin of bookmarklets.

Bookmarklets are small programs stored in Boemarks. Bookmarklets are shared on web pages as Web links to use a Bookmarklet We don't need to actually "install" it, you just need to simply add it to your bookmarks Bar. It just sit there patiently until you want to use it.

4.2. CryptoJS

CryptoJS is nothing but the collection of standard and secure cryptographic algorithms like AES, DES, and Hashers algorithm. All this algorithm are implemented in

Javascript. They are fast and they have a consistent and simple interface. Cipher Input and Output: The cipher input accept either string or instances of CryptoJS.lib.WordArray. The plain text you get back after decryption is WordArray object. It is a CipherParams object.

JS Code for AES is:

```

<scriptsrc="http://crypto-
js.googlecode.com/svn/tags/3.1.2/build/rollups/aes.js"></scrip
t><script> var encrypted
=CryptoJS.AES.encrypt("Message", "Secret Passphrase"); var
decrypted =CryptoJS.AES.decrypt(encrypted, "Secret
Passphrase"); </script>

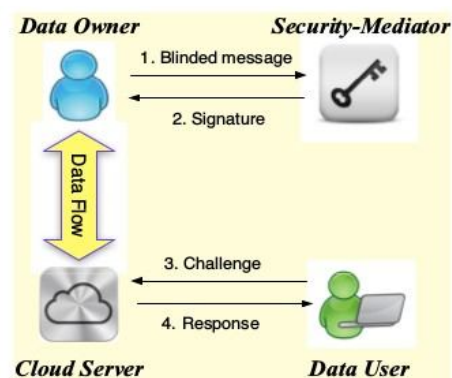
```

4.3. Cloud

The model introduced is mainly consists of following four entities:-

- Data owners
- Data users
- The cloud server
- Security mediator (SEM)

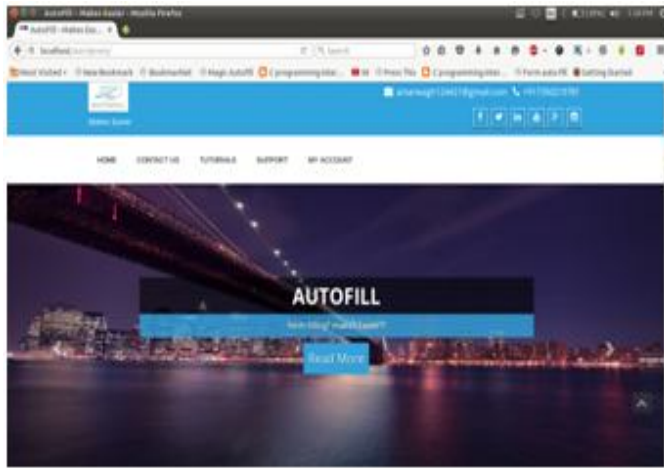
which are described in Figure 4.1. owners generate and upload data to the cloud for sharing. We will first look at the single owner scenario, which means each data file stored in the cloud is managed and maintained by only single owner. Users are able to access data uploaded by data owners, but they are not given permission to modify data in the cloud. owners and users are sometimes collectively termed as cloud users. The cloud server provides data storage and data sharing services to owners and users. Both the cloud server and users are public entities, who are not the owner of data but need to verify data integrity when it is necessary.



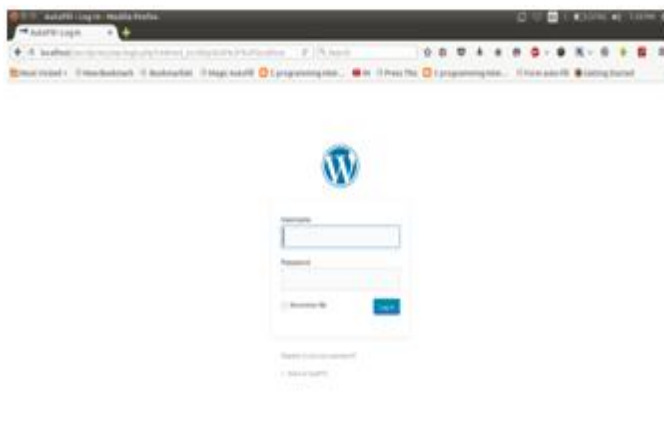
The system model includes data owners, data users, the cloud server, and a security-mediator (SEM).

V. RESULTS

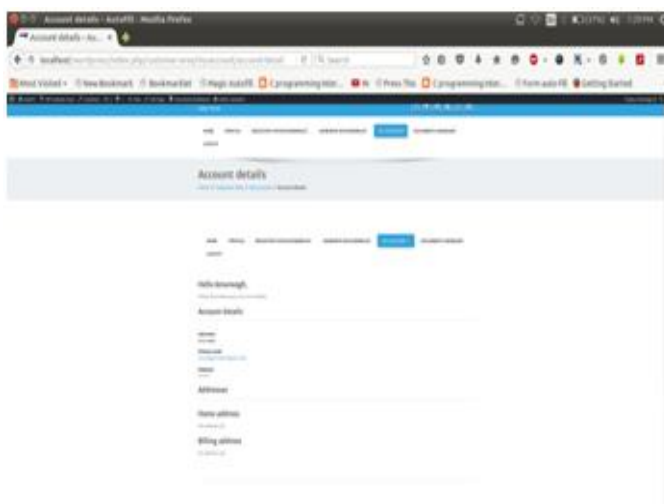
1) Home page of Web Site:-



2) New user registration/login page:-



3) Logged in users homepage/Account page:



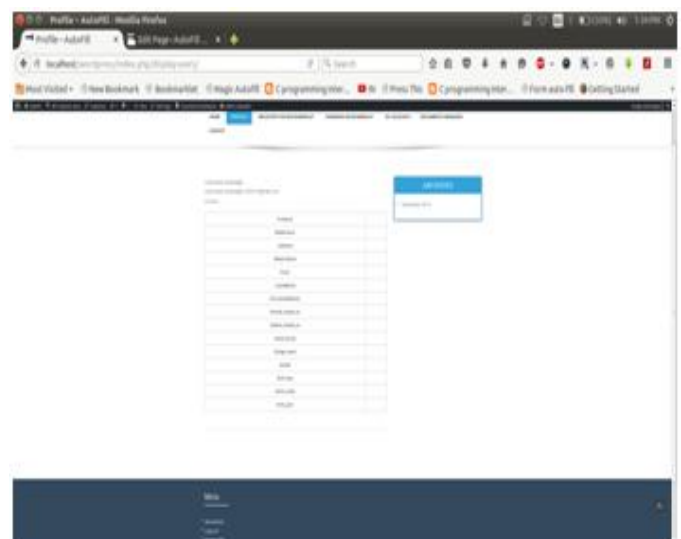
4) Registration Form:-



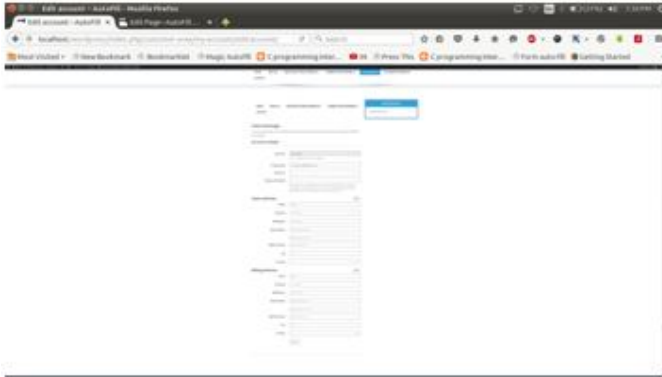
5) Registration form after filling information:-



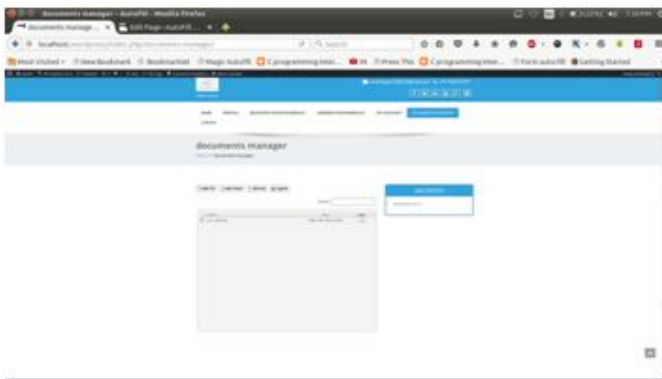
6) Profile page of the user:



7) Document Manager for user:-



8) User Account editing page:-



VI. CONCLUSION

This system is an innovative web based application that can be employed while registering on digital application form with the facility of uploading the required documents. By utilizing the user's information stored at back-end database, the developed auto filling form system improves the productivity and security of registration procedure.

ACKNOWLEDGMENT

We would like to thank all those who somehow have given a contribution to this project and taking this project on the path of success.

REFERENCES

- [1] Yahaya Abd Rahim, "A Secure and Fast Auto Filling Form System", 2013 IEEE Symposium on Industrial Electronics & Applications (ISIEA2013), September 22-25, 2013, Kuching, Malaysia..
- [2] Rishabh Jain., Rahul Jejurkar , "AES Algorithm Using 512 Bit Key Implementation for Secure Communication". International Journal of Innovative Research in Computer

and Communication Engineering (An ISO 3297: 2007 Certified Organization) Vol. 2, Issue 3, March 2014

- [3] Ning Oiu, GuoyongDai, "Design and Implement of Online Intelligent Form Filling" 2012 9th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD 2012).
- [4] Rakesh Chawla, Puneet Dhiman ,Sudhanshu Prakash Tiwari "A SSL Based Approach to Enhance Security and QOS in Grid Networks"2012 Second International Conference on Advanced Computing & Communication Technologies.
- [5] Quist-Aphesti kester "A Novel Cryptographic Encryption technique for securing Digital Images in the cloud using AES and RGB pixel displacement" 2013 European Modelling Symposium.
- [6] Dongliang Lei, Ke Zhou, Hao Jin, Junping Liu, Ronglei Wei SFDS: A Security and Flexible Data Sharing Scheme in Cloud Environment".2014 International Conference on Cloud Computing and Big Data.
- [7] Boyang Wang, Sherman S.M. Chow, Ming Li, and Hui LiStoring, "Shared Data on the Cloud via Security-Mediator".ICDCS '13 Proceedings of the 2013 IEEE 33rd International Conference on Distributed Computing Systems, 124-133,2013.