Survey on Iot Security Issues And Solutions

Vishakha Lall[1], Sourabh Kumar Jain[2]
[1]Dept of CSE
[2] Assistant Professor, Dept of CSE
[1, 2] Gyan Ganga Institute of Technology and Sciences, Jabalpur, Madhya Pradesh, India.

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. 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 random image and text based OTP generation with SHA-512 algorithm and again the concept of actual and fake OTP is introduced in the work.*

Keywords- One Time Password (OTP), Image Based OTP, and SHA based One Time Password, Time-based One Time Password (TOPT), Cryptography, Email, and Authentication.

I. INTRODUCTION

Implementing easy to use, yet secure user authentication methods is a challenging task. While passwords are still the most widely used authentication method for web services, they come with tremendous disadvantages concerning both, usability and security [1]. In 2016, more than 81% of all security-related breaches were feasible because of weak or stolen passwords [2]. Multi-factor authentication schemes are deployed in corporate environments from time to time, where hardware tokens (e.g. smartcards) are used in addition to passwords. However, after a successful implementation the costs of operation, replacement and administration are often considered too high. In cases where smartcards are bundled with Universal Serial Bus (USB) interfaces, security issues could develop, as these could be compromised or replaced by malicious counterfeits [3]. Finally, the user-friendliness of most tokens is perceived as low since users are always required to have them in reach. Nowadays, an increasing number of popular web services are enhanced by multi-factor authentication schemes which are user friendly and affordable. These schemes rely on a smartphone as *something the user has* and a password which is *something the user knows*. A valid login or verification requires the legitimate user to employ both provided factors.

Security, convenience, and functionality are all essential factors for having an efficient authentication process. However, there are tradeoffs between these factors. When one increases, the others might decrease. There is an inverse relationship between security and convenience where organizations stand on a fine line between both. Therefore, there is an urgent need to find balance between security and convenience; otherwise, the organization will fall into either side. Unfortunately, nowadays, convenience keeps winning, which causes sacrificing security. Using passwords as an authentication method, despite the convenience, is full of security flaws [4]. One problem is that users may prefer passwords that are easy to remember, which makes them vulnerable to different types of attacks including guessing, dictionary, and brute force. Even if we suppose that the user was careful enough to set a strong password, he might fall a victim for a simple, yet smart, phishing attack. The increasing number of attacks on passwords has led organizations to force strict policies on how the user sets his password in order to provide an appropriate level of security.

On the other hand, those policies have raised the inconvenience issue for the user, when the member needs to access his accounts frequently, not mentioning that the login session is relatively short in some systems. The main question here is “How to increase the security of the authentication mechanism without scarifying the convenience?” Authentication is the process of ensuring the identity (username, PIN code etc.) of an entity, person or process to which it was issued. Authentication depends on the information provided by the user, it is either something you know, something you are, something you own or a combination of all of them. The main goal of the authentication process is securing the data accessed after the authentication process. Therefore, designing a secure authentication system is a sensitive process that requires paying attention to the tiniest details.

In this thesis, the proposed authentication system has been studied carefully to satisfy the system’s requirements. It

is important to note that computers are getting faster and more capable every day. As a result, an 8-character password is not safe against brute force attack as it used to be. Unfortunately, this issue has not been taken into consideration by other researchers focusing on solving authentication issues [5]. Nowadays, many researchers are headed toward graphical password approaches to solve the password inconvenience issues. Unfortunately, despite their convenience, graphical passwords raise new problems themselves as will be shown later in this paper.

Hence, the main contribution of our thesis is to provide security and convenience without sacrificing with one for the other, using PassText and OTP. PassText mechanism supports security and convenience at the same time. In this paper a new methodology to be proposed to enhance the use of PassText by hashing the whole text instead of certain parts, which helps in increasing the system's security. Furthermore, in this thesis, we introduce convenient re-authentication to provide a convenient authentication experience. Hence, the main contribution of our thesis is improving the one time password mechanism through processing the whole text and not only a part of it. Then, hash it using SHA algorithm. Additionally, this thesis focuses on increasing the security and convenience of the system that uses Fake and Actual OTP by adding a second factor of authentication which is a re-authentication mechanism that requires Actual OTP only if the user has the original OTP converter.

1.1 One Time Password

One-time passwords have mainly been used in Internet banking systems up to this point, and they were seen as a secure authentication method. However, in February 2015, Japan's National Police Agency published case studies regarding illegal remittances in Internet banking for 2014 [6], indicating that over 100 financial institutions had incurred losses, and that damages were also on the increase. Particular attention was given to attacks targeting corporate accounts that involve comparatively large remittance amounts, and an alert was issued. In addition to methods that simply use a password, those that incorporate client authentication based on X.509 certificates (strong authentication) have also been introduced for corporate Internet banking authentication, and these are known to provide more security. However, when these digital certificates are used there have still been cases disclosed in which automatic transfers have been made from PCs or browsers infected with malware [7]. This demonstrates that even authentication methods using SSL/TLS client certificates are not foolproof.

Based on these circumstances, an effort is being made to improve the authentication methods used in Internet banking systems. In the past, random number tables listed on paper or card was used, along with hardware devices that display a one-time password. This latter case is an authentication method in which a temporary password is input at the same time as the primary PIN number (a 4-digit number) that corresponds to the password used for authentication at ATMs, etc. Even if the primary PIN number leaked, use of a onetime password that is discarded each time reinforces the identity verification with particularly important processes, such as address changes or large remittances. However, as shown by Man-in-the-Browser attacks and MITM attacks, if banking system transaction details such as the destination account number or transfer amount are re-written, illegal remittances are possible even when a disposable one-time password is used. This demonstrates the problem of it being impossible for a user to explicitly confirm whether or not a transaction is legitimate based on the information shown in the browser alone, even if authentication methods are improved. There is an undeniable possibility that transactions may have actually been rewritten by an attacker. In response to this problem, progress is being made towards migrating to the use of hardware devices equipped with an input device. In the past, measures featuring the combined use of hardware devices during authentication have also been adopted at a number of banking systems. However, because this hardware device was simply a one-time password generator with no input interface, it could only be used to identify whether the user has the correct token, which has nothing to do with the transaction at hand. In addition to the abovementioned use of X.509 certificates, corporate banking systems also incorporate secondary measures such as only accepting transactions from specific IP addresses and PCs. However, because it is possible that the transactions a user sees have been rewritten, there is no fundamental countermeasure for Man-in-the-Browser attacks.

In other words, even with improved identity verification the countermeasures were ineffective. In response, there were announcements from financial institutions in 2015 that they would start using one-time password cards. Unlike previous devices, these new hardware devices have a keypad input interface, enabling identity verification while also incorporating techniques that enable users to confirm the legitimacy of transactions that may have been rewritten. The devices are not merely for inputting the one-time password that is output from the hardware device during authentication like before. They feature functions for generating and displaying one-time passwords that guarantee the correctness of account numbers, by having users themselves input the account number they want to transfer money to. This prevents

money being sent to the account intended by an attacker, and by also recording the transaction log generated at this time, it is possible to automatically create blacklists for the attackers accounts. Additionally, for user convenience, they can also be used as one-time password generators with no input device. The technique is based on the premise that accounts registered by users in advance are safe, and omits the input of account numbers for transfers to these pre-registered accounts. Broadly speaking, two specifications are currently being drawn up by the FIDO Alliance. The UAF (Universal Authentication Framework) standard deals with biometrics authentication, while the U2F (Universal 2nd Factor) standard deals with multifactor authentication.

1.2 Authentication Requirements:

Phishing is a cyber-fraud which uses deception techniques to break secure authentication schemes. Most of the existing authentication schemes, either one way (user authenticates himself to the website) or two way (the website and user both authenticate each other), are vulnerable to the latest cyber phishing attacks [8, 9] such as Real Time (RT) and Control Relay (CR) Man In The Middle (MITM) phishing attacks [10] and malicious browser extension based phishing (MEP) attack. MEP attacks generally involve key-logging, password & form data sniffing, screen logging etc. The consistent number of phishing attacks can be inferred as an indirect indication of the ease with which existing authentication schemes can be compromised.

RT MITM Phishing: In an RT MITM phishing attack, attackers place themselves between a client and a server by means of a phishing website appearing as a genuine one. The attacker captures the authentication information entered by the user on the phishing website and relays this information to the genuine website in real time via manual or automated means, thereby gaining access to the user's account. RT MITM, unlike traditional phishing, can utilize remote desktop monitoring modules, malicious browser extensions/screen loggers that can help in providing the additional information (keystrokes, CAPTCHAs, QR codes etc.) in real time to break the authentication scheme.

CR MITM Phishing: CR MITM is more invasive. In a CR MITM phishing attack, the attacker relays his desktop over the user's terminal, eventually deceiving the user into entering his credentials directly on his computer. Both one-way and two-way authentication schemes are vulnerable to such attacks as an authentication token provided by the user can be captured on the phishing website and can be relayed to the genuine website in real time to complete a successful authentication. Only separate hardware token based schemes or schemes which store at least one part of user credentials over the client can handle such attacks.

MEP Attacks: Malicious browser extensions can also be used to perform phishing attacks for stealing user credentials. Schemes for malicious browser extension detection have been proposed in the past [11, 12], but little work has been done in this area. Malicious browser extensions can acquire permissions needed for carrying out any stealth activity by providing functionality to the users in the foreground that requires the same set of permissions. For example, a malicious browser extension can provide grammar checking facility to the users in the foreground and hence can get permissions to access the contents of websites opened in the browser and tab related data (URL typed in the tab's address bar). Using these permissions malicious browser extensions can carry out credential stealing, spying and phishing activities in the background.

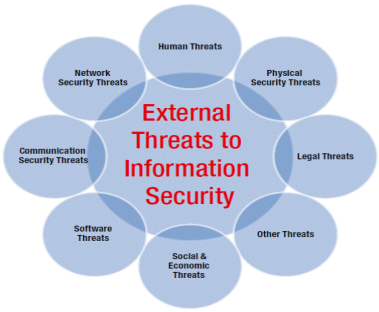

Figure 1.1: Threats to Security.

Malicious browser extensions can also be installed covertly by an insider on the victim's PC. A malicious browser extension can perform key logging, screen logging, or password sniffing to steal credentials. No matter whether the scheme is a CAPTCHA based scheme, a picture password-based scheme, or a dynamic security skin based scheme, it can be compromised if a malicious browser extension running on a user's PC captures the screen and relays this information to the attacker in real time. A malicious browser extension can sniff the information entered on the browser even before the application or transport layer (TLS) encrypts it, hence password manager based schemes can also be compromised. In one of our recent publications, we have discussed in detail the attacks which can be launched via malicious browser extensions. Most of the existing multifactor authentication schemes are incompetent in handling the attacks described above. Our experiments show that OTP/PIN-based schemes, QR/Barcode based schemes, Password manager, and push notification based login schemes is vulnerable to these attacks. Graphical password-based schemes can be phished using a malicious browser extension that can log the screen when the user enters his password or CAPTCHA on the website opened

in the browser. Also, graphical password based schemes are not user-friendly. Biometric authentication is still not 100% accurate, robust, mature, and user-friendly. Environment and usage can affect the measurements and they also need additional hardware. User-friendly biometric schemes which are commercialized (such as fingerprints, facial recognition etc.) can be spoofed. Separate hardware token based schemes (such as Yubikey U2F, RSA SecurID, DUO etc.) provide a better layer of security compared to the other schemes but they have following drawbacks.

- Firstly, the user needs to buy and carry these hardware tokens always which make them nonuser friendly.
- Secondly, some of the hardware token based schemes that use security keys for OTP/PIN generation and their subsequent entry on browsers can be compromised via malicious browser extensions through sniffing of HTML form data during its submission.
- Thirdly security keys such as Yubikey and RSA SecurID tokens can also be compromised through reverse engineering and spoofing onto other hardware .

New protocols such as Yubikey U2F may handle most of the sophisticated attacks but the need for buying and carrying a separate authentication token makes them unattractive. Also, attacks that weaken the strength of RSA key generation on Yubikey have been recently recorded in October 2017 [30]. This can be inferred from the ratio of the number of users who use separate hardware authentication tokens to log in over websites to the number of users who use soft tokens generated from Authenticator Apps installed on their smartphones or OTPs as a second factor for authentication.

Hence there is a need for immediate research and development of secure authentication schemes which can address the latest phishing threats from the latest cyber phishing attacks. The schemes should also be easy to use and must use existing hardware and/or technology so that the cost incurred for carrying out login authentication can be reduced.

II. LITERATURE REVIEW

In this section important background information is provided. It starts with a general overview of OTP methods. Subsequently, the employed QR based OTP authentication method is introduced. Providing baseline security for the presented remote maintenance scheme, the contribution of overall OTP based communication security is presented. Finally, the limitations of the baseline security are highlighted, motivating the proposed improvement method using two OTPs.

A. OTP Primer

The technology that helped to enable the wide distribution of multi-factor authentication relies on OTPs. These are passwords that are valid only for the time of one login session or transaction and that are generated with algorithms based on cryptographic hashes. Instead of handing out special hardware to generate OTPs, it is becoming increasingly common to generate them on smartphones or utilise text messages as well as voice call transmissions. The underlying OTP algorithms can be divided into the following three categories:

1. Counter-based OTP generation.
2. Time-based OTP generation, and
3. challenge-response-based OTP generation.

B. QR Authentication Primer

In 2015, the usage of OTPs as an authentication method for industrial applications was presented in [13]. The publication introduced a yet novel OTP scheme combining challenge response-based OTP generation for client authentication with a time factor and industrial component data. Since then it has been further refined, reviewed and implemented. One significant enhancement compared to the initial publication concerns that the scheme is now capable of mutual authentication between server and client. The scheme can either be used to replace password authentication or as an additional second factor enhancing a different authentication method (e.g. password or certificate based). To mitigate simple replay attacks, mutual authentication was identified as important.

C. TLS Primer

To provide baseline security for the employed communication channels outlined in the previous section, a cryptographic protocol implementing the protection goals Confidentiality, Integrity and Authenticity (CIA) is utilised. Realisation of CIA for communication channels requires interaction of different cryptographic functions. As each of these functions typically addresses one specific protection goal, TLS combines them to implement the full set of protection goals for the specific use-case. It is generally not recommended to specify new cryptographic protocols if there are existing and proven schemes. Therefore, the trustable and widely used TLS protocol was chosen to meet the security requirements.

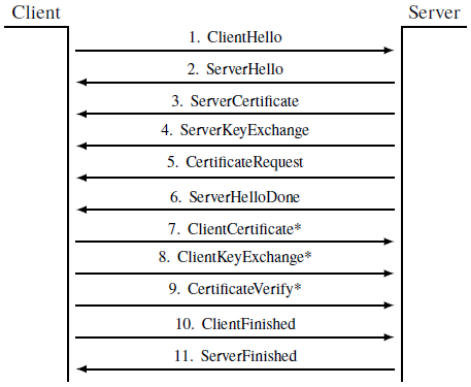

Figure 2.1: Simplified Mutual TLS Handshake – optional steps marked with *.

Figure 2.1 illustrates the basic steps of a full-fledged TLS handshake. Combining cryptographic functions, each phase of the protocol has the responsibility to assure a certain protection goal. The protocol starts with a hello-procedure, where client and server agree on the protocol version and a set of cryptographic functions supported on both sides. This set is known as cipher suites. Additionally, random numbers are exchanged which is the basis for reliable cryptography. As these are never used twice, they also contribute to the uniqueness of the handshake which is one element in spoofing-attack prevention. Since both parties are not yet familiar with each other, spoofing is feasible. Therefore, in the next step the client validates the authenticity of the server. Being the foundation for spoofing protection, this step is mandatory in the protocol specification. For this task TLS makes use of asymmetric cryptographic functions where a public available key is used to prove the authenticity of an entity. To facilitate scalability, the public key is embedded into a certificate signed by a third party authority known and trusted by the client. The certificate is received by the client who has to validate its integrity and the authenticity of the provided signature. Subsequently, the public-key is extracted to validate the authenticity of the remote server. If these checks have been successfully completed, the communication channel is unilaterally authenticated. This guarantees a direct connection which is not degraded by a MitM attack. For use-cases where client systems are also required to prove their authenticity to the server, TLS provides an optional feature marked with an asterisk in Figure 2.1. The client authentication follows the same scheme as it was described for server authentication.

On successful completion a mutually authenticated TLS channel is established. Although, mutual TLS authentication greatly reduces the attack surface for MitM attacks, it increases the administrative burden for the PKI. Therefore, password-based client authentication is still widespread, especially in industrial environments. Provided that the client carefully checks the server's authenticity, an authentication inside an established unilateral authenticated TLS channel is also a valid approach and not impairing overall channel security at all. After successful authentication a symmetric secret is agreed between server and client. As it is only valid for the lifetime of the established channel, it is called session-key. To ensure that an eavesdropper is not able to reconstruct this secret, some of its components are chosen by the client and sent to the server using an asymmetric cipher. Therefore, it is also ensured, that only the authenticated peer has access to the session-key. For the lifetime of the established channel, the session-key ensures confidentiality and authenticity for the transferred data. For all described operations the use of different cryptographic functions is possible. The actual used ciphers are negotiated during the hello-procedure. Ruling out that an attacker has successfully mounted a MitM scenario, both parties create a cryptographic hash of their complete handshake-data-history. Finally, these values are exchanged. If the hash values are the same on both sides, the handshake process is successfully completed and the secure channel is established.

2.1. Smartphone Centered Systems

Cheong et al. [14] used NFC smartphone plus encrypted steganography graphical password (ESGP) as a two-factor authentication system to unlock doors of hotels and etc. users feed a photo to the system at the registration time to generate stego-photo. For usability achievement, Least Significant Bits (LSB) technique was used for the sake of speed whereas other techniques like DCT and DWT were refused because of their complexity Decoding, verification part of steganography key and graphical password are handled on the server side. In the case of sharing or stealing steganography key, no serious threat exposes the system to any risk. This is because the second factor of authentication asks users to specify their valid graphical password through their application which is installed on their phone. In the worst case, the intruder must have access to all images of the victim's smartphone and the graphical password patterns. Thus, the system can use a biological factor as its second factor to authenticate users to enhance the security perspective in the worst case.

Azimpourkivi, et al. [15] changed the common ways of using a camera in authentication systems. Pixie is responsible to capture an object which is unique to the user called trinket. A trinket here can be any object which has unique features among other objects to the users. Actually users must carry the object to pass further authentication attempt. So the original picture will be compared to the captured image for authentication. If both match together, the

user can log in to its smartphone. In this authentication system, password picture is mixed with what user has and what he knows about the dimensions of the object to capture in order to provide a two-factor authentication. Pixie achieved 0.02% of False Accept Rate and 4.25% of False Reject Rate in 122,500 attempts.

Crossman & Liu [16] presented a two-factor authentication based on NFC smartphone devices. The system is combined with two models, what you have and what you know. Firstly users are asked to enter a passcode which unlocks the protection of the key on their mobile phones. Then, the key is transferred through NFC to complete the rest of login process. Although the privacy and protection of the key-store leaves to android OS sandboxing, there is not an alternative solution in order to confront with zero-day vulnerabilities.

Chakraborty et al. [17] named their system MobSecure to provide more dependable security in smartphones authentication for unlimited attempts against bystanders. In the authenticating application, there are two static orbits which contain digits and alphabets in different colors. Every two nodes have the same color in each orbit. This is why a challenge will ask the user to respond the puzzle according to his secret password. Consecutively, the user would not locate his real password on the two orbits and each time he points to different nodes in two orbits as the challenges are randomized. Even If the attacker has access to challenges, still he cannot find out the answers and this is why he doesn't know the secret password at the time of registration.

Abdurrahman, et al. [18] proposed a new system which utilizes user's current GPS location and a pre-hash number to produce a secret hash code to increase security aspect like multi-factor authentication. As traditional verification systems, after providing username/password for the system, the server waits for the security token. The application on that user's smartphone generates a security token according to the aforementioned formula and sends GPS location and Timestamp to synchronize the server. As a security prevention mechanism toward stealing a smartphone, the application is protected with PIN code; however, reverse engineering for finding strings or even excavation of private isolated application's space is not a difficult issue. The elimination of additional hardware as a security token is acceptable and cost-effective.

Siadat et al. [19] proved an attack against many proposed two-factor authentication systems based on SMS. The heart of the problem lies in the type of confirmation code message which was sent to the user who requested reset-password. They studied different models of phishing text messages which could convince the user to forward the received verification code to the attacker. As a result, they developed some principles for more informative text messages to reduce the successfulness attack rate. Nematollahi et al. (2016) developed a multi-factor authentication scheme based on speech, PIN code and OTP which were mixed up to produce a reliable watermarking. The combination of the biometric of the user and the PIN code with applying a key to generate a hash to use in watermarking technique; does not let the password crackers to run brute force attack simply. By concealing the PIN code and the OTP within speech signal, they considered that it is hard to Eavesdrop or theft these items. Also, the uniqueness of the OTP as a time-stamp can stop session hijacking threat.

III. PROPOSED METHOD

Our proposed system will generate secure One Time Password using text encryption with image. It uses SHA512 for this purpose. For providing extra layer of security proposed system will generate two OTPs – one is fake OTP (send to user) and another is actual OTP (used at the time of authentication). Genuine user uses an application which converts received fake OTP into actual than user enters actual OTP for authentication. This extra layer protects the system in various intrusions like email hack, device theft etc. The system is based on a synchronous stream cipher that uses images, instead of passwords, as the secret key. A synchronous stream cipher is a type of symmetric key algorithm that generates a pseudo-random sequence of bits, called the key stream, independent of the plaintext and cipher text. These bits are then combined with the plaintext bits (usually using exclusive-or) to produce the cipher text, and then system will generate two OTP from cipher text. One is Actual OTP and another is Fake OTP. This Fake OTP will be sent to user's email or mobile. If user enters same OTP for authentication it will not work. Authorized user should use an application for converting this fake OTP to generate actual OTP. Than this actual OTP will be entered for authentication, , it increases more security in the system.

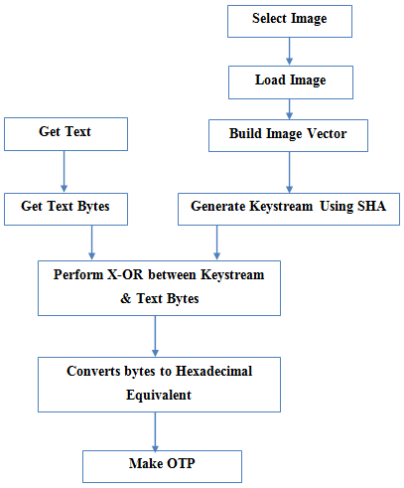

Figure 3.1: Proposed OTP Generation Method.

OTP generation starts by loading the image into memory and getting the input text bytes, and then building a vector by applying a transformation function to the image's pixels to be used later as the secret key. The system will then generate the keystream by combining multiple keys together. A single key is generated by a sequence of bit-shifting the image vector, then hashing it (using one of the Secure Hash Algorithms) and finally performing an exclusive-or between the image vector and the hash value. After generating all the keys required so that their combined bytes are equal to or greater than the input text bytes, the remaining process is simply performing an exclusive-or operation between each keystream byte with the input text bytes. The system will then represent the resulting bytes by a readable form, which may be the hexadecimal values of the encrypted bytes. Theses bytes will produce fake and actual OTP.

3.1 Process of Building the Image Vector

After loading image proposed system will now going to build the image vector by applying a transformation function on the image's pixels.

The transformation function is (red + green + blue) modulo alpha, and it is illustrated in figure 3.3.

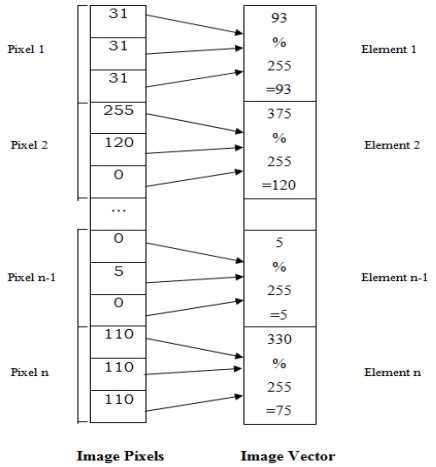

Figure 3.2: Image Vector.

The final result of this step is a byte vector of length equal to the number of pixels, and this vector will represent the secret key in the coming steps. To do the transformation, we will first need to gain access to the pixel's underlying values (i.e. the red, green, blue and alpha values). If the image is of type TYPE_3BYTE_BGR, it will be processed by a method that assumes each pixel is represented by 3 elements in the array, and it will always use an alpha value of 255.

3.2 Steps for OTP generation are as follows:

Step-1. System will have to select one random image and random text field from large corpus.
Step-2. Select OTP generation method and number of threads.
Step-3. Load image & text field.
Step-4. Build the image vector for generating the key stream using hash function.
Step-5. Key stream bytes and text bytes are XOR.
Step-6. Convert output bytes to hexadecimal equivalent.
Step-7. Select first six characters of cipher text & generate actual OTP.
Step-8. Encode this actual OTP into fake OTP by own encoding method. This will be sent to user's mail id or mobile.

3.3 Generation of fake OTP

For generating fake OTP input is actual OTP. System initializes a new string as empty, than selects character by character from actual OTP, take a new predefined different character for each one, and concatenate with new string. After conversion of all characters from actual OTP, new string will be encoded fake OTP. At user side reverse approach will be used for generating actual OTP.

IV. RESULTS & EVALUATION

Information and data security are based on factors such as authenticity, accuracy, availability, data credibility, confidentiality and no repudiation. The proposed approach has the ability to contribute to the necessary data and information security. It uses authentication token as certificates to prove authenticity. Proposed system generates One Time Password by manual selection of image, text, method and number of threads because parameters can be analyzed, but in actual system these will be automatically selected. Automatic selection of images and text from large corpus will increases randomness in the system, which will increase more security also. Image and text may also be selected in real time from web. So there is no way to guess selected image and text. Our proposed system will generate OTP in alphanumeric form which is also more secure than numeric OTP.

Proposed system will generate two OTP- one is actual and another is fake (generated from actual by own encoding method). Actual OTP is stored at the side of server while fake OTP will be sent to user. So every user will get fake OTP whether user is authorized or not. If unauthorized user enters fake OTP for authentication, it will not match. Authorized user should use an application provided by system for converting fake OTP into actual OTP. Then user can be authorized. So it will provide more security against device theft or email hacking. Proposed system uses SHA 512 method to encrypt text from image. Key stream will be generated from image which is more secure than other versions of SHA like 128, 256. Our proposed system surpasses all the problems of password based mechanism. It keeps resistance against the following security hazards and susceptibility:

1. **Token theft**: Since we have two security tokens as OTP- actual and fake OTP. User should also require an application to decode fake OTP into actual OTP. There is no chance of token theft.
2. **Token Duplication**: Due to randomness of image and text there exists no chance of duplication of OTP.
3. **Replay Attack**: No chances of replay attack.
4. **Eavesdropping**: OTP received by user is fake. Fake OTP makes eavesdropping almost impossible for attackers.
5. **Man-in-the-middle attack**: System protects against Main In the Middle attack because of system generated fake OTP and use of own application for conversion of fake OTP into actual.

The proposed system has been evaluated on two main parameters

1. OTP Generation Time
2. Key stream Generation Time

For evaluating different algorithm we have taken three samples with fixed image and fixed text field. Experimental parameters are represented below in terms of table and chart for each sample.

4.1 Evaluation on the basis of OTP Generation Time

Table and chart below represents comparison of OTP generation time for all samples:

Table 4.1: Comparison of All Samples.

Algorithm	OTP Generation Time (ms) Sample 1	OTP Generation Time (ms) Sample 2	OTP Generation Time (ms) Sample 3
SHA 128	14	24	9
SHA 256	13	22	8
SHA 512	11	19	6

Chart representing above table is show below:

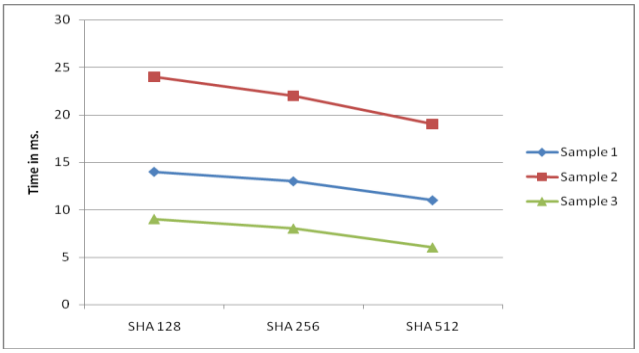

Figure 4.1: Comparison Chart for All OTP Generation Sample.

4.2 Evaluation on the basis of Key Stream Generation Time

Table and chart below represents comparison of key stream generation time for all samples:

Table 4.2: Comparison of All Key stream Generation Sample
Chart representing above table is show below:

Algorithm	Key Stream Generation Time (ms) Sample 1	Key Stream Generation Time (ms) Sample 2	Key Stream Generation Time (ms) Sample 3
SHA 128	8	13	6
SHA 256	6	11	5
SHA 512	5	10	4

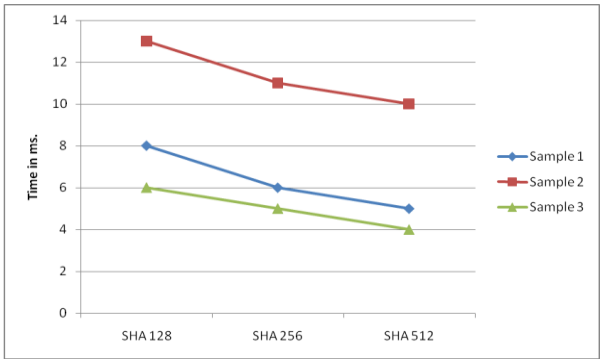

Figure 4.2: Comparison Chart for Key stream Generation

All results show that SHA 512 performs better in all samples for OTP generation as well as key stream generation. Since OTP generations are mostly used for authentication and when number of users is larger, we also apply multithreading for generating key stream. This will enhance speed. So sample OTP generation with multithreading are summarized below:

Table 4.3: Comparison of All Key stream Generation Sample Chart for comparing samples with multithreading are shown below:

NO OF THREADS	SHA 128	SHA 256	SHA 512
1	204	31	26
3	58	41	37
5	46	62	41

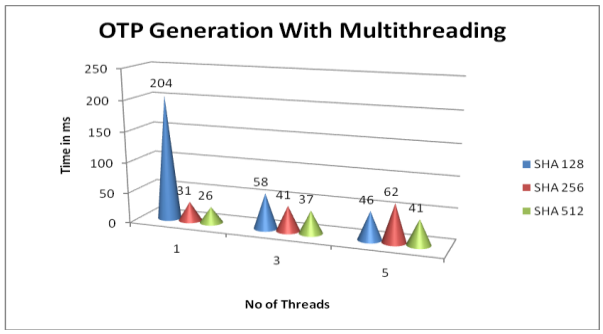

Figure 4.3: Comparison Chart in Case of Multithreading.

V. CONCLUSION AND FUTURE WORK.

To sum up, authentication systems can be simple at the first glance. But in fact, they are complex in security, usability and availability aspects. As poorly chosen passwords could not protect users properly, Multi-factor authentication systems presented in different ways to increase the reliability of authentication systems. In this thesis, we reviewed many of those factors in different systems such as SMS, time-based and hardware-based tokens. Despite the security advantages and deficiencies, each of the aforementioned factors affects the usability, cost-effectiveness, availability and implementation of an authentication system. Clearly, the user must come first in a multi-factor authentication system and since the influence of the technical issues is significant, it is an arduous task to achieve this trade-off in an authentication system. However, each of three authentication methods has some issues which make them not reliable separately.

In the knowledge-based method, from an attacker viewpoint, social networks expose sensitive information regarding users which could be used in breaking passwords or security questions. On the other hand, biometric type authentication systems such as face, fingerprint, and speech recognition should become secure against advanced threats like a 3D modelling of a face or finger which were recently used to bypass the restriction of biometric-based systems. At the same time, a backup plan is a vital matter which increases the complexity of an ownership authentication system. For instance, if a user loses his token, an emergency way for him to access his account should be designed. Besides, another challenge which requires attention is the privacy of authentication systems. For instance, authentication systems tend to know more about the user such as phone number of a user which leads to collect more personal information. This also decreases the ability of a user to make a decision in data collection phase of a service. Such authentication systems could facilitate the linkage and profiling of different users across different platforms and services. Lastly, utilizing artificial intelligence and using brain signals to carry out user authentication would seem to have this potential to make good research topics.

REFERENCES

[1] J. Bonneau, C. Herley, P. C. Van Oorschot, and F. Stajano, “The quest to replace passwords: A framework for comparative evaluation of web authentication schemes”, in Security and Privacy (SP), 2012 IEEE Symposium on, IEEE, 2012, pp. 553–567.

[2] Verizon Inc., 2017 data breach investigations report. [Online]. Available: http://www.verizonenterprise.com/verizon-insights-lab/dbir/2017/ (visited on 08/10/2017).

[3] Jakob Lell, Karsten Nohl, Bad USB - on accessories that turn evil, 2014. [Online]. Available: https://srlabs.de/wpcontent/ uploads/2014/07/SRLabs-BadUSB-BlackHatv1. Pdf.

[4] Idrus, S. Z. S., Cherrier, E., Rosenberger, C., & Schwartzmann, J. J. (2013). A Review on Authentication Methods. Australian Journal of Basic and Applied Sciences, 7(5), 95-107.

[5] A. Brostoff, Improving Password System Effectiveness, PhD Dissertation, Department of Computer Science University College London, September 30, 2004.

[6] National Police Agency, Status of Incidents of Illegal Remittance Related to Internet Banking in 2014 (February 2015) (in Japanese),http://www.npa.go.jp/cyber/pdf/H270212 banking.pdf.

[7] Trend Micro Security Blog, Analyzing digital certificate theft attacks targeting corporate net banking, http://blog.trendmicro.co.jp/archives/9417.

[8] Gupta BB, Arachchilage NAG, Psannis KE. "Defending against phishing attacks: taxonomy of methods, current issues and future directions," Telecommun Syst. 2018; 67(2):247–67.

[9] Gupta BB, Tiwari A, Jain AK, Agrawal DP. "Fighting against phishing attacks: state of the art and future challenges". Neural Comput Appl. 2017; 28(12):3629–54.

[10] Xie M, Li Y, Yoshigoe K and Seker R, Bian J. CamAuth: Securing Web Authentication with Camera. In: High Assurance Systems Engineering (HASE), 2015 IEEE 16th International Symposium on; 2015. p. 232–9.

[11] Varshney G, Mishra M, Atrey P. Browshing - a new way of phishing using a malicious browser extension. Presented at the International Conference on Innovations in Power and Advanced Computing Technologies (IPACT 17) (In Press) Vellore, Chennai, India; 2017.

[12] Barth A, Felt AP, Saxena P, Boodman A. Protecting Browsers from Extension Vulnerabilities. In: Proceedings of the Network and Distributed System Security Symposium, NDSS 2010; 2009. p. 1–12.

[13] A. Borissov, "A novel approach for user authentication to industrial components using QR codes", in 39th Annual Computer Software and Applications Conference, IEEE, 2015, pp. 61–66. [Online]. Available: http://dx.doi.org/10.1109/COMPSAC.2015.214 (visited on 06/25/2017).

[14] Cheong, S. N., Ling, H. C., & Teh, P. L. (2014). Secure Encrypted Steganography Graphical Password scheme for Near Field Communication smartphone access control system. Expert Systems with Applications, 41(7), 3561–3568. https://doi.org/10.1016/j.eswa.2013.10.060.

[15] Azimpourkivi, M., Topkara, U., & Carbunar, B. (2017). Camera Based Two Factor Authentications Through Mobile and Wearable Devices. Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies, 1(3), 1–37. https://doi.org/10.1145/3131904.

[16] Crossman, M. A., & Liu, H. (2016). Two-factor authentication through near field communication. 2016 IEEE, Symposium on Technologies for Homeland Security, HST 2016. https://doi.org/10.1109/THS.2016.7568941.

[17] Chakraborty, N., Randhawa, G. S., Das, K., & Mondal, S. (2016). Mob-Secure: A Shoulder Surfing Safe Login Approach Implemented on Mobile Device. Procedia Computer Science, 93(September), 854–861.https://doi.org/10.1016/j.procs.2016.07.256.

[18] Abdurrahman, U. A., Kaiiali, M., & Muhammad, J. (2013). A new mobile-based multi-factor authentication scheme using pre-shared number, GPS location and time stamp. 2013 International Conference on Electronics, Computer and Computation, ICECCO 2013, 293–296. https://doi.org/10.1109/ICECCO.2013.6718286.

[19] Siadati, H., Nguyen, T., Gupta, P., Jacobson, M., & Memon, N. (2017). Mind your SMSes: Mitigating social engineering in second factor authentication. Computers and Security, 65, 14–28. https://doi.org/10.1016/j.cose.2016.09.009.