Image Based Authentication System with Sign-In Seal*

Nitin[1], Vivek Kumar Sehgal*[2], Durg Singh Chauhan[3], Munish Sood and Vikas Hastir
[1,2,3]*Member, IEEE* and *ACM*

Abstract—**A distributed system provides user with the ease of being accessible globally through the internet. Our Image based Authentication system provides exactly that. JUIT-IBA system provides a host of features that makes it user friendly and one of the most advanced Image Based Authentication System till date. This paper is a study of the various new features that have been added to the system. The system now provides a fully functional interface for the faculty, which is different from that of the user. Advanced security features like Sign In seal have been introduced to make the system more secure. Making user familiar with the concept of images is also important, so a training facility is also provided.**

Index Terms— **Ajax, Image Based Authentication System, Sign-In-Seal, PHP**

I. Introduction and Motivation

Authentication plays an important role in protecting resources against unauthorized use. Many authentication processes exist from simple password based authentication system to costly and computation intensive biometric authentication systems [3, 4, 5, 6, 7, 8]. Passwords are more than just a key. They serve several purposes. They authenticate us to a machine to prove our identity-a secret key that only we should know. They ensure our privacy, keeping our sensitive information secure. They also enforce non-repudiation, preventing us from later rejecting the validity of transactions authenticated with our passwords. Our username identifies us and the password validates us. However, passwords have some weaknesses: more than one person can posses its knowledge at one time. Moreover, there is a constant threat of losing your password to someone else with malicious intent. Password thefts can and do happen on a daily basis, so we need to protect them. Now merely using some random alphabets grouped together with special characters does not ensure safety. We need something new, something different as our password to make it secure. Besides being different, it should also be easy enough to remembered by you and equally difficult to be hacked by someone else. This is what Image Based Authentication system provides you with.

The human brain is more adept in recalling a previously seen image than a previously seen text. In a recent user study conducted at University of California at Berkeley, image based authentication (IBA) systems have been found to be more user-friendly than the usual text-based systems. Besides being user friendly, we need to strengthen the security during authentication also. This is done using the Kerberos protocol [9-15, 29-37]. We have already developed an Image Based Authentication System known as the JUIT-IBA System.

The paper is organized as follows: Section II deals with the new features built in the system. It contains details of all features along with the screenshots and algorithms followed by the conclusion and the references.

II. Features

A. Signup Process

To enroll with the IBA system, one has to sign up the login form wherein user has to fill in the required details. The users who have already been authenticated manually by the administrators can exercise the login process. This is done by feeding the unique roll numbers and the email ids in the authentication server beforehand. Next, the user initiates filling up the sign up page. If the user is authenticated, i.e. the entered roll number matches one of those already fed in the authentication server then the corresponding email id is displayed. Username selected by the user should again be distinctive and should be chosen according to the requirements of the designer. If the needful information has been submitted successfully, the user gets a confirmation mail on his/her email id, which was submitted by hand. The confirmation mail contains a verification code and a link to the page from where the user is permitted to decide on his new set of images as password. The clicking of mouse on images does not highlight the selected image. Instead each click sets off the calculation of md5 (hashing algorithm) to generate a 32-byte encrypted code of the file (each image is considered to be a file) and stores the encrypted text in a table. On choosing the images once, the user has to confirm his selected password by re-selecting the same images. Note the order of images does not matter. Finally, if the login has been done successfully the user is all set to use the system.

Nitin, is with Department of Computer Science Engineering and Information Technology, Jaypee University of Information Technology, Waknaghat, Solan-173215, Himachal Pradesh, INDIA (E-mail: delnitin@juit.ac.in, delnitin@ufl.edu and delnitin@gmail.com).
Vivek Kumar Sehgal, Vikash Hastir and Munish Sood are with Department of Electronics and Communication Engineering, Jaypee University of Information Technology, Waknaghat, Solan-173215, Himachal Pradesh, INDIA (E-mail: vivekseh@gmail.com).
Durg Singh Chauhan is Presently Vice Chancellor and Professor in Department of Computer Science Engineering and Information Technology, Jaypee University of Information Technology, Waknaghat, Solan-173215, Himachal Pradesh, INDIA (E-mail: pdschauhan@gmail.com).
*This Image Based Authentication System has been developed for Jaypee University of Information Technology (JUIT). It has been developed using Scripting languages. It is uses PHP (ver. 5) [6, 20, 21, 22] and MySQL and AJAX [6, 16-21] has also been used extensively. It is up and running within our university and is globally accessible through the website www.juit-iba.org.Two papers [1, 2] for this system have already been published with IEEE. One paper analyses the security of the system and second one provides a view of the Finite State Automata of the system. In earlier papers, we also compared our system with the UFL-IBA system and showed how our approach is better than them. This paper is a continuation of the earlier papers and describes the new work done on the JUIT-IBA System. The future work mentioned in the earlier paper has been implemented and many new features have been introduced.

Algorithm: SIGNUP_PHP
Step 1: Select all rows from server where roll number is equal to the entered roll number.
Step 2: Store the above query inn variable '$as'.
Step 3: Execute the query using function mysql_query().
Step 4: If the fetched result contains rows greater than zero then the email off the user is checked against the registration status of the user.
Step 5: If the user is not registered then all the details are inserted into the database.

Algorithm: CONFIRM_PASSWORD_PHP
Step 1: The values of the variables username, the activation code and the email id get posted to CONFIRM_PASSWORD.PHP.
Step 2: Email with the activation code is sent to the user.
Step 3: Clicking on the activation code redirects the user to PASSWORD.PHP where the user can select the password.

B. Student Interface

1) Submissions: This module facilitates students to submit their assignments online. The submissions are accepted till a specific date and time after the expiry of which the links to upload assignment is deactivated. Once more, the submission of the assignments should be done as per the requirements of the designer. Following the submission of the assignment, the user gets a confirmation mail saying that the assignment has been successfully uploaded. Besides all this, the module also calculates md5 of the filename thereby making it trouble-free for the administrator (course in-charge) to differentiate between duplicate copies of the same assignment

Algorithm: UPLOAD_FILE.PHP:
Step 1: The file to be uploaded is checked for its extension and size. Only files having 'zip', 'tar' or 'rar' extensions and size not exceeding 5 MB can be uploaded.
Step 2: If the user then the script rapports do not follow the above constraints an error.
Step 3: The file name also has a constraint stating that there should not be any blank spaces present in its name.

2) Personal Information (fig.1): JUIT-IBA tool also allows students to view their personal information wherein they can upload/update their picture. Their personal information is extracted from the sign up page.

3) Training (fig.2): The training module facilitates students to get accustomed with the selection of images as their password set. This feature allows users to remember their password by giving them a practice of selecting the images in their password set in a span of 180 sec.

4) Forgot Password: Forgot password makes it handy for users off JUIT-IBA system to select new set of images as their password set if they forget the old ones. Unquestionably, this module requires users to remember their username (private key) and the answer to the secret question they had filled while signing up the login form. On entering the information correctly, the user receives a confirmation mail and a link, which activates your account thus letting user to select a completely new set of password.

Fig.1. Personal Information Page

```
function grid(str)
{
xmlHttp=GetXmlHttpObject()
if (xmlHttp==null)
 {
 alert ("Browser does not support HTTP Request")
 return
 }
var url="password_grid.php"
url=url+"?set="+str
url=url+"&sid="+Math.random()
xmlHttp.onreadystatechange=gridisp
xmlHttp.open("GET",url,true)
xmlHttp.send(null)
}
function gridisp()
{
if (xmlHttp.readyState == 1)
{
                                document.getElementById("displ
ay").
                                innerHTML="<embed
src=black.swf quality=high width=20 height=20></embed>";
}
if                  (xmlHttp.readyState==4                  ||
xmlHttp.readyState=="complete")
 {
 document.getElementById("grid").
 innerHTML=xmlHttp.responseText;
document.getElementById("display").
                                innerHTML="";
 }
}
```

Fig.2. AJAX Script explaining the working of the JUIT-IBA training system

5) Forgot Username (fig.3): Apart from the password selected the username also acts as a private secret key. So in cases user forgets it, this component of the JUIT-IBA tool will be of help. The requirement is more or less same as that of Forgot Password except for the fact that this module requires users to enter their date of birth as part of the verification process. Subsequently user receives an email, which informs the user of his/her username.

Fig.3. Forgot Username Interface

6) Sign-In Seal (fig.4): A sign-in seal is a secret between the computer you set it up on and IBA. Therefore, when you sign in to IBA from this computer, your sign-in seal tells you that you are seeing a genuine IBA site, not a phishing site. Your sign-in seal is associated with your computer, not your ID. It is a convenient way to instantly recognize a genuine IBA sign-in page and be sure that you are not on a page created by fraudsters attempting to steal your IBA ID and password. Because we associate your sign-in seal with your computer, after you create as seal, there are no additional steps to signing in. Even if a hacker knows or guesses your ID or other personal information, they cannot use it to discover your sign in seal. You can customize your seal either by creating a text seal or by uploading an image.

Algorithm: SEAL_PHP
Step 1: User is asked to select either an Image based Sign-In Seal or a Text based Sign-In Seal.
Step 2: If the Image based Sign-In Seal is selected then the user is required to upload an image with extension jpeg or gif and size not exceeding 2 MB. If Text based Sign-In Seal is selected then the user is required to enter a relevant text.
Step 3: The image or the text is set as the value of the cookie and is displayed on the sign in page of that particular user

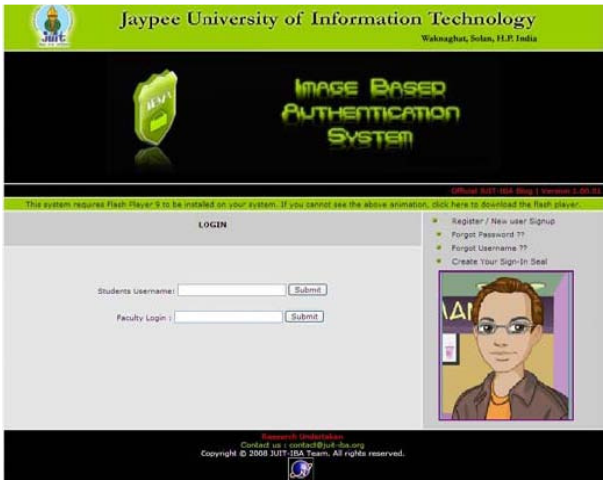

Fig.4. Sign-In Seal on the sign in page.

C. Faculty Interface

1) Read Notices: Module designed to facilitate a basic formal && official communication system of notices. Module is common to both faculty & students. On home page, only most recent five notices are displayed & rest of them can be read through READ ALL NOTICES option. Only list of titles & date is posted & details of each can be seen by clicking on its title.

Apart from this, we have a system of removing the notices automatically when it expires. That is, a notice is automatically deleted from the database when it needs to be deleted.. The following code shows the list of notices i.e. display the title & author. When a user clicks on a particular title, its contents are displayed.

2) Post New Notices (fig.5): Here, faculty is provided with an interface where he/she can post new notices. This goes into a database from which above mentioned scripts extract title etc.

Algorithm: NOTICE__PHP
Step 1: The values off the variable title, content, expiry date, faculty's name and designation are posted to NOTICE.PHP
Step 2: The Current date is fetched using the function getdate().
Step 3: If the date of expiry is older than the current date then the corresponding notice gets deleted from the database.
Step 4: The new notice along with the details is inserted into the database

3) Upload Assignments (fig.6): Using this module, faculty uploads assignments in the respective subjects. As soon as the upload is finished, all students can download && submit the solution (before the deadline –which is set by the faculty himself). While uploading, faculty has to follow some instruction as follows:

1) While submitting your assignments/tutorials or project that contain multiple files, first zip them into one .zip/.tar file.
2) Choose your filename appropriately. Name your file as Assiggnment_x.zip or Project_x.zip or Project_x.tar where 'x' is a number 1, 2 & so on signifying the assignment number. NOTE: Spaces not allowed in the filename
3) DO NOT USE .RAR EXTENSION.

The upload form contains the following fields:

1) Subject 2)
2) File / Filename

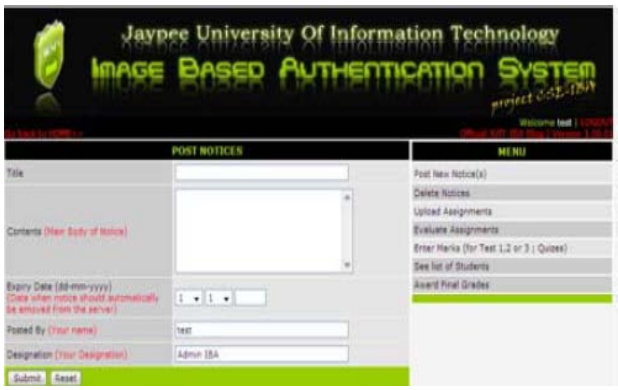

Fig.5. Post New Notices Column

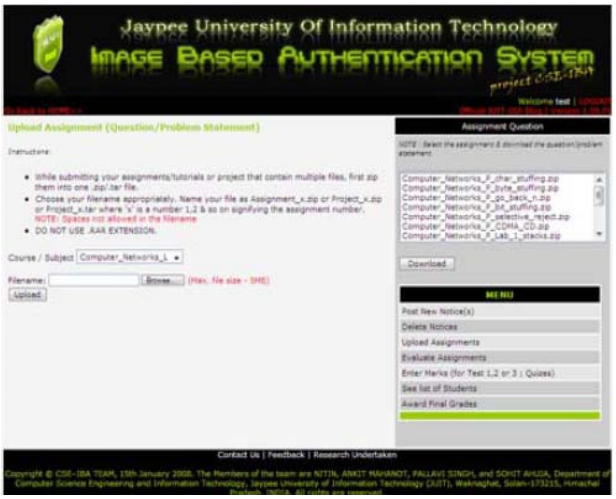

Fig.6. Uploading Assignments

4) Evaluate Assignment: Using this feature, a faculty can evaluate the solutions that have been uploaded by all the students. Firstly, the faculty has to select the assignment & the subject for which he wishes to evaluate.

Then to evaluate a particular assignment, AJAX is used to enter marks/grade of each student. In this, a text box is provided, where the faculty just inputs the grade/marks & they are automatically recorded in the database. All this is done parallel, i.e. without refreshing the whole page. Faculty can download the assignment from the link provided against the name of each student. NO SUBMISSION is displayed in front of the defaulters and is automatically awarded a N.A. grade.

5) Enter and Display Marks: This module was designed to give marks to students for their Quizzes, Minors etc. In this, faculty can enter the marks for each student against his name, which is viewable by the student (using his/her interface). In this thee faculty is provided with the list of all the students registered under a particular course. Against each student, a text box is given in which grades or marks are entered.

After the marks have been put, faculty can print this list by clicking on PRINT VIEW option, which can be displayed on notice board as a hardcopy. In addition, each student gets a popup under the UPDATE section regarding the same. AJAX is used while entering the marks i.e. all computation & database updating is done in parallel.

III. Conclusion and Future Scope

The new feature called Sign-In Seal added to the JUIT-IBA system makes it highly secure. In near future not only we will add more features but also make our system customizable.

References

[1] Nitin et. al.: "Security Analysis and Implementation of JUIT— Image Based Authentication System Using Kerberos Protocol," icis, pp. 575-5800, Seventh IEEE/AACIS International Conference on Computer and Information Science (icis 2008), 2008.

[2] Nitin et. al.: "Finite-State Modeling and Testing of Image Based Authentication System," icis, pp. 427-432, Seventh IEEE/ACIS International Conference on Computer and Information Science (icis 2008), 2008.

[3] Faulkner, Information Services: Enterprise Network Security Guidelines: Prevention and Response and Hacker Attacks, Digital Edition, June 1, 2001.

[4] Vijay K. Bhargava, H. Vincent Poor, Vahid Tarokh, and Seokho Yoon, Communications, Information and Network Security (The Springer International Series in Engineering and Computer Science), Hardcover, December 31, 2002.

[5] Security in Distributed and Networking Systems (Computer and Network Security) by Yang Xiao (Hardcover - Sep 30, 2007).

[6] Cristian Darie, Bogdan Brinzarrea, Filip Cherecches-Tosa, and Mihai Bucica, AJAX and PHHP: Building Responsive Web Applications, Paperback, March 1, 2006.

[7] Melcher,, "The persistence of visual memory for scenes," Nature, 412(68445) pp. 401, July 2001.

[8] Rachna Dhamija and Adrian Perrrig, "A user study Using Images for Authentication," Proceedings of the 9th Usenix Security Symposium, August 2000.

[9] William Stallings, "Cryptography and Network Security," Pearson Education.

[10] http://www.Kerbberos.info.

[11] http://en.wikipeddia.org/wiki/Kerbeeros.

[12] Jason Garman, Kerberos: The Definitive Guide, Paperback, August 26, 2003.

[13] Brian Tung, Kerberos: A Network Authentication System, Paperback, May 4, 1999.

[14] B. Clifford Neumann and Theodore Ts'o, Kerberos: An Authentication Service for Computer Nettworks, IEEE Communications, 32(9) pp33–38. September 1994.

[15] John T. Kohl, B. Clifford Neumman, and Theodorre Y. T'so, The Evolution of thee Kerberos Authentications System. Distributed Open Systems, pp78–994. IEEE Computer Society Press, 11994.

[16] www.asp.net/ajaax/

[17] www.ajax.org/

[18] www.java.sun.com/developer/techhnicalArticles/J2EEE/AJAX/

[19] Chris Ullman annd Lucinda Dykes, Beginning Ajax ((Programmer to Programmer), Paperback, March 119, 2007.

[20] Cristian Darie, Bogdan Brinzarrea, Filip Cherecches-Tosa, and Mihai Bucica, AJAX and PHHP: Building Responsive Web Applications, Paperback, March 1, 2006.

[21] Lee Babin, Beginning Ajax with PHP: From Novice to Professional, Paperback, October 16, 2006.

[22] http://en.wikipeddia.org/wiki/PHP

[23] Win van Eck, "Electromagnetic Radiation from Video Display Units: An Eavesdropping Risk?"" Computers and Security, vol. 4, pp. 269-286, 1985.

[24] Markus G. Kuhn, "Electromagnetic Eavesdropping Risks of Flat-Panel Displays," Proceedings of the 4th Workshop on Privacy Enhancing Technologies, May 2004.

[25] http://dev.mysql.com/doc/refman/5.0/en/tutorial.html

[26] http://www.tutorialspoint.com/mysql/

[27] http://dev.mysql.com/tech-resources/articles/mysql_intro.html

[28] Simon Blake-Wilson and Alfred Menezes, "Authenticated Diffie-Hellman Key Agreement Protocols", Lecture Notes in Computer Science, vol 1556 pp. 339-361, Jan 1999.

[29] M. Burrows, M. Abadi, R.M. Needham: A logic of authentication, Proceedings of the Royal Society of London A 426, 1989. A short version appeared in the Proceedings of the 12th Symposium on Operating System Principles, ACM, 1989.

[30] D.E.R. Denning, G.M. Sacco: Timestamps in Key Distribution Protocols, CACM, 24(8), August 1981.

[31] S.M. Bellovin, M. Merritt: Limitations of the Kerberos Authentication System, ACM Computer Communications Review, 20(5), October 1990.

[32] L. Gong: A Security Risk of Depending on Synchronized Clocks, ACM Operating Systems Review, 26(1), 1992.

[33] R.M. Needham, M.D. Schroeder: Using Encryption for Authentication in Large Networks of Computers, CACM, 21(12), December 1978.

[34] http://web.mit.edu/Kerberos/papers.html#k5-protocol

[35] J. G. Steiner, B. Clifford Neuman, and J.I. Schiller. Kerberos: An Authentication Service for Open Network Systems. In Proceedings of the Winter 1988 Usenix Conference. February 1988.

[36] S.P. Miller, B.C. Neuman, J. I. Schiller, and J.H. Saltzer. Section E.2.1: Kerberos Authentication and Authorization System. Project Athena Technical Plan, MIT Project Athena, Cambridge, Massachusetts, October 1988.

[37] Marlena E. Erdos and Joseph N. Pato. Extending the OSF DCE Authorization System to Support Practical Delegation. In Proceedings of the 1993 PSRG Workshop *on Network and Distributed System Security*, February 1993.