A Multilevel Security Authentication for Internet Banking Based on Cryptography

Mahalaxmi C[1], Geethamani G.S[2]
[1]Dept of Msc (IT)
[2]Assistant Professor, Dept of Msc (IT)
[1, 2] Hindusthan College of Arts and Science, Coimbatore, India.

Abstract- *There are a continuously growing number of customers who use Internet banking because of its convenience. But the security and privacy of Information may be one of the biggest concerns to the Online Banking users. The problem with Online banking applications is that they send data directly to customer in plain text form compromising with security. The solutions to the security issues require the use of software-based solutions that involve the use of encryption algorithms. For this we propose a challenge/response -based short-time password authentication methods using Symmetric cryptography in combination with Software Security model. In this approach bank hides customer transaction data is secure SMS using IDEA symmetric cryptographic algorithm and send it to customer application supported handset. Customer application decrypts data in secure manner the encryption and decryption are characterized by a secret key that the legal parties have to posses. So, in face of the current security issues and the growing number of attacks and consequent frauds, new internet banking systems should be designed as to provide better authentication and identification methods. And these methods can be implemented to the Mobile banking to address the Security concern. Due to advancements and improvements in internet and communication systems, more people are relying on internet to store their confidential information. Earlier the idea of Static passwords was being used but most of the users try to use easily guessable, weak passwords or keywords from their personal information, which makes it easy for the intruders to guess their passwords in few combinations using Brute Force attack. Thus idea of using Multi-Factor Authentication has been introduced in the world of internet to harden the security of network and make it difficult for the attackers to crack systems. In this mechanism, users are required to provide some extra information along with their login Id and password. Most popular is using One-Time Passwords that are generated randomly and valid only for single login and even for short duration of time (usually 30 to 60 seconds). One-Time Passwords can be generated either online or offline via various mechanisms. In this paper, review of various Multi-step Authentication schemes has been performed to compare various authentication mechanisms*

Keywords- Multi-factor Authentication, One-Time Passwords (OTP) ,Static Passwords, Short Message Service (SMS), Image based Authentication.

I. INTRODUCTION

The security models for online banking systems currently in use are strongly based on Internet banking user identification and authentication methods [2], which are also the components where most Internet banking systems' vulnerabilities are found. The existing SMS system does not have any built -in procedure to authenticate [1] the text and offer security for the text transmitted as data, because most of the applications for mobile devices are designed and developed without taking security into consideration.

1(a) Security

Security of the transactions is the primary concern of the Internet -based industries. The lack of

Security may result in serious damages the security issue [3] will be further discussed in the next section along with the possible attacks due to the insufficient protections. The examples of potential hazards of the electronic banking system are tra nsferring funds, and minting electric currency, etc.

1(b) Authentication

Encryption may help make the transactions more secure. The Internet of today has become an integral part of our

Everyday life and the proportion of users expecting to be able to manage their bank accounts anywhere anytime is constantly growing [4]. As such, Internet banking has come to age as a crucial component of any financial institution’s multi-channel strategy. Based on the assumption that only an authentic user is able to d o so, successful authentication eventually enables an authorized user to access his private information.[5]

1(c) Short-Time Password Solution

Considering today's pervasiveness of malicious software and phishing attacks, any Internet banking solution must be resistant against stealing attacks . For this we propose a challenge/response-based short-time password authentication method using symmetric cryptography in combination with a Software Security Model [6, 7]. The short-time password solution is in use at in software-based security systems, the coding and decoding of information is done using specialized security software.

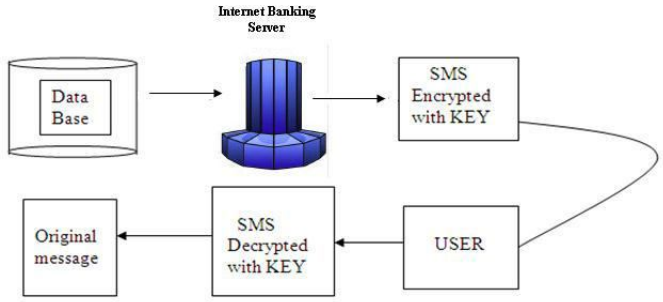

Fig 1: Architecture of Secure Transaction

The Fig (1) shows the Architecture of the Secure Transaction, where the Bank Server sends the Encrypted message to the User and the User Decrypt's the message [8, 9], the encryption and decryption are characterized by a secret key that all legal parties have to posses. The application is develop using programming language Java and t he J2ME environment. [10, 11]

II. SECURE SMS MESSAGES USING CRYPT OGRAPHY

Most of the attacks directed at online banking systems target the user (the weakest link in the Chain), focusing on obtaining authentication and identification information through th e use of Social engineering and compromising the user's Internet banking access device in order to install malware which automatically performs banking transactions [13,14], apart from obtaining authentication data. This fact indicates that secure internet banking systems should provide security mechanisms as user independent as possible, mitigating the risk of user related information's leaks and security issues affecting the system and leading to fraud [15].

2(a) SMS Security Algorithm: The International Data Encryption Algorithm (IDEA)

SMS messages are sometimes used for the interchange of confidential data such as social security number, bank account number, password etc [23]. A typing error in selecting a number when sending such a message can have se vere consequences if the message is readable to any unauthorized receiver. In this application for sending encrypted SMS messages using cryptographic methods based on the IDEA Algorithm. The encryption algorithm is characterized by a secret key. The application is develop using programming language Java and the J2ME environment [12].

International Data Encryption Algorithm (IDEA) is very secure; IDEA operates on 64 bit blocks using a 128-bit key, and consists of a series of eight identical transformations (a *round*) and an output transformation (the *half-round*). The processes for encryption and decryption are similar [16, 17]. IDEA derives much of its security by interleaving operations from different groups — modular addition and multiplication, and bitwise eXclusive OR (XOR)

2(b) Multi-factor authentication

Multi-factor Authentication is a method of computer access control which a user can pass successfully presenting various authentication stages. In this, instead of asking just single piece of information like passwords, users are asked to give some additional information which makes it more difficult for any intruder to fake the identity of the actual user. This additional information can include various factors like finger prints, biometric authentication, security tokens etc. It has emerged an alternative way to improve the security by requiring the user to provide with more than one authentication factor rather than only a single password. Authentication factors are of these kinds:

- *Knowledge* – something that the user knows, e.g., a username and a password;
- *Possession* – something the user has, e.g., a hardware token(as a security token);
- *Inherence* – something verifies the user is, e.g., fingerprints [10].

Multi factor authentication can be performed in various ways, most common of them is using login credential with some additional information but a different technique also include authentication in which usage pattern of input data is used in determining the authenticity of user like the time taken by user to input his details, or the pressure exerted by the user's finger.

III. IMPLEMENT AT ION

In addition, communicating over wireless networks with a mobile device brings its own set of inherent risks. With knowledge a nd equipment, it is possible to intercept data most anywhere between your device and the end point of your intended communication [18, 19]. A J2ME system which is installed in all the mobiles of the users those who are registered with this system. The user who received the encrypted message has to enter into this application and provide the corresponding key to decrypt and see the message. The encryption application-related data is in Java 2 Micro Edition (J2ME) application (MIDlet).

3(a) Bouncy Castle

Bouncy Castle is an open source encryption library; there is a lightweight version suitable for use with J2ME. Bouncy Castle is an open source Java API for encrypting and decrypting data. Bouncy Castle is a Java Cryptography Extension (JCE) provider. JCE is an optional package that provides support for ciphers, keys, and message authentication within the Java 2 Platform, Standard Edition (J2SE). Essentially, a provider offers one or more algorithms for the encrypting and decrypting of data. To date, Bou ncy Castle supports over 20 engines. **ProGuard** is an open source Java class files obfuscator. This is used to obfuscate the contents, which requires less memory to store.

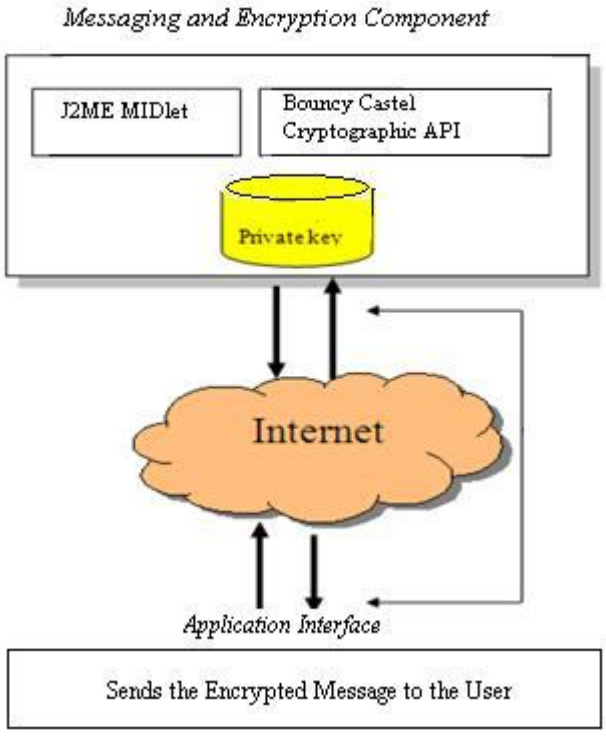

Fig 2: Shows the Messaging and Encryption Component

To implement the following process, the Sun Microsystems Wireless To olkit formerly known as Java 2 Platform, Micro Edition (J2ME) Wireless Toolkit) is used. It is a state-of-the-art toolbox for developing wireless applications that are based on J2ME's Connected Limited Device Configuration (CLDC) and Mobile Information De vice Profile (MIDP), and designed to run on cell phones, mainstream personal digital assistants, and other small mobile devices. The toolkit includes the emulation environmen ts, performance optimization and tuning features . Wireless Toolkit is an integrated development environment (IDE) creates J2ME MIDlet [20]. The WTK contains an IDE, as well as the libraries required for creating MIDlets.

A MIDlet is created which accepts the text to be encrypted along with the key (Secret key) by using which the Encr yption process is done by using IDEA algorithm. The Symmetric ciphers use the same key to encrypt and decrypt data, which is known as *secret key*. That is, the value of the key is kept secret between the two parties -- those who encrypt the data and those who decrypt it.

3(b) RSA SecurID

The RSA SecurID authentication System consists of a token that can be hardware (e.g.- a USB dongle) or a software (a soft token) which is given to the computer user and it is used to generate one time unique passwords that lasts for a maximum of 60 seconds time span. Generation of this one time password is done using encoded-random key that is known as seed. This seed is unique for each token and is loaded into their corresponding to RSA SecurID server. Tokens are also available On-Demand, in which token codes or unique passwords can be sent to the user via email or text SMS, which eliminates the need of a provision of token to the user. In this authentication scheme, seed is the secret key used to generate unique passwords. It also allows token to be used as Smart Card-like device to store certificates securely.

(a)

(b)

Fig 3: RSA SecurID hardware token: (a) older style, Model SD600, (b) New style Model SID800 with smartcard functionality

While RSA SecurID authentication mechanism provides stronger layer of security to a network as well as it protects the network from replay attack. But it is more vulnerable to man-in-the-middle attacks when used alone. If the intruder manages to block the legitimate user from authenticating him to the server until the next code will be valid, the attacker will be able to login to the server. Also, the difficulty may occur in this system if the authentication server's clock becomes out of sync with the clock built into the authentication tokens. However, the security of this system can be improved using mechanisms for encryption/ authentication such as SSL.

Hard Tokens are on the other hand can be physically stolen (like they can be stolen by social engineering attacks) from the authenticated end users. Also the user will not report immediately after the theft of the security token. The user will at least wait for one day before reporting the device as missing. This will give intruder a plenty of time to breach the protected system. However this could only occur if the unique username and password of account is known.

3(c) Google Authenticator

Google introduces 2-step verification or authentication scheme in September 2010 for Google Applications users. After enabling this service user have to provide an extra verification code after logging into their Google accounts. This verification code could be received by a Short Message Service (SMS) text message or voice over text message, or even through a token or code generating application developed by Google. Google's 2-step verification requires something you have (like smart phone with Google authenticator installed to generate verification code) and something you know (that is the password of your Google account) that is required to access into your account . The verification code could be retrieved via a token generator on a Smartphone. These token based verification codes are generated using a time-based algorithm.

And application that performs this verification code generating is called as Google Authenticator. Google authenticator is a software-based OTP generation scheme based on Time-based One Time Passwords (TOTP). It implements TOTP; security token from RFC 6238 in mobile apps made by Google or may be referred to as 'Two-Step Verification'. Google Authenticator uses an offline scheme of TOTP, where it's user's device which generates one-time passwords for the user rather than the server. Authenticator provides a six to eight digit one time unique password which user must provide in contrast with username and password to get access or login to Google services or other sites. It is an open source project that is available for android, iOS and BlackBerry devices. The application generates one-time time-based code using open-standards, including HMAC-based One-Time password (HOTP) algorithms and Time-based One-Time password (TOTP) algorithms. The generated Token or code is six digits in length and is valid for a 30 second timeframe.

Before the application could generate the unique tokens, it has to be linked with user's Google account either using Quick Response (QR) code which is created by Google and has to be scanned by the user Smartphone, or by using a secret-key provided by Google [4].Once the account is linked to the device, the app can generate a token for 30 seconds time span, after which a new token will be generated.

When a user has enabled his Google applications with Google's 2-step verification, his login process will be protected through an extra layer of security. Firstly, the user will as usual has to enter his username or login id and password and then in second step, he will enter the 6-digit verification code generated by the application Google Authenticator installed on the users Smartphone [4]. But, before the application could generate verification codes, it has to be linked to users Google account. This can be done via two methods. Firstly, link can be made using Quick Response (QR) code which is generated by Google in the browser and has to be scanned by the Smartphone device. Another method is by using a secret key provided by Google. Once the account and Google Authenticator are linked to each other, the Application would generate security token or verification code that are valid not more than 30 seconds time span i.e. it will automatically expires after 30 seconds and a new code will be generated.

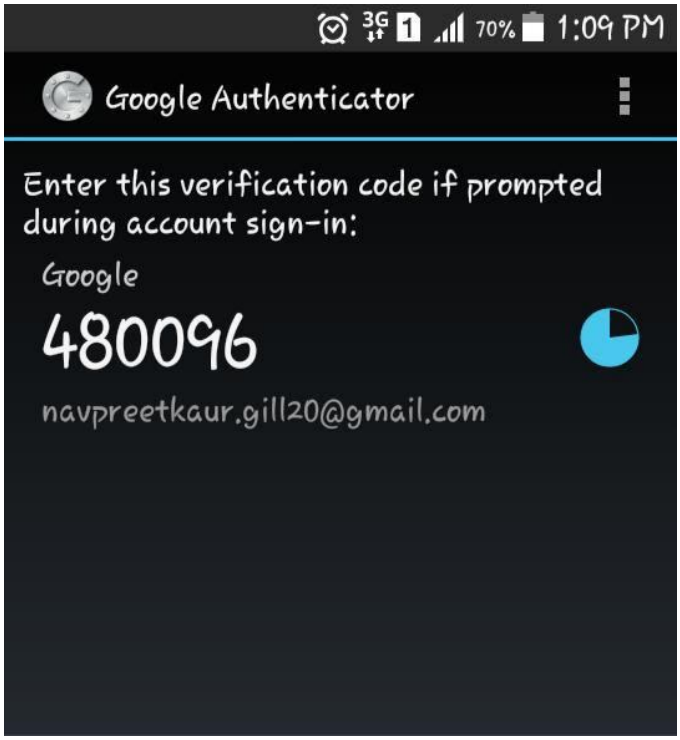

Fig 4: Verification Code generated by Google Authenticator

The major advantage of this system is that it generates tokens offline i.e. it can also generate verification codes even if there is no network connectivity. Also it ensures that only the rightful owner is given access to the account. The Time-based One Time Password (TOTP) generated verification codes based upon a synchronize time between the Google services and the users mobile device provides a robust login system that is not prone to attacks. But the major drawbacks of this system comes down to two things: seed transmission and seed storage. When it comes to transfer the seed to the mobile phone, Google relies on QR codes in which the seed travels in plaintext during transmission. This is more vulnerable to be attacked by any intruder. Also secret seed and login credentials of user are stored on Android device in plaintext so can be accessible to anyone easily and can be used to enrol the same seed on multiple devices.

3(d) Time-based OTP Authentication via Secure Tunnel

Time-based OTP Authentication via Secure Tunnel (TOAST) is a Smartphone app that obtains its secret seed from the server through a secure tunnel- TLS/SSL, stores this seed value on the Smartphone using a password protected keystroke, and then uses the seed to generate all time one time unique codes. Mainly three entities are involved in TOAST: a client who wants to login to the service, a server that listens for authentication requests, and a Smartphone with this app installed that possess by the client[2].Firstly the client or user has to register on the website of the service and then tends to download and install the Smartphone app. Then a secret seed will be generated by the server and shared with the client through secure TLS tunnel. This seed is stored on the mobile and will generate unique one time code offline every time the user wants to login. Whenever a user wants to access the service, the client may begin logging onto the account using his/her username, password and six-digit code generated by the phone. It uses the existing cryptographic standards and web protocols to increase the time and effort needed to crack a given system. It makes use of Blowfish algorithm for encryption or decryption and SHA-1 for generating one-time codes from a number of random keys.

Main advantages of TOAST authentication system are that it works offline for OTP generation i.e. can even work without network coverage. Secondly, the seed is not exposed in the plaintext form during enrolment which means the secret seed is generated at the server end and is transmitted over the network through TLS/SSL secure tunnel [2]. Lastly, the seed is stored inside the phone once transmitted and it is too stored in encrypted form and is password-protected using UBER Keystroke. Once the TOAST starts generating OTPs, it is self-reliant and will continue to operate reliably independent even of network conditions. But it also has some drawbacks or vulnerabilities that include no secure initial authentication setup during the transfer of secret seed.

3(e) Generation of secure One Time Password based on Image Authentication

The Image-based Authentication (IBA) is based on Recognition Technique. It is almost similar to text one time passwords as in this also the user is provided a shared secret as an evidence of his/her identity. However, text-based OTPs use alphanumeric characters to represent the secret and IBA uses visual information. When the user registers for the first time on the website, they are required to select a set of images that are easy to remember such as natural scenery, automobiles etc [8]. Every time a user login into the website or service, they are provided a grid of images randomly generated. Then, the user can identify the images previously selected by them. The user is authenticated by correctly identifying the password images. The category of images is stored by the authentication system on Image Identification Set (IIS). When a user login, the IIS for that user is only retrieved and is being used to authenticate that particular user. The human is more adept in retrieving or recalling a previously seen image rather than a previously seen text. In a study conducted at University of California at Berkeley, Image-based authentication (IBA) systems have been found as more user-friendly than usually used text-password systems.

Main advantage of IBA is that it is more secure and requires less memory. Image-based authentication also prevents from social engineering attacks, as it is easier to verbally describe the text password to the attacker but rather in case of image passwords nobody can reveal practically describe the passwords. Although graphical passwords may be shared via taking photos, taking screen shots or even through drawing but it obviously require more time than text passwords. Also, idea of using images as one time passwords makes it difficult for the attacker to intrude using Brute Force attack. But this also facilitates to data manipulation and interpretation to a greater extent than the alphanumeric characters does. This complexity, however, makes IBA harder to implement and deploy, requiring environments with increased computational power and graphical capabilities. This prevents it to be used by most of the services of websites because of complexity.

Hotspots: The major drawback in case of security in Image-based authentication is Hotspots. Hotspots are the specific areas in an image that have a higher probability of being selected by most of the users as a part of their passwords. If any attacker can accurately predict the hotspots in that image, a dictionary of images can be built basis on these hotspots. Thus, hotspots are meant to be problematic in Image-based authentication.[8]

IV. DISCUSSIONS

We now discuss some findings that are explored by the study and highlighted items for further references.

Adoption: Multi-step or Multi-factor authentication technologies are adopted at different rates, depending upon their context, complexity and motivation. Mostly, in the work environments, the verification codes generated by security tokens are popular way of adopting multifactor authentication. In personal context or financial transactions, one time passwords received via SMS or email is generally popular
The adoptability rate of Smartphone applications is little as a reason of complexity in their implementation.

Usability: Today, in the world of internet everybody is trying to harden the security of their data and every web service is trying in order to provide extra layer of security to their users so that it becomes difficult for the attacker to breach in to the account of the user .

Firstly, the most popular multi-factor authentication is using SMS-based OTP authentication. It is used in financial areas mostly like in banking transaction, internet banking, Google two-step authentication via SMS, Mater or Visa Debit-Credit card transactions, online payments on E-commerce websites etc.

Google Authenticator is a Smartphone based application that generates unique codes for accounts. It is used with only Google associated accounts like Gmail accounts. Image-based authentication scheme is also now becoming popular in various web service providers like in HDFC, ICICI bank use images as one-time unique codes in their online transactions like internet banking.

Multi-factor authentication's most important use is in financial services like in online banking or internet banking etc for providing security to the users so that only the authenticate or the legitimate user can process their financial transaction. As security risks are of great concern for the servers providing these services as well as for the users . In this we rely on the public key infrastructure for authentication and key generation . Another way of providing authentication in financial transactions like mobile banking is through analyzing user's usage pattern to input data into mobile and pressure of finger determine whether the user using the mobile device is legitimate or not .

Future scope: Various OTP generating mechanism are provided with more security day by day. New ideas may be introduced to remove any remaining points of vulnerability still remaining in systems in use today. There is a need to further harden existing authentication schemes such that they are easier to use but more complex to crack.

V. CONCLUSION AND FUTURE WORK

Internet banking is offering its customers with a wide range of services: Customers are able to interact with their banking accounts as well as make financial transactions from virtually anywhere without time restrictions. In order for electronic banking to continue to grow, the security and the privacy aspects need to be improved. With the security and privacy issues resolved, the future of electronic banking can be very prosperous. The future of electronic banking will be a system where users are able to interact with their banks ―worry-free‖ and banks are operated under one common standard. The security models for online banking systems currently in use are strongly based on Internet banking user identification and authentication methods, which are also the components where most Internet banking systems' vulnerabilities are found.

This paper describes current online banking problems and discusses the need for security testing for online banking. The system allows user to carry out all banking transaction

securely from anywhere, anytime. We have implemented system using symmetric key IDEA algorithm.

In future better power consumption algorithm like blowfish can be tried out. Steganogrpahy can also be applied for secure Internet banking and mobile banking transactions. We can use concept of STK, SIM application toolkit where bank can stored the applic ation and encryption keys on SIM.

Reviewing the pros and cons of various available login authentication schemes, firstly we reported on already available multi-step authentication mechanisms, how they work, how they are used, where and why. A few popular multistep authentication schemes include: one time pass code or passwords received via SMS, one time codes generated by security token i.e. RSA SecurID, Smartphone applications for generating verification code like Google authenticator and TOAST, using images as verification passwords i.e. Image-based authentication. Almost every kind of authentication system discussed above is widely used today to provide security to the users. One Time Passwords are an efficient technique to generate passwords randomly each time for user. OTP prevent users from replay or eavesdropping attacks. These passwords are valid only for given timeframe thus there is no threat that they can be reused by an intruder to login to user account as they are invalid after one time use. One Time Passwords can be generated either online or offline but offline generation is better as it can also be generated even if there is no network connectivity and it also prevents from the man in the middle attack. Thus it will be better for the services or websites to use offline method of generating one time unique codes like Google Authenticator or TOAST as they provide more confidentiality and authentication to the user on internet.

REFERENCES

[1] Singh, S., The Code Book: The Secret History of Codes and Code-breaking. Fourth Estate, 1999.

[2] Uymatiao, Mariano Luis T., and William Emmanuel S. Yu. "Time-based OTP Authentication via Secure Tunnel(TOAST): A mobile TOTP scheme using TLS seed exchange and encrypted offline keystroke." 4th *IEEE International Conference on Information Science and Technology(ICIST)*, 2014,Pp. 225-229,IEEE,2014.

[3] Muliner, C., Borgaonkar, R., Stewin, P.,Seifert, J., "SMS-based One-Time Passwords: Attacks and Defense", volume 7967,Pp. 150-159 Springer-Verlag Berlin Heidelberg 2013.

[4] Appelman, M., Scheelen, Y., "Analysis of Google's 2-step Authentication",University of Amsterdam, May 2012, www.scribd.com/doc/95267199/Analysis-of-Google-s-2-StepVerification#scribd

[5] Subashini, K., and Sumithra, G., "Secure multimodal mobile authentication using one time password." 2nd *International Conference on Current Trends in Engineering and Technology (ICCTET)*, 2014, pp. 151-155. IEEE, 2014.

[6] Takasuke Tsuji, Akihiro Shimizu, "A One-Time Password Authentication Method", January 2003, www.kochi-tech.ac.jp/library/ron/2002/g5/M/1055124.pdf

[7] Wikipedia-RSA SecurID, http://en.wikipedia.org/ wiki/ RSA_SecurID, 2015.

[8] Parmar,H., Nainan, N.,Thaseen, S.,"Generation of Secure One time passwords based on Image Authentication System", Pp. 195-206, 2012.© CS & IT-CSCP 2012.

[9] Kalaikavitha, E.,Gnanaselvi, J., "Secure Login using Encrypted One Time Password(OTP) and Mobile based LoginMethodology",InternationalJournalof

[10] Engineering and Science, Vol. 2, Pp. 14-17, Issue 10(2013).Emiliano De Cristofaro, Honglu Du, Julien Freudiger,Greg Norcie, "A Comparative Usability Study of Two-Factor Authentication", Cornell University Library, 31 January 2014.

[11] Munjal N., Moona R., "Secure and Cost effective Transaction Model for Financial Services", *International Conference on Ultra Modern Telecommunications and Workshops*, 2009, Pp. 1-6, IEEE, ICUMT'09.

[12] Mohammed M.M., Elsadig M., "A multi-layer of multi factors authentication model for online banking services", *International Conference on Computer, Electrical and Electronics Engineering (ICCEEE)*, Pp 220-224, August 2013.

[13] Hojin Seo, Huy Kang Kim, "User Input Pattern-based Authentication Method to Prevent Mobile e-FinancialIncidents", Ninth IEEE International Symposium onParallel and Distributed Processing with Applications Workshops (ISPAW), Pp 382-387, May 2011.

[14] Sandeep Singh Ghotra, Baldev Kumar Mandhan, Sam Shang Chun Wei, Yi Song, Chris Steketee, ‖ *Secure Display and Secure Transactions Using a Handset*‖,Sixth International Conference on the Management of Mobile Business.

[15] Dilla Salama Abdul Minaam. Hatem M. Abdul Kadir, Mohily Mohamed Hadhoud,*" Evaluating the effects of Symmetric Cryptographic algorithms on PowerConsumption for different data types*‖, International Journal of Network Security, Volume 11, September 2010.

[16] Managing the Risk of Mobile Banking T echnologies, Bankable Frontier Associates.

[17] Richard E. Smith. Authentication: From Passwords to Public Keys. Addison Wesley, 2001.

Encryption Issues.
Http://www.muc.edu:80/cwis/person/student/lockett/encryption.html

[18] Internet Security.
Http://cfn.cs.dal.ca/Education/CGA/netsec.html

[19] MICROSOFT, An in-depth perspective on software vulnerabilities and exploits, malicious code threats, and potentially unwanted software, focusing on the first half of 2008 [Report]. Security Intelligence Report, January through June 2008.

[20] M. JOHNSON, A new approach to Internet banking. University Cambridge. (PhD) 2008, p. 113.http://www.forum.nokia.com/main/1,35452,1_0_75,00.html.