Internet Banking Two Factor Authentication Using Smartphone

P.Y.Pawar, Sagar Acharya, Apoorva Polawar, Priyashree Baldawa, Sourabh Junghare

Abstract— In our project security and authentication plays a major role. It can be mainly used in online banking. The mobile phone acts as a security token for authentication. The user login's the online banking account by entering the username and password. For providing more security separate token number is used for performing the banking operation like money withdrawal, checking the balance etc. This token number is generated using the SHA algorithm and XOR operation. The user mobile number, IMEI number, pin number and IMSI number were included to generate the token number. The token number is a six digit random number that were obtained from the included number. The token number is sent to the user mobile. This token number is given for accessing or performing the banking operations. The token number is generated for every interval of time. For more than three times if the user gives any invalid pin number the account is blocked.

Index Terms— Authentication, IMEI, IMSI, OTP, SHA-1, SMS, token

1 INTRODUCTION

The mobile phones have a huge potential of interacting with services anywhere, anytime. The application described in this paper uses a mobile phone as a one time password (OTP) generator. The generated password will then be used in the authentication process of a client to an Internet Banking demo application where the client can also sign money orders using a similar challenge-response generator from their mobile phone. We will describe how the application resists on different attacks, in particular phishing attacks. The Internet Banking generally accepted definition, as it allows clients to perform transactions and payments over the internet through a bank's secure website, is used in this research.

Security concerns are rising today in all domains, such as banks, government, medical assistance, military organization, etc. Governmental organizations are passing laws that are forcing other organizations and agencies to comply with standards. There are several issues when it comes to security concerns in these industries and one common issue is represented by passwords.

Today, most systems are using static passwords for verifying their client's identity, passwords that come with major security concerns. Users tend to use easy to guess passwords; they use the same password for multiple accounts, write the passwords and store them on their machines, etc. Furthermore, hackers can use many techniques to steal passwords, such as shoulder surfing, snooping, sniffing, etc.

Many solutions have been proposed in order to fix this concern. Some of them are hard to implement, others don't meet the security concerns of the companies while others are difficult to be used. Two-factor authentication using hardware devices, such as tokens or ATM cards has also been proposed to solve these problems and proved to be successful and difficult to be hacked. However, this solution has also disadvantages such as the cost of manufacturing and managing tokens or cards. Also, from the consumer's point of view, using more than one token or card has an increased chance of them being lost or stolen. Mobile phones have traditionally been considered tools for making phone calls. But today, given the progress that has been done in the hardware and software, mobile phones use has been expanded to internet, email, games, music, photos and thousands of other rich applications added by third-party developers.

Installing third-party applications allows mobile phones to provide expanded new services other than communication. The use of mobile phone as a software token will make it easier for the customer to deal with multiple two-factor authentication systems and will also reduce the cost of manufacturing, distributing and maintaining millions of hardware tokens.

1.1 Basic Concept

Server Module: Server mainly stores the information of each individual user. It will generate OTP on the request and match

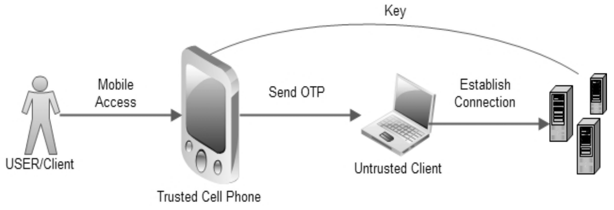

Figure 1: Basic Concept and modules

- P.Y.Pawar is currently assistant professor at Sinhgad Academy of Engineering, Pune, India. E-mail: poonam.y.pawar@gmail.com
- Sagar Acharya is currently pursuing bachelor degree program in Information Technology in University of Pune, India. E-mail: sagar_acharya@live.in
- Apoorva Polawar is currently pursuing bachelor degree program in Information Technology in University of Pune, India. E-mail: apoorva_polawar2006@yahoo.co.in
- Priyashree Baldawa is currently pursuing bachelor degree program in Information Technology in University of Pune, India. E-mail: priyabaldawa@gmail.com
- Sourabh Junghare is currently pursuing bachelor degree program in Information Technology in University of Pune, India. E-mail: sourabh_junghare@yahoo.co.in

that OTP with user’s OTP. Server accepts OTP via GSM modem.

Client Module: Client can able to perform transactions if and only if client’s OTP perfectly matched with server side OTP.

Smartphone App: In smart phone app an user friendly GUI is provided in which user have to put some data which known to him only to generate OTP

2 PROBLEM DESCRIPTION

In this paper we will introduce a system of two-factor authentication using smart phones, which consists of a server, client and a smart phone application that allows running third-party applications.

By definition, authentication is the use of one or more mechanisms in order to prove that you are who you claim to be. Once your identity is validated, access is granted. Three universally recognized authentication factors exists today: what you know (passwords), what you have (tokens, cards) and what you are (biometrics). Recent work has been done in trying alternative factors, for example somebody you know, a factor that can be applied in social networking.

Two-factor authentication is a mechanism that implements two of the above mentioned factors and is considered stronger and more secure than the traditionally implemented one factor authentication system. For example, withdrawing money from an ATM machine uses two factor authentication: the ATM card (what you have) and the personal identification number (what you know). Passwords are known to be one of the easiest targets of hackers. Therefore, most companies are searching more ways to protect their customers and employees. Biometrics are known to be very secure, but are used only in special organizations (such as military organizations) given the expensive hardware needed and their high maintenance costs. As an alternative, banks and companies are using tokens as a way of two-factor authentication.

A token is a physical device that generates passwords needed in an authentication process. Tokens can either be software or hardware. Hardware tokens are small devices that can be easily carried. Some of these tokens store cryptographic keys or biometric data. Anytime a user wants to authenticate in a service, he uses the one-time password displayed on the token in addition to his normal account password. Software tokens are programs that run on computers and provide a one time password that it is changed after a short amount of time (usually 30 seconds to 10 minutes). OTP algorithm's security is very important because no one should be able to guess the next password in sequence. The sequence should be random to the maximum possible extent, unpredictable and irreversible. Factors that can be used in OTP generation include PIN, time, DOB, etc.

Even if hardware tokens provide a safer environment for users, they can be costly for companies. For example, a bank with a million customers will have to purchase, install and maintain a million tokens, as well as provide support for non-initiated users on how to use the tokens. The bank also has to be ready to replace any broken or stolen token. Replacing a token is a lot more expensive than replacing an ATM card or than replacing a password. From the user's perspective, having accounts at multiple banks comes with the need of carrying and maintaining several tokens, which represents a big inconvenience. In many cases, customers are being charged for lost, stolen or broken tokens. The following is the description of an application that is cost-efficient for the companies that will choose to implement it. It will also allow customers to install multiple software tokens on their mobile phones, thus making them worry only about their mobile phone instead of worrying about several other tokens.

3 DESIGN IMPLEMENTION

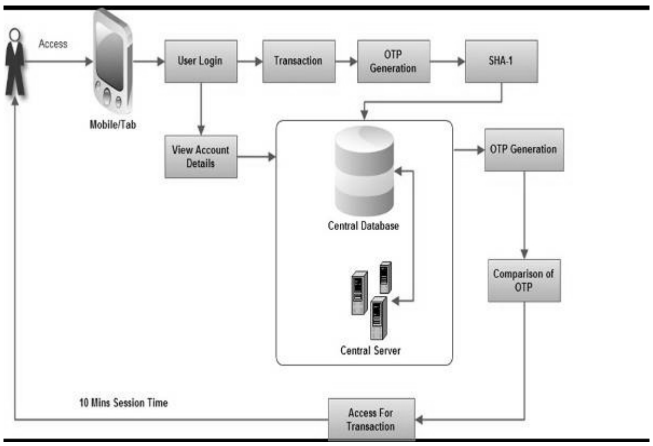

Figure 2: Proposed work and System Architecture

In this paper, we propose a mobile-based software token system that is supposed to replace existing hardware and computer-based software tokens. The proposed system is secure and consists of three parts: (1) software installed on the client’s mobile phone, (2) server software, and (3) a GSM modem connected to the server. The system will have two modes of operation:

- **Connection-Less Authentication System:** An onetimepassword (OTP) is generated without connecting the client to the server. The mobile phone will act as a token and use certain factors unique to it among other factors to generate a one-time password locally. The server will have all the required factors including the ones unique to each mobile phone in order to generate the same password at the server side and compare it to the password submitted by the client. The client may submit the password online. A program will be installed on the client’s mobile phone to generate the OTP.
- **SMS-Based Authentication System:** In case the first method fails to work, the password is rejected, or the client and server are out of sync, the mobile phone can request the one time password directly from the server without the need to generate the OTP locally on the mobile phone. In order for the server to verify the identity of the user, the mobile phone sends to the server, via an SMS message, information unique to the user. The server checks the SMS content and if correct, returns

a randomly generated OTP to the mobile phone. The user will then have a given amount of time to use the OTP before it expires. Note that this method will require both the client and server to pay for the telecommunication charges of sending the SMS message.

3.1 OTP Algorithm

The unique OTP is generated by the mobile application offline, without having to connect to the server. The mobile phone will use some unique information in order to generate the password. The server will use the same unique information and validate the OTP. In order for the system to be secure, the unique OTP must be hard to predict by hackers. The following factors will be used to generate the OTP:

IMEI: International Mobile Equipment Identity, unique to each mobile phone and allows each user to be identified by his device. This is accessible on the mobile phone and will be stored in the server's database for each user.

IMSI number: The term stands for International Mobile Subscriber Identity which is a unique number associated with all GSM and Universal Mobile Telecommunications System (UMTS) network mobile phone users. It is stored in the Subscriber Identity Module (SIM) card in the mobile phone. This number will also be stored in the server's database for each client.

ATM PIN: Needed for verifying the authenticity of the client. If the phone is stolen, a valid OTP can't be generated without knowing the user's PIN. The PIN isn't stored in the phone's memory. It is only being used only to generate the OTP and destroyed immediately after that.

Timestamp: Used to generate unique OTP, valid for a short amount of time. The timestamp on the phone must be synchronized with the one from the server.

DOB: Date of birth of user whose going to use the application.

Algorithm 1: OTP Generation Algorithm

Input: IMEI, IMSI, ATM Pin, Timestamp, DOB
Output: One Time Password

```
Accept uniq data of user. (data=IMEI, IMSI etc)
randomString=IMEI+IMSI+TimeStamp+DOB
SHA-1 (randomString)
byte barr = new byte [5]
for i=0 to i<5 do
    barr[i] = (byte)(sha1hash[(i+0)] ^ sha1hash[(i+5)] ^
    sha1hash[(i+10)] ^ sha1hash[(i+15)]
for i=0 to i=barr.length do
divide string into 2 equal length of halfbyte
do
    if(0<=halfbyte & halfbyte <=9)
      then '0'+halfbyte
    else
    'a' + (halfbyte – 10)
    End if
while(two_halfs++ < 1)
End for
```

The above factors are concatenated and the result is hashed using SHA-256 which returns a 256 bit message. The message is then XOR-ed with the PIN replicated to 256 characters. The result is then Base64 encoded which yields a 28 character message. The message is then shrunk to an administrator-specified length by breaking it into two halves and XOR-ing the two halves repeatedly. This process results in a password that is unique for a ten minute interval for a specific user. Keeping the password at 28 characters is more secure but more difficult to use by the client, since the user must enter all 28 characters to the online webpage or ATM machine. The shorter the OTP message the easier it is for the user, but also the easier it is to be hacked. The proposed system gives the administrator the advantage of selecting the password's length based on his preference and security needs.

3.2 Client Design

An Android program is developed and installed on the mobile phone to generate the OTP. The program has an easy-to-use GUI that is developed using the NetBeans drag and drop interface. The program can run on any Android -enabled mobile phone. The OTP program has the option of generating the OTP locally using the mobile and user credentials, e.g. ATM Pin and IMSI numbers, or requesting the OTP from the server via an SMS message. The default option is the first method which is cheaper since no SMS messages are exchanged between the client and the server. However, the user has the option to select the SMS-based method.

Generating the unique passwords is an offline process once the user has installed, configured and synchronized the application on his mobile, for which an active internet connection is required.When the application starts, the user will be prompted for the ATM PIN, DOB credentials. Since the application does not store the PIN code, it will not be able to verify if the user entered the right one or not. If the PIN is different from the one used at registration, the OTP result will be different from the one computed on the server and therefore the user won't be able to authenticate in the Internet Banking application.

The unique OTP is generated with the OTP algorithm, using the hash of the credentials (stored on the phone), the phone's IMSI (requested when opening the application) and the user's ATM PIN (also requested every time the user uses the application). These credentials are then signed with the current timestamp, as defined in OTP. The one-time password will be valid for 30-10 minute's seconds since its generation.

3.3 SHA (Secured Hash Algorithm)

Hashing which is used in many encryption algorithms is the transformation of a string of characters into a shorter fixed-length value or key that represents the original string. The hashing algorithm is called the hash function. A cryptographic hash function is a procedure, which takes a block of data and gives a fixed-size bit string, the (cryptographic) hash value. They have many information security applications, including digital signatures, message authentication codes, and other forms of authentication.

SHA (Secure Hash Algorithm) is one among a number of cryptographic hash functions. It is a series of cryptographic hash functions: SHA-1, the 160-bit version. SHA-2, a newer revision with four variants: SHA-224, SHA-256, SHA-384 and SHA-512.

Though SHA-2 has some similarity to the SHA-1 algorithm, it includes a significant number of changes from SHA-1, and security flaws identified in SHA-1 are avoided.

SHA-256 and SHA-512 are new hash functions computed with 32 and 64-bit words, respectively and use different shift amounts and additive constants, differing only in the number of rounds. SHA-224 and SHA-384 are truncated versions of the first two, computed with different initial values.

SHA-256 is used to authenticate and in the DKIM message signing standard. SHA-512 is part of a system to authenticate archival video. SHA-256 and SHA-512 are proposed for use.

3.4 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, pin, password, mobile IMEI number, IMSI number, unique symmetric key, and the mobile telephone number for each user. The password field will store the hash of the 10 minute password It will not store the password itself. MySQL is used as a back-end database.

3.5 Server Design

A server is implemented to generate the OTP on the organization's side. The server application is multithreaded. The first thread is responsible for initializing the database and SMS modem, and listening on the modem for client requests. The second thread is responsible for verifying the SMS information, and generating and sending the OTP. A third thread is used to compare the OTP to the one retrieved using the connection-less method. In order to setup the database, the client must register in person at the organization. The client's mobile phone/SIM card identification factors, e.g. IMEI/IMSI, are retrieved and stored in the database, in addition to the username and PIN. The J2ME OTP generating software is installed on the mobile phone. The software is configured to connect to the server's GSM modem in case the SMS option is used. A unique symmetric key is also generated and installed on both the mobile phone and server. Both parties are ready to generate the OTP at that point.

Table 1 Existing and Proposed System

S. No	Existing System	Proposed System
1	Transaction deal with single authentication	Transactions is accessed by handling security token
2	Possible of brute-force attack	Secure transactions will be maintained
3	No storage of customer details for Future Reference(except Customer's copy)	Stores customers digital certificates in the database

4 Conclusion

Today, the use of one-factor authentication (e.g. password) is not considered secure anymore in the internet banking world. Passwords which are easy to guess, like name, date of birth are sure targets for automated password collecting programs.

Two-factor authentication has been recently introduced to meet the demand of organizations for providing a stronger and safer authentication process for their users. In most cases, a hardware token is given to each client for every Internet Banking application he uses. With the number of users that are requesting for a token viral increasing, the cost for manufacturing, maintaining and replacing them is becoming a burden for both organizations and clients. Since most of the clients carry a smartphone at all times, an alternative is to install all the tokens on the mobile phones, as applications. This approach will help in reducing the costs and the number of devices carried by the client.

This paper focuses on the implementation of two-factor authentication on any smart-phone that allows third-party developers to add and run applications (such as Apple's iOs, Android, BlackBerry OS, Windows Phone, Symbian). The implemented demo application was created for Android, but can be easily written on any other operating system, using the design implementation described.

Acknowledgment

We would like to thank our Institution and faculty members, without whom this Paper would have been a distant reality. Last but not the least, we would also like to thank to our friends for listening to our ideas, asking questions and providing feedback and suggestions for improving our ideas.

References

[1] Fadi Aloul, Syed Zahidi, "Two Factor Authentication Using Mobile Phones," in *Proceedings Proceedings of the IEEE International Conference on Computer Systems and Applications, pg. 641-644, 2009.*

[2] RFC 4226, HOTP: An HMAC-Based One-Time Password Algorithm, *http://tools.ietf.org/html/rfc4226*

[3] TOTP: Time-based One-time Password Algorithm, *http://tools.ietf.org/html/draft-mraihi-totp-timebased-00*

[4] Alecu F., Internet Banking, *Informatica Economică*, nr. 4 (40), 2006, pp. 104 – 106, ISSN 1453-1305

[5] D. de Borde, "Two-Factor Authentication," Siemens EnterpriseCommunications UK- Security Solutions, 2008. *Available at http://www.insight.co.uk/files/whitepapers/Twofactorauthentication(Whitepaper).pdf*

[6] Wikipedia, Time-based One-time Password Algorithm, *http://en.wikipedia.org/wiki/Time-based_Onetime_Password_Algorithm*