Design and Implementation of a Two-Factor, One Time Password Authentication System

TokulaUmaha I. and EsiefarienrheBukohwo Michael
Math/Statistic/Computer Science Department
University of Agriculture, Makurdi, Benue State, Nigeria

***ABSTRACT --** Most people now access all the important areas of their life—banking, shopping, insurance, medical records, and so on—simply by sitting at their computer and typing a username and password into a website. Getting access to something this way is called one-factor authentication, because you need to know only one thing to get into the system: the combination of user name and password. In theory, this kind of protection should be reasonably secure; in practice, it's less and less trustworthy. This paper presents an approach to further increase security using a two-factor authentication scheme. This approach required the user to login with a username and password and also generate a One Time Password which will be sent to his email. The One Time Password will be used for authentication any time the user wishes to access a restricted resource. The one time password as the name implies will expire after a single use and after a period of 60 seconds. The system uses the HMAC-SHA-256 algorithm to develop a more secured two factor, one time password. Java Enterprise Edition (JEE) technology and MySQL was used and the frontend and backend respectively and was deployed on a single user computer using Java Bean Open Source Software (JBOSS) application server. The results from the system implementation show a more secured system difficult to compromise.*

***KEYWORDS --** One Time Password (OTP), HMAC-based One Time Password (HOTP), Time-based One Time Password (TOPT), Cryptography, Email, Authentication*

I. INTRODUCTION

Private and sensitive information about everyday life is becoming more and more stored and passed across on the internet. People can now access such important data about all areas of their life — banking, shopping, insurance, medical records, and so on— simply by sitting at their computer and typing a username and password into a website. However, there is the risk that such data can be intercepted and stolen by malicious software. It is therefore necessary to authenticate users. If a user is authenticated using the same password in every session, the password can be stolen in several ways including using keystroke logging programs that sit quietly on your computer remembering all the keys you press, including any passwords you enter. Yet another way of stealing passwords is the man-in-the-middle attack, in which an attacker sits in the data flow of a communication, masquerading as the sender to the receiver, and vice versa. Another well-known technique is called a dictionary attack, where a hacker can program a computer to log into someone's system by brute force, trying a list of common words (or names) as passwords, one after another, until it hits the right one by chance. In 1981, Lamport proposed a One Time Password (OTP) authentication scheme using cryptographic hash functions. The purpose of an OTP is to make it more difficult to gain unauthorized access to restricted resources (Niharika and Rama, 2015). Therefore, the user requires a one-time password (OTP) that is valid for only one login session. A new password will be required for the next login.

II. LITERATURE REVIEW

The concern of user authentication as well as authorization in public network has always been a matter of concern in the area of computer networking as well as security system. Authentication is the method of verifying the user while authorization is the methods of verifying that user have an access to resources (Humaira Dar *et al,* 2013).

Various researchers have work on data security system and remarkable work has been done. Ayushi (2010) presented a new symmetric algorithm designed specifically for a minimal amount of data. Various cryptographic algorithms have been developed to achieve goals like Confidentiality, Data integrity, Authentication etc. He explains that for a very minimal amount of data those algorithms wouldn't be cost effective since they are not designed for small amount of data. His proposed algorithm was designed for small amount of data, in a much simplified manner but ended up compromising for security.

Mansoor*et al* (2013) presented a detailed analysis of symmetric block encryption algorithms on the basis of different parameters. The algorithms analyzed are DES, 3DES, Blowfish, IDEA, TEA, CAST, AES

(Rijndael), RC6, Serpent, Twofish and MARS. The main objective was to analyze the performance of these algorithms in terms of Authentication, Flexibility, Reliability, Robustness, Scalability, Security, and to highlight the major weakness of the mentioned algorithms, marking each algorithm's strength and limitations. During this analysis it was observed that AES (Rijndael) was the best among all in terms of Security, Flexibility, Memory usage, and Encryption performance. Although the other algorithms were also competent, most of them have a tradeoff between memory usage and encryption performance.

Ranjeet*et al* (2014) did an analysis of different symmetric key algorithms for various features like data type, data density, data size and key size, and analyzed the variation of encryption time for different selected cipher algorithms. From the simulated results they concluded that encryption time does not depend on data type and data density of the file, rather, encryption only depends on the number of bytes present in the file. They also showed that encryption time and data size is proportional to each other. As the size of data increases the encryption time also increases proportional to data size and vice versa.

Prashant*et al* reviewed asymmetric key algorithms RSA , DSA, ECC, Diffie-Hellman and ElGamal. RSA is the most widely used public key technology today but the use of simpler connected devices and demand for higher level of security will make continued reliance on RSA more challenging over time. These trends highlight a clear need for an efficient public key cryptosystem that can lower the capacity threshold for small devices to perform strong cryptography and increase a server's capacity to handle the secure communication. The RSA keys will need to grow to 2048 bits. Public-key schemes can provide all functions required by modern security protocols but the major drawback in practice is that encryption of data is extremely slow with public key algorithms. Many block and stream ciphers can encrypt about one hundred to one thousand times faster than public key algorithms. Thus somewhat ironically, public key cryptography is rarely used for actual encryption of data. On the other hand, symmetric algorithms are poor at providing non-repudiation and key establishment functionality. In order to use the best of both worlds, most practical protocols are hybrid protocols which incorporate both symmetric and public key algorithms. An example is the SSL/TLS protocol that is commonly used for secure web connection.

Furthermore, Nivedita and Sapna (2015) did a comparative study of encryption techniques in terms of symmetric key and asymmetric key algorithms. Their work showed that symmetric key algorithms is viewed to be good in terms of speed and power consumption while asymmetric key algorithms in terms of tenability. In the symmetric key encryption AES algorithm was found to be better in terms of cost, security and implementation. In asymmetric key encryption RSA algorithm is better in terms of speed and security.

Hongfeng*et al* (2015) proposed a novel and complete biometrics-based and one-time identity-password authentication scheme for e-coupon systems. Their scheme used biometrics method and dynamic ID-password to achieve high-level security. The core ideas of the proposed scheme are the features of security and efficiency in the mobile device and server's side, and the user friendliness on the user's side.

Ahmad (2013) presented a novel approach to combat Phishing attacks. He proposed an approach where a user will retrieve the one time password by SMS or by alternate email address. After receiving the one time password the web server will create an encrypted token for the user's computer/device for authentication. The encrypted token will be used for identification, any time user wishes to access the website he/she must request the new password.

Ankit*et al* (2015) presented a three level authentication system based on cued click points in an image. The user- friendly System named as 3 Level Security ensures its security through three levels– which are Text based password, image based password and One Time Automated Password. If the user's click on the image is within the level of tolerance, the user is presented with the next proper image, otherwise the user is shown a random image, which an authentic user will immediately come to know about. Here, the hacker won't get a clue about a correct or an incorrect image, whereas an authentic user will know about it and he/she can restart the authentication process.

III. METHODOLOGY

The methodology used in this research is the Object Oriented approach. The discussion of this approach follows:

Stage-1: User Profile Registration: Before a user can be successfully authenticated on the system, he must be registered. The new user information includes first name, last name, username and password. At this stage, the password is encrypted before being saved in the database and a secret key is generated from the password using a Password-based Key Derivation Function called HMAC-based Extract-and-Expand Key Derivation Function (HKDF) specified in RFC 5869.

Stage-2: Static Password authentication (one factor authentication): This stage involves the simple and popular login scheme used by many systems. Here the user enters his username and password. If they are correct, he moves on to the next stage of authentication, else, he is redirected to the login page.

Stage-3: One Time Password authentication (two factor authentication): In this stage, an OTP is generated and sent to the user as described below. The OTP authentication is shown in the activity diagram below.

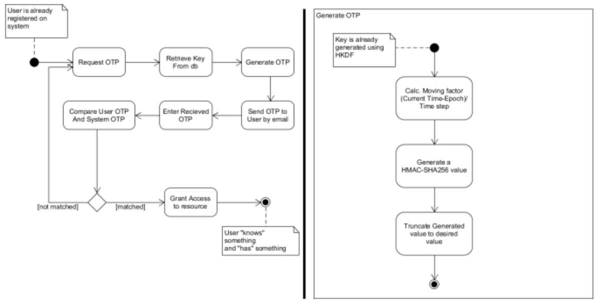

User is already registered on system
Request OTP
Retrieve Key From db
Generate OTP
Compare User OTP And System OTP
Enter Recieved OTP
Send OTP to User by email
[not matched]
[matched]
Grant Access to resource
User "knows" something and "has" something
Generate OTP
Key is already generated using HKDF
Calc. Moving factor (Current Time-Epoch)/ Time step
Generate a HMAC-SHA256 value
Truncate Generated value to desired value

Figure 1: OTP Authentication Activity Diagram

A. Generating an OTP

The Time-Based One-Time Password (TOTP) Algorithm is an extension of the HMAC-based One-Time Password (HOTP) Algorithm that uses time as the moving factor.

The TOTP algorithm has the following requirements:

(i) The Prover (e.g., token, soft token) and Verifier (authentication or validation server) must know or be able to derive the current Unix time (i.e., the number of seconds elapsed since midnight UTC of January 1, 1970) for OTP generation.

(ii) The Prover and Verifier MUST either share the same secret or the knowledge of a secret transformation to generate a shared secret.

(iii) The algorithm MUST use HOTP as a key building block.

(iv) TheProver and verifier MUST use the same time-step value X.

(v) There MUST be a unique secret (key) for each prover.

(vi) The keys SHOULD be randomly generated or derived using key derivation algorithms.

(vii) The keys MAY be stored in a tamper-resistant device and SHOULD be protected against unauthorized access and usage.

This work satisfies the above requirements in the following ways:

(i) The apache server used knows the current time

(ii) The prover and verifier know the shared secret which is generated at registration

(iii) The HMACSHA256 algorithm was used

(iv) A time step of 60 seconds was used

(v) A unique secret key is generated for each user at registration

(vi) Keys are randomly generated using the HMAC Key Derivation Function

(vii) Keys are stored in a secure database

An OTP is generated by running the HMAC-SHA-256 algorithm with the secret key and current timestamp.

HMAC-SHA-256 (K, T)

where K = Secret Key

T = (Current Unix time - Epoch)/Time step

The output of the HMAC-SHA-256 calculation is 160 bits (20 bytes), which was truncated to something that can be easily entered by a user. Ankit.et al (2015) state that utilizing a hash calculation, which creates a 20 byte yield, is more secure and requires less memory.

For example, given the output:

HS = 1f|86|98|69|0e|02|ca|16|61|85|50|ef|7f|19|da|8e|94|5b|55|5a

The offset is defined as the lower 4 bits of the last byte. The last byte of HS is hex 5a; its lower 4 bits are therefore hex**a** (10 in decimal).

Flatten together the 4 bytes starting at offset 10:

1f|86|98|69|0e|02|ca|16|61|85|**50**|**ef**|**7f**|**19**|da|8e|94|5b|55|5a

Dynamic Code = 0x50ef7f19

Then converting 0x50ef7f19 to a decimal number gives 1357872921.

Now if you want a 6-digit OTP, use the last 6 digits which is**872921**.

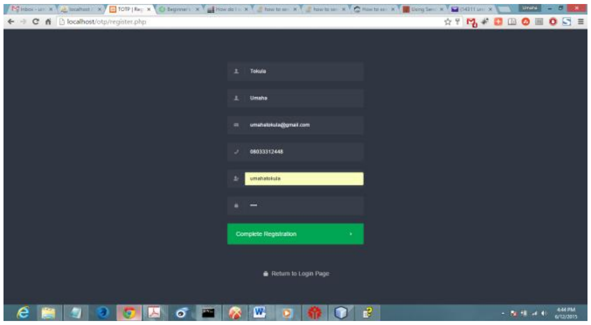
IV. RESULTS AND DISCUSSION

A. Registration

The landing page of the system is the login page but registration must be done before login will be successful. Figure 2 below shows the user registration page. It contains the First Name, Last Name, Email, Phone Number, Username and Password fields.

When the user submits the registration, a user object is created which calls the *keyGen()* function to generated a secret key using the password provided. The generated secret key is stored in the database.

After registration, the user is required to login using the username and password provided during registration. This is the one factor authentication


localhost/otp/register.php

Tokula

Umaha

umahatokula@gmail.com

08033312448

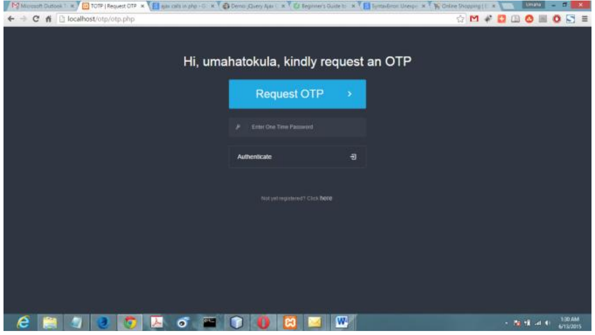
umahatokula

••••

Figure 2: User registration page

B. *One Time Password Authentication*

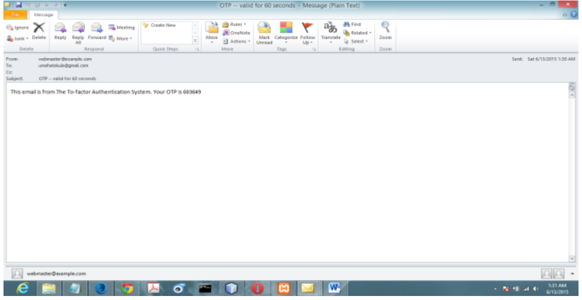
C. Upon successful login, the user is redirected to this page where he will request an OTP. The OTP is sent to the email provided during registration. The received OTP is then entered for authentication. An OTP is valid for 60 seconds. Figure 3 shows a userwho has requested for an OTP. Figure 4 shows the OTP in the users mail box. Figure 5 shows the user inputting the OTP. These three steps must happen within the span of 60 seconds.


localhost/otp/otp.php

Hi, umahatokula, kindly request an OTP

Enter One Time Password

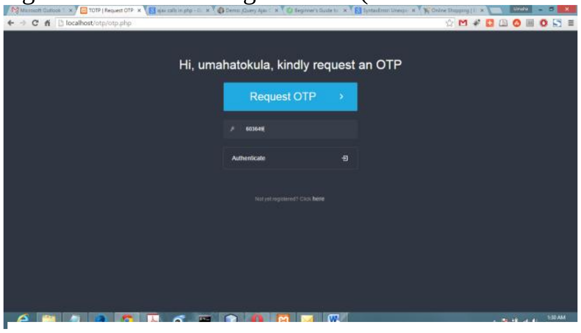
Figure 3 OTP authentication page


OTP -- valid for 60 seconds - Message (Plain Text)

From: webmaster@example.com
To: umahatokula@gmail.com
Cc:
Subject: OTP -- valid for 60 seconds
Sent: Sat 6/13/2015 1:30 AM

This email is from The To-factor Authentication System. Your OTP is 603649

Figure 4 Email showing the OTP (valid for 60 seconds)


localhost/otp/otp.php

Hi, umahatokula, kindly request an OTP

603649

Figure 5: Page showing entered OTP

V. CONCLUSION

Most online applications and processes use a one-factor authentication system which haveproven to be insufficient as hackers have had a field day breaking and stealing such passwordsover the years. Even in cases where passwords are required to be alphanumeric and have a mix of upper and lower case alphabets, it still isn't sufficient.

The time-based One Time Password authentication system will help a great deal in securing applications and transactions. Even if an attacker is somehow able to intercept on OTP, he has very limited time to use it and once an OTP has been used, it becomes useless. This we strongly believe will make information to be better secured if used.

REFERENCES

[1] Ahmad Alamgir Khan.(2013). Preventing Phishing Attacks using One Time Password nd User Machine dentification.*International. Journal of Computer Applications* (0975 – 8887) Volume 68– No.3

[2] AnkitAggarwal, DarshilDoshi, Vijay Gore and JigneshSisodia. (2015). Three Level Security Using Cued Click Points in Image Based Authentication.*International Journalof Innovative and Emerging Research in Engineering*-ISSN: 2394 – 3343 p-ISSN: 2394 – 5494

[3] Ayushi. A (2010) Symmetric Key Cryptographic Algorithm. *International Journal of Computer Applications* (0975 - 8887) Volume 1 – No. 15

[4] Hongfeng Zhu, Yu Xia and Hui Li. (2015) An Ancient and Secure Biometrics-based One-Time Identity-Password Authenticated Scheme for E-coupon System towardsMobile Internet.*Journal of Information Hiding and Multimedia Signal Processing* Volume 6, Number 3.

[5] Humaira Dar, WajdiFawzi Mohammed Al-KhateebAnd Mohamed HadiHabaebi. (2013). Secure Scheme For User Authentication And Authorization In Android Environment. *Int. Journal of Engineering Research and Applications*. Vol. 3, Issue 5, pp.1874-1882

[6] Lamport L. (1981) Password Authentication with Insecure Communication. Communications of the ACM, vol. 24, no. 11, pp. 770-772.

[7] MansoorEbrahim, Shujaat Khan, Umer Bin Khalid. (2013). Symmetric Algorithm Survey: A Comparative Analysis. *International Journal of Computer Applications*.Volume 61 No.20.

[8] Niharika Gupta and Rama Rani.(2015). Implementing High Grade Security in Cloud Application using Multifactor Authentication and Cryptography. International Journal of Web & Semantic Technology (IJWesT) Vol.6, No.2

[9] NiveditaBisht, Sapna Singh. (2015). A Comparative Study of Some Symmetric and Asymmetric Key Cryptography Algorithms. *International Journal of Innovative Research in Science, Engineering and Technology*.Vol. 4, Issue 3.

[10] Prashant Kumar Arya, DrMahendra Singh Aswal, DrVinod Kumar. (2012). Comparative Study of Asymmetric Key Cryptographic Algorithms. *International Journal of Computer Science & Communication Networks*,Vol 5(1),17-21

[11] RanjeetMasram, VivekShahare, Jibi Abraham, RajniMoona. (2014). Analysis and comparison of symmetric key cryptographic algorithms based on various file features. *International Journal of Network Security & Its Applications (IJNSA)*, Vol.6, No.4.

[12] Y. Huang, Z. Huang, H.R. Zhao and X.J. Lai.(2013).A new one-time password method. Proceeding of the Informational Conference on Electronic Engineering and Computer Science, pp 32-37.