SET with SMS OTP using Two Factor Authentication

D.Parameswari [a,*], L.Jose [b,1]

Abstract - **This paper describes a method of implementing two factor authentication using SMS OTP - One Time Password to Secure an E-Transaction (SET). The proposed method guarantees authenticated transactions in services, such as online banking, e-shopping or ATM machines. The proposed system involves generating and delivering a One Time Password (OTP) to a mobile phone in the form of SMS – Simple Messaging Service. The generated One Time Password is valid for only a short user defined period of time and it is generated and verified using Secured Cryptographic Algorithm. The proposed method has been implemented and tested successfully.**

Index Terms **– SET,OTP,SMS.**

I. INTRODUCTION

Security is a major concern today in all sectors such as banks, governmental applications, military organization, educational institutions, etc. Government organizations are setting standards, passing laws and forcing organizations and agencies to comply with these standards with non-compliance being met with wide-ranging consequences. There are several issues when it comes to security concerns in these numerous and varying industries with one common weak link being *passwords*.

The rapid growth in the number of online services leads to an increasing number of different digital identities each user needs to manage. But passwords are perhaps the most common type of credential used today [1]. To avoid the tedious task of remembering difficult passwords, users often behave less securely by using low entropy and weak passwords. Most systems today rely on *static passwords* to verify the user's identity.

However, such passwords come with major management security concerns. Users tend to use easy-to-guess passwords, use the same password in multiple accounts or store them on their machines, etc. Furthermore, hackers have the option of using many techniques to steal passwords such as shoulder surfing, snooping, sniffing, guessing, etc. Moreover passwords can be guessed, forgotten, written down and stolen, eavesdropped or deliberately being told to other people.

Several proper strategies for using passwords have been proposed [2]. Some of which are very difficult to use and others might not meet the company's security concerns. Some solutions have been developed to eliminate the need for users to create and manage passwords. A typical solution is based on giving the user a hardware token that generates one-time-passwords, i.e. passwords for single session or transaction usage.

Unfortunately, most of these solutions do not satisfy scalability and/or usability requirements, or they are simply insecure.

Moreover they also have disadvantages which include the cost of purchasing, issuing, and managing the tokens or cards. From the customer's point of view, using more than one two-factor authentication system requires carrying multiple tokens/cards which are likely to get lost or stolen. So we have a provision of OTP in Mobile, but there are major hurdles in that, we have to install OTP generation software in all clients mobile, the time in both mobile and server has to be always synchronized, if client purchase a new mobile, the mobile have to be registered and installed with the OTP generation software, updated software have to re-installed in all client mobile.

In this paper, we propose a securely generated and verified OTP which is sent to user mobile phone as a *SMS* with Transaction details. SMS is riveted because SMS is a ubiquitous communication channel, being available in all handsets and with a large customer-base, SMS messaging has the greatest potential to reach all consumers with a low total cost of ownership. Tokens, smart cards and other traditional authentication methods are more costly to implement, pricey to maintain and frequently resisted by consumers. Also tokens can be lost, and delivering OTP into mobile might be more secure and simpler, because consumers do not have to carry an extra portable device.

Generally e-transactions are vulnerable to man-in-the-middle (MITM) and man-in-the-browser (MITB) attacks, in which phishers hijack online sessions by tricking customers into providing OTP generated by tokens or smart cards. But proposed system can effectively fight against man-in-the-middle (MITM) and man-in-the-browser (MITB) attacks, because transaction details are sent along with the OTP. If a MITM/MITB modifies or add his transaction to user's original transactions, then user can find that from SMS and drop the current transaction.

Moreover one time password will contain numbers and random characters and they are changed at each login. The one time password will not be delivered through the computer network, so it will be harder for an intruder to intercept it. As they are changed constantly people cannot write them down. Furthermore, if someone gets your password it cannot be used the next time it is needed as it expires automatically after some time. All in all, a two-factor password is harder to guess and intercept.

Mobile phones have traditionally been regarded as a tool for making phone calls. But today, given the advances in hardware and software, mobile phones use have been expanded to send messages, check emails, store contacts, etc. Mobile connectivity options have also increased. After standard GSM connections, mobile phones now have infra-red, Bluetooth, 3G, and WLAN connectivity. Most of us, if not all of us, carry mobile phones for communication purpose.

In the next section we provide a general background about authentication factors and existing two factor authentication systems. Section III describes the proposed system, the OTP algorithm, SMS integration, the database. Section IV concludes the paper and provides future work.

II. BACKGROUND

Authentication is the process of verifying the correctness of a claimed identity. It is a way of ensuring that users are who they claim to be when they access systems. Authentication relies on at least one of three types of information: *something you know* (e.g., Password or Pin), *something you have* (e.g., Smartcards or Token),

Manuscript received, 12-Dec-2011.

D.Parameswari [a,*],
Senior Assistant Professor,
Department of Computer Applications,
Jerusalem College of Engineering, Chennai.
E-mail: vai_sn10@yahoo.co.in
L.Jose [b,1],
MCA student,
Department of Computer Applications,
Jerusalem College of Engineering, Chennai.

or *something you are* (e.g., a Finger prints or Iris scan, Biometrics) [4].

The traditional system only uses one level of authentication — the humble password. Two-factor authentication requires that two pieces of data be presented, each being from a different category. This dramatically reduces the risk of a system being compromised because the chance of both authentication factors being broken or lost at the same time is minimal.

In this system, we are going to implement two factor authentication. Two-factor authentication [5] (TFA or 2FA) means using two independent means of evidence to assert an entity's identity to another entity. Two factor authentications are referred to as *possession factor* and *knowledge factor*. Authentication Mechanism may require users to provide a password (knowledge factor) and a pseudorandom number, an OTP (possession factor). Two-factor authentication seeks to decrease the probability that the requestor is presenting false evidence of its identity. It is generally accepted that any independent two of these authentication methods (e.g. password + OTP token value) is two-factor authentication. Two-factor authentication (T-FA) or (2FA) is a system wherein two different factors are used in conjunction to authentication. Using two factors as opposed to one factor generally delivers a higher level of authentication assurance. Two-factor authentication typically is a signing-on or approving transaction process where a person proves his or her identity with two methods.

Passwords are known to be one of the easiest targets of hackers. Therefore, most organizations are looking for more secure methods to protect their customers and employees. Biometrics are known to be very secure and are used in special organizations, but they are not used much to secure online transactions or ATM machines given the expensive hardware that is needed to identify the subject and the maintenance costs, etc. Instead, banks and companies are using tokens as a mean of two factor authentication. A security token is a physical device that an authorized user of computer services is given to aid in authentication. It is also referred to as an authentication token or a cryptographic token. Tokens come in two formats: hardware and software. Hardware tokens are small devices which are small and can be conveniently carried. Some of these tokens store cryptographic keys or biometric data, while others display a PIN that changes with time. At any particular time when a user wishes to log-in, i.e. authenticate, he uses the PIN displayed on the token in addition to his normal account password. Software tokens are programs that run on computers and provide a PIN that change with time. Such programs implement a One Time Password (OTP) algorithm. OTP algorithms are critical to the security of systems employing them since unauthorized users should not be able to guess the next password in the sequence. The sequence should be random to the maximum possible extent, unpredictable, and irreversible. Factors that can be used in OTP generation include names, time, seed, etc. Several commercial two factor authentication systems exist today such as BestBuy’s BesToken, RSA’s SecurID, and Secure Computing’s Safeword [3].

BesToken applies two-factor authentication through a smart card chip integrated USB token. It has a great deal of functionality by being able to both generate and store users’ information such as passwords, certificates and keys. One application is to use it to log into laptops. In this case, the user has to enter a password while the USB token is plugged to the laptop at the time of the login. A hacker must compromise both the USB and the user account password to log into the laptop.

Secure ID from RSA uses a token (which could be hardware or software) whose internal clock is synchronized with the main server. Each token has a *unique seed* which is used to generate a pseudo-random number. This seed is loaded into the server upon purchase of the token and used to identify the user. An OTP is generated using the token every 60 seconds. The same process occurs at the server side. A user uses the OTP along with a PIN which only he knows to authenticate and is validated at the server side. If the OTP and PIN match, the user is authenticated [7]. In services such as ecommerce, a great deal of time and money is put into countering possible threats and it has been pointed out that both client and the server as well as the channel of communication between them are imperative.

Using tokens involves several steps including registration of users, token production and distribution, user and token authentication, and user and token revocation among others [6]. While tokens provide a much safer environment for users, it can be very costly for organizations. For example, a bank with a million customers will have to purchase, install, and maintain a million tokens. Furthermore, the bank has to provide continuous support for training customers on how to use the tokens. The banks have to also be ready to provide replacements if a token breaks or gets stolen. Replacing a token is a lot more expensive than replacing an ATM card or resetting a password.

From the customer’s perspective, having an account with more than one bank means the need to carry and maintain several tokens which constitute a big inconvenience and can lead to tokens being lost, stolen, or broken. In many cases, the customers are charged for each token. So we propose a mobile-based software token that will save the organizations the cost of purchasing and maintaining the hardware tokens. Hence, they will only worry about their mobile phones instead of worrying about several hardware tokens.

III. Design Implementation

In this paper, we propose a computer-based software token. This is supposed to replace existing hardware token devices. The System involves generation of Secured OTP using Cryptographic algorithm and delivering it to user’s mobile in the form of SMS with transaction details and validating the OTP using same Cryptographic algorithm. The proposed system is secured and consists of two parts: (1) the server software, (2) a GSM modem connected to the server.

A. OTP Algorithm

In order to secure the system, the generated OTP must be hard to guess, retrieve, or trace by hackers. Therefore, it is very important to develop a secure OTP generating algorithm. Several factors can be used by the OTP algorithm to generate a difficult-to-guess password. Users seem to be willing to use simple factors such as their mobile number and a PIN for services such as authorizing mobile micro payments [8]. So we propose a Secured Cryptographic algorithm, **Threshold Secret Sharing Scheme** to generate and verify the OTP.

In this *n* participants hold shares generated from secret *S*. A (k, n) threshold scheme is followed, where any information about *S* cannot be obtained from *k-1* or less shares. *S* can be recovered only from *K* and more shares. A fast (2, n) threshold is used here, where we could get *S* just by XOR operation. This was proven fast and secure [9].

In this scheme we generate a random number and divide secret into 4 shares by making XOR. Then we encrypt and deliver one share to user in the form of SMS. If he is a valid user, then he will provide approval by typing the OTP during transaction approval. Then that share is decrypted and compared with the other share which is stored in the database. Thus this Threshold Secret Sharing Scheme (TSSS) helps to generate a Secured and Random OTP.

B. Database Design

A database is needed on the server side to store the client's identification information such as the first name, last name, username, password and the mobile phone number for each user. And also user id, details of transaction, OTP sent and date and time of transaction for every transaction. The OTP field will store the hash of the 10 minute One Time Password. It will not store the OTP itself. Should the database be compromised the hashes cannot be reversed in order to get the OTP used to generate those hashes. Hence, the OTP algorithm will not be traced.

C. Server Design

A server is implemented to generate the OTP on the organization's side. The server consists of a database as described in Section III.B and is connected to a GSM modem for SMS messages exchange. The server application is multithreaded. The first thread is responsible for initializing the database and SMS modem. The second thread is responsible for generating and sending the OTP and transaction details in the form of SMS. A third thread is used to validate the OTP given by user to the OTP stored in database. The fourth thread would trigger the bank about the approval of transaction and payment to vendor. Fig.1 shows the workflow of OTP system.

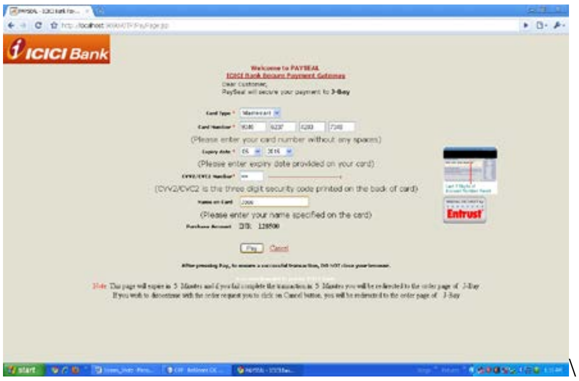
D. OTP System Workflow

The OTP system works as below. First user purchase items over web and agrees to pay for the same. Now e-shopping system will notify the OTP system. OTP system will look up for the mobile number of user from the database.

Now OTP system will generate a random number sequence and divide them into 4 shares. Every share will be encrypted and stored in database. One share will be sent to the user mobile as SMS with the transaction details.

Then SMS delivery details such as date and time, OTP sent and transaction details will be stored in the database. After user gets the OTP and transaction details as SMS, he approves the transaction by entering that in the web application. Now this OTP entered by user is sent to OTP system. Now that OTP share entered by user is decrypted and validated with the other shares to satisfy the threshold that is maintained.

If the validation is successful, the OTP system notifies bank to make payment to the vendor. Here if MITM/MITB tries to alter transaction beneficiary or adds his transaction to the user's original transaction, then user is notified about this modification made, so he can deny the payment, which will drop the transaction. Thus user and his transactions are safe from MITM/MITB. Fig.2 shows the payment page of the banking system.



Welcome to PAYSEAL
ICICI Bank Secure Payment Gateway
Dear Customer,
PaySeal will secure your payment to J-Bay

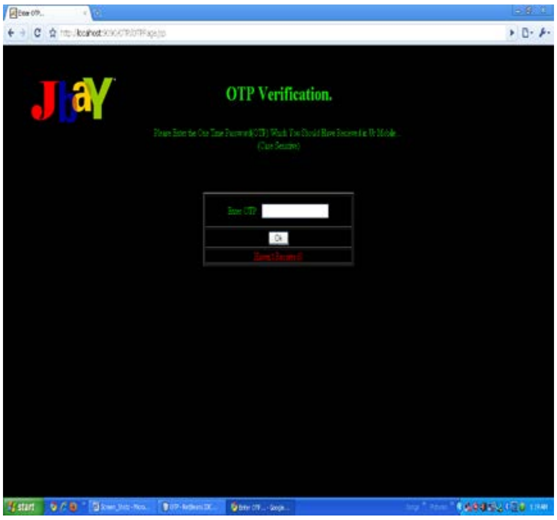
Card Type * Mastercard
Card Number * [illegible]
(Please enter your card number without any spaces)
Expiry date * [illegible]
(Please enter expiry date provided on your card)
CVV2/CVC2 Number*
(CVV2/CVC2 is the three digit security code printed on the back of card)
Name on Card [illegible]
(Please enter your name specified on the card)
Purchase Amount [illegible]

After pressing Pay, to ensure a successful transaction, DO NOT close your browser.

Note: This page will expire in 5 Minutes and if you fail complete the transaction in 5 Minutes you will be redirected to the order page of J-Bay
If you wish to discontinue with the order request you to click on Cancel button, you will be redirected to the order page of J-Bay

\

Figure1. Payment page of the bank



OTP Verification.

Please Enter the One Time Password(OTP) Which You Should Have Recieved in Ur Mobile...
(Case Sensitive)

Figure2. Verification page of the OTP system

IV. Conclusion

Today, single factor authentication, e.g. passwords, is no longer considered secure in the internet and banking world. Easy-to-guess passwords, such as names and age, are easily discovered by automated password-collecting programs. Two factor authentications have recently been introduced to meet the demand of organizations for providing stronger authentication options to its users. In most cases, a hardware token is given to each user for each account. The increasing number of carried tokens and the cost the manufacturing and maintaining them is becoming a burden on both the client and organization. Since many clients carry a mobile phone today at all times, an alternative is to install all the software tokens on the mobile phone. It is again a tedious process to deploy software token system in all users mobile and implementing changes to software require new installation which is even more tedious. So we deliver it as SMS. This will help reduce the manufacturing costs, installation cost and the number of devices carried by the client.

This paper focuses on the implementation of two-factor authentication methods using mobile phones. It provides the reader with an overview of the various parts of the system and the capabilities of the system. The proposed system have been successfully implemented and tested, and shown to be robust and secure. The system has several factors that make it difficult to hack. Future development includes using an image as OTP [10].

References

[1] The mobile phone as multi otp device using trusted computing http://eprints.qut.edu.au/37711/

[2] A. Jøsang and G. Sanderud, "Security in Mobile Communications: Challenges and Opportunities," in Proc. of the Australasian information security workshop conference on ACSW frontiers

[3] Aladdin Secure SafeWord 2008. Available at http: // www . secure computing.com/index.cfm?skey=1713

[4] Authentication http://en.wikipedia.org/wiki/Authentication

[5] B. Schneier, "Two-Factor Authentication: Too Little, Too Late," in Inside Risks 178, Communications of the ACM, 48(4), April 2005.

[6] D. de Borde, "Two-Factor Authentication," Siemens Enterprise Communications UK- Security Solutions, 2008. Available at http://www.insight.co.uk/files/whitepapers/ Two factor %20 authentication %20(White%20paper).pdf.

[7] J. Brainard, A. Juels, R. L. Rivest, M. Szydlo and M. Yung, "Fourth-Factor Authentication: Somebody You Know," ACM CCS, 168-78. 2006.

[8] NBD Online Token. Available at http://www.nbd.com/NBD/NBD_CDA/CDA_Web_pages/Internet_Banking/nbdonline_topbanner

[9] Http://isc08.twisc.org /slides/S10P2_A_Ne(k,n)-Threshold_Secret_Sharing_Scheme_and_Its_Extension.pdf

[10] Problem with 2FA Solution https://www.infosecisland.com/blogview/13734-The-Problem-with-Two-Factor-Authentication-Solutions.html.

BIOGRAPHY

Ms.D. Parameswari received MCA degree from Bharathiar University,M.Sc(C.S) degree from university of Bharathidasan in 1996 and M.Phil(C.S) from Manonmaniam University in 2003.Currently she is working as a Senior Assistant Professor in Jerusalem college of Engineering,Chennai,Tamilnadu. She has 14 years of teaching experience on graduate level.Her area of interest includes Network Security and Data Minig.