American Academic Scientific Research Journal for Engineering, Technology, and Sciences
ISSN (Print) 2313-4410, ISSN (Online) 2313-4402
http://asrjetsjournal.org/

E-CHEQUE: Re-Defined Era for Financial Transactions

Sithma Tilakaratne[a*], Pasindu Jayasuriya[b], Shuhaib Riyaj[c], Maleesha Rodrigo[d], Kanishka Yapa[e], Amila Senaratne[f]

[a,b,c,d]*Undergraduate Student, Sri Lanka Institute of Information Technology Malabe Campus, Malabe, 10115 Sri Lanka*

[e]*Msc, Lecturer, Department of Information Security Engineering, Sri Lanka Institute of Information Technology Malabe Campus, Malabe, 10115 Sri Lanka*

[e]*Msc, Industry Engagement Unit, Department of Information Security Engineering, Sri Lanka Institute of Information Technology Malabe Campus, Malabe, 10115 Sri Lanka*

[a]*Email:* IT19060736@my.sliit.lk, [b]*Email: IT19026794@my.sliit.lk,* [c]*Email: IT19366128@my.sliit.lk,* [d]*Email: IT19001180@my.sliit.lk ,* [e]*Email: kanishka.y@sliit.lk,* [f]*Email: amila.n@sliit.lk*

Abstract

Cheques are used to transfer money from one party to another, has the potential to capture a massive amount of financial value but on the other hand is a piece of paper which can be tarnished and torn into pieces and is fragile. The main objective is to create a E-Cheque application, where the mentioned issues will be eradicated by simply digitizing the cheque. Using an E-Cheque would raise a handful of security questions but utilizing the help of four security technologies these problems are minimized. The approach for dynamic password generation is to generate a password which would be resistant to a selected cyber security attack and would be a key-helping hand to remember the password. Secondly, with the use of OTP together with Voice Biometrics, where an OTP would be used as the first level of security and voice biometrics as the second level to increase security. To cover the compliance point of view, a comprehensive compliance policy is created hence applied to the application. Finally, QR Code generation which is generated with a E-cheque details received from user, then encrypted to generate the QR code and transferred through a chat socket where digital signature will be mandatory to transfer the QR based E-cheque, and therefore when all components are paired together creating a world security standard E-Cheque application.

Keywords: E-cheque; money; paper; compliance; public key encryption; QR Code; password; dynamic.

1. Introduction

The proposed application is an electronic checkbook application or E-Checkbook, where the main objective of the application was to digitize the cheque process.

---

* Corresponding author.

A Cheque is basically a piece of paper which has the capability to hold an immense financial value and is one of the most used forms of money transaction in the world. Since this is a piece of paper, cheques are prone to get destroyed by fluids, squished into a ball of paper, flown away by wind, or even torn into small pieces. Due to the hassle when it comes to the protection of cheques, the use of cheques is declining.

The application made consists of to address the aforementioned issues by digitizing a cheque into a QR Code which can be easily shared and is versatile rather than the ancestral piece of paper.

To address the security aspect of this implementation, the authors have compiled 04 security-based components together whereby to mention: -

- Dynamic Password Generation
- OTP Using Voice Biometrics
- Compliance and Quality Assurance with new features.
- E-Cheque Generation.

2. Literature Review

2.1 Dynamic Password Algorithm Generation

Credentials are an essential component of a person's life, particularly when applying for a banking related application. An individual uses an average of 191 passwords. According to statistics, using a password that is not strong enough results in roughly 81 percent of breaches [5].

According to Statistics [6]: -

- 90% of internet users are concerned that their credentials may be compromised.
- 53% of people remember their passwords
- 51% of people use the same passwords for their personal and business accounts.
- About 23 million users still log in using the password "123456."
- Because individuals forget their passwords, 37% of internet users request changes once per month.

Password breaches are mainly caused by the use of simple passwords. As, shown in the statistics above (04th point), 23 million still use a very breakable password. It is also clearly visible that many more other users use easily crack passwords, simply because it is easier to remember. Brute-forcing attacks, Dictionary attacks and Rainbow table attacks are quite common with the use of easily penetrable passwords. The user could also use this password in another platform also since the password was already created resistant to brute-force attacks. Human error could compromise the confidentiality of the password. Random Forest model is used as a very versatile model, which mainly runs on the decision tree model schematics. The process for generation of the model is slow but creates a good result because the algorithm combines the decisions of all decision trees to get

the final result. Therefore, as for the above-mentioned points it is clearly visible that there is an opportunity in the market for a password generator that generates passwords with the capability of generating a strong password, not capable to be susceptible to brute force attacks, dictionary attacks and to create a password that is uniquely relevant to the individual at least up to a certain extent which could help the user to remember the password.

2.2 OTP Using Voice Recognition

Voice biometrics have been researched and implemented by a multitude of research on different devices but voice recognition for OTPs on a banking system has not been implemented. Voice recognition has been implemented in different ways for different applications and purposes. Secure Authentication: Defending Social Networks from Cyber Attacks Using Voice Recognition by L.S.Y. Dehigaspege, U.A.A.S. Hamy, H.A.H. Shehan, S.A. Dissanayake, H.P. Dangalla, W.H.I. Wijewantha and Dhishan Dhammearatchi (L.S.Y. Dehigaspege, 2016)have implemented voice recognition in the way where when the user offers a voice command, the system generates a vocal password for them. The inputted voice will be checked against the database server's stored data, and if the user's voice is legitimate, they will be able to securely login to their accounts. If the inputted voice is a fraud, the server will record it and save it in the database so that the true user of the account may be heard. Furthermore, when it comes to the voice that we supply, it will record both the voice and the noise, allowing users to input any voice command as a vocal password since the system will recognize it based on the user's noise and voice. However, certain efforts to login with a vocal password may fail. To address this, the user is given three chances to offer a voice to the login attempt, following which the system will match the voice with the voice recorded in the database. In addition, if the user needs to reset the voice, the user can login with the supplied voice and change the vocal password by providing the account the new vocal password (L.S.Y. Dehigaspege, 2016). Figure 1 shows a diagram on how the system is designed to work.

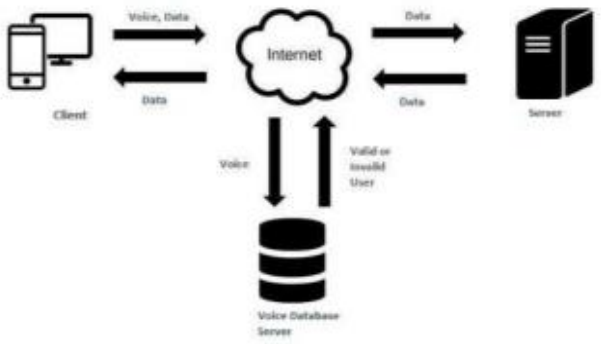

Figure 2.2.1: Voice Biometrics.

Another method of voice recognition explained in Voice Recognition Using MFCC Algorithm by Koustav

Chakraborty, Asmita Talele, Prof. Savitha Upadhya is using MFCC (Mel frequency Cepstral coefficients) algorithm. Which is a method where the system takes voice samples from the user as inputs and processes the samples to get a coefficient which is unique to each user. Figure 2 shows a diagram how the Mel-coefficient is calculated (Ruby Shukla, 2011).

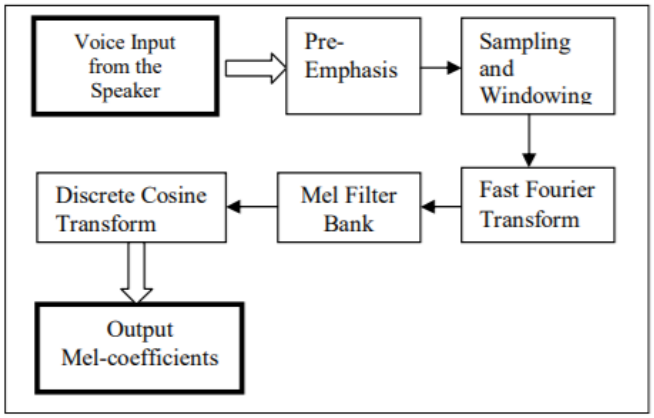

Figure 2.2.2: Mel CoEfficient.

2.3 Compliance with Quality Assurance

Compliance helps you protect your business's resources and reputation. It takes time to build trust with customers, prospects, and vendors, and a big part of that centers on your ethical behavior. Compliance lays the foundation on which you build your company's reputation.

In general, this QR code-based e-cheque mobile application must be more than the current mobile banking applications. The reason is that every component of this application is implemented focusing on the security. For instance, as the authentication of this application voice recognition will be used. In addition to that QR code will be generated instead of cheque and that must be secured while transferring between two parties. As other component, there will be a customized password generator. The specialty of this password generator is the passwords will be generated using the user's data and there will be several passwords. The goal is to generate easy to remember passwords. Since the password contains user data and it is generating several passwords, the user can select a password as his or her will.

2.4 E- Cheque Generation

QR (Quick Response) Codes are the most advantageous pilots to your web-based items and administrations. Present day QR Codes can store a plenty of data that can assist you with drawing in your buyers better A QR code can store and convey information including web interface URLs (UniformAsset Locators), plain text, email addresses, contact data, etc. As it were, a QR Code helps to make your online based presence open to your customer base.[4] It was at first intended for reason for following vehicle parts during make in industry system. Creating a QR code instead of providing a piece of paper for a Cheque transaction is more efficient and secure, and which has not been implemented in the banking Sector. There are plenty of publications related to QR code

generators as “Enhancing QR Code Security” which is written by Shuang Zheng, Linfan Zhang, where it explains the method of using cryptography and cloud technology to prevent unwanted alterations and authentications to increase the QR codes security [3]. The thesis specifically specifies the method of encrypting the data before creating a QR code which prevents the data being transparent for an unauthorized scan. The below figure 3 explains the methodology behind the strategy. Whereas method

(a) is the normal method of creating QR codes and method (b) is the proposed method.

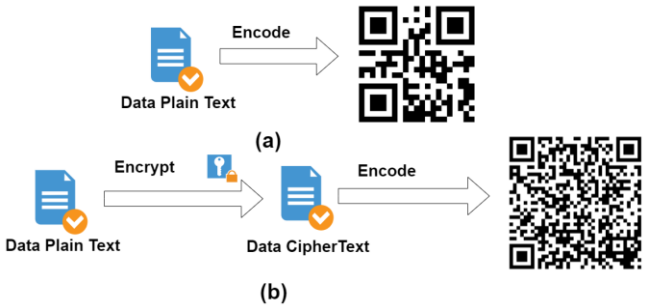

Figure 2.4.1: QR Creation.

Then the research by Katharina Krombholz, Peter Frühwirt, Peter Kieseberg, Ioannis Kapsalis, Markus Huber & Edgar Weippl which is “QR Code Security: A Survey of Attacks and Challenges for Usable Security” where they explain the attackers methods to exploit or take use of the QR code generating algorithm and the strategy which should be used in order to prevent the threats regarding them [(Weippl) (P. Kiran)].

After the Process of creating the QR code, the transaction of the QR code will be done by a verification of the signature where image processing will be used to match the signature of the user or fingerprint biometrics to authenticate the user to transfer the E-Cheque [5]. The following research is mainly concluded for the process of identifying the signature “Offline Signature Recognition Using Image Processing Techniques and Back Propagation Neuron Network System” written by P. Kiran, B. D. Parameshachari, J. Yashwanth & K. N. Bharath. [5] explains the process to be continued.

Then after the signature verification a socket is used to transfer the QR code it was understood using the research BROADCASTING CHAT SERVER by Puli sri Lakshmi, devada pawan, Thota Ganesh which explains Service-oriented architectures are used to create and make available a number of network systems that can connect with one another. The client server architecture is employed in this project to create a chat application.First, a chat application based on Transmission Control Protocol (TCP), a connection-oriented and dependable connection protocol, is developed for both clients and servers.It uses GUI applications and socket programming to implement the "Chat Server" concept. Multiple user connections should be supported by the chat server, and any messages provided to the server are broadcast to every user that is presently logged in, This clarifies the fundamental ideas of threading in network programming.

3. Methodology

The approaches that were used to produce the intended findings have been discussed by the authors in this section. The diagram that follows, provides a general overview of the completed product.

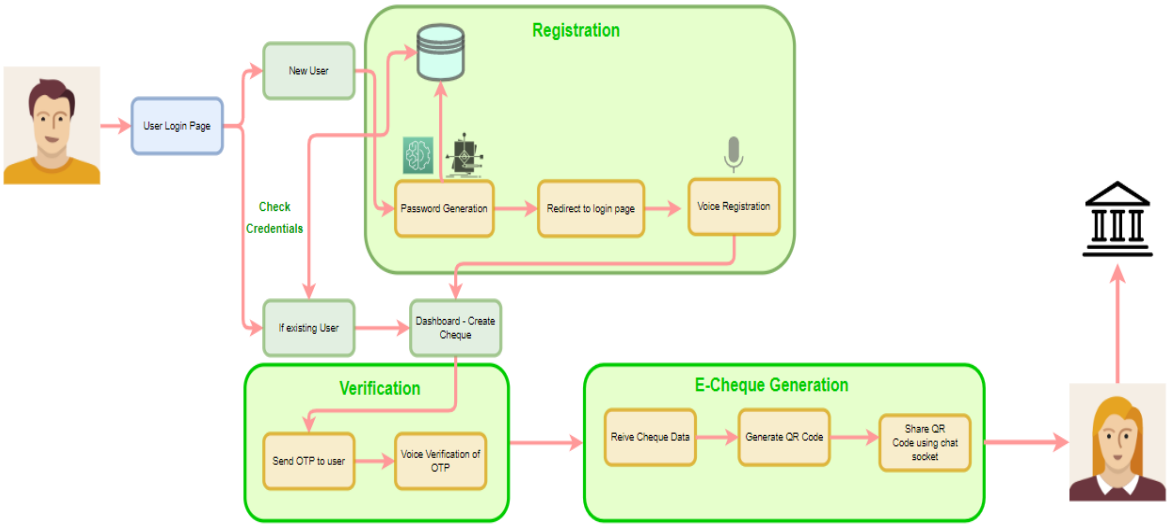

Figure 3.1: Overall System.

3.1 Dynamic Password Algorithm Generation

The component is comprised of a series of activities that, when combined, provide a system for generating passwords. The component itself is divided into two parts namely being: -

- Creation of the password list.
- Strength Checking of the password.

The component's overall overview proceeds as described below.

The password generation component is used by the user at the stage of registering our checkbook application. The user is provided with dummy credentials by the bank which owns the application and the user types in those credentials into the application, therefore elimination of the same user enrolling a number of times when the user forgets the password. After the credentials are typed in the system API handles and processes the request and checks if the provided credentials are present in the database or if the credentials were a randomly generated. Afterwards the application proceeds into the registering phase where the user would be provided with an online which would be needed to fill, this form would only be utilized at the registering phase. All data afterward would be cleared off after the registering process instance is completed. The user form would be comprised of three main components which are: -

- The users' basic details (names maiden name etc.)
- User Unique Data - Comprises of data which are unique to the user. Secret data.
- Characters – Insertion of a limited number of characters (#$%).

The components are designed such that the algorithm can be fed with a variety of data types, including strings, numbers, and characters. The User Unique Data plays a critical function in the overall generator since it is essential to supply the user with user-friendly data that would greatly aid in the user's ability to remember the provided password.

3.1.1. Phase one

The algorithm consists of a series of internal functions that take the user's input data to transform the data and produce a text document.

Figure 3.1.1.1: Arrays.

The user is asked to type in a certain set of questions to which some questions would be asked customized to the users' taste, such as university attended, whether married or if a child is present, and incase depending on the number of children present for the questions to be iterated. The user would again be asked for user unique data which is data which only the user would know or data which the user prefers to be used in the password generation process, typed with spaces in between and no amount is specified, hence is processed as a security measure against brute force attacks. For this category the user can type in any characters ranging all alpha numeric characters and is treated by the algorithm as characters. Next the user is prompted to type in special characters i.e., symbols separated by spaces and is treated as strings by the algorithm. From the algorithm, the data which is entered by the user is appended into three lists namely strings, numbers, and characters.

Firstly, the strings list is processed and sorted out. The collected strings would firstly be transformed into title case using the title() function and then inserted into a new created list namely modified_strings. Then the same strings list would be converted as upper case using the upper() function and then appended into the modified_strings list. As the third step the strings list would again be transformed into lower strings using the lower() function and appended into the modified_strings list.

Secondly, to introduce more ambiguity into the passwords created for a text file by the end of the algorithm, randomly the algorithm chooses certain elements of the modified_strings list and replaces @ for a, 3 for E, 4 for

A, $ for s, 1 for i, 0 for o. Each of those characters replaced are done under separate loops in which the elements are chosen randomly by the algorithm using the random function. Separate lists for each character replacement loop were created and then appended into on new list called replaced. The replaced list is also then extended into the modified_strings list.

To reduce the element of brute force attacks and rainbow attacks, all four lists namely modified_strings, numbers, characters, replaced are randomly shuffled.

Then selecting the modified_strings list, two elements of the array are concatenated together to create a new string, as a single element and thus repeats and loops until the end of modified_strings list. The new strings are appended together in each iteration to a new list named as concat_mod_strings.

The number lists are also converted into a string list for the convenience of appending lists together.

The numbers list is then concatenated by a loop which the iterations are chosen by a random number between 2 and 10. During each concatenation the numbers list is randomly shuffled. Thus, therefore recreation of the same password is very hard. The same is done with the characters list. The characters list would only have a limited number of elements, depending on the users' input. Therefore, in order to extend the number of elements in the characters list, the same set of characters is extended to the original char list and then appended to a new list called more_char after being randomly shuffled through each iteration. All the lists created is then appended together to one final list named as arrays and then randomly shuffled. This list is again joined by two elements each at a time to create length in a password with a minimum of 08 characters and also to minimize wastage of passwords because the passwords would again be sorted out checking the length to minimize brute force attacks. The passwords are then stored in a new list. The password list is then sorted for minimum of 08 characters and then written to another list and then written to a text file which is to be next used is phase two.

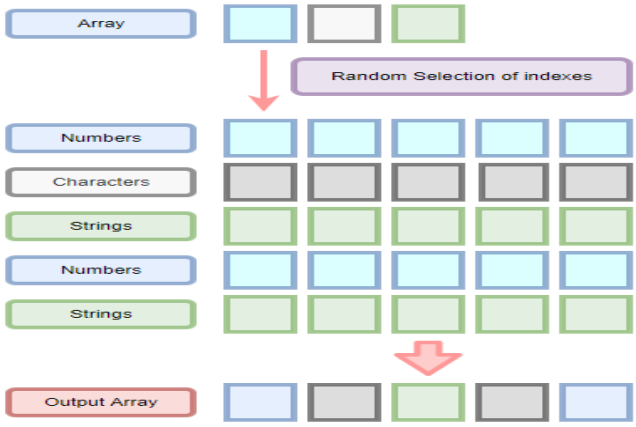

Figure 3.1.1.2: Random Selection.

3.1.2. Phase Two

The phase consists of a machine learning model where the model predicts the strength of the provided password list and outputs the results into a .csv file. Randomly five passwords will be then selected and shown as output to the user.

The model the author used for this application is Random Forest classification algorithm since this model is mostly used for classification and predictive instances. [5]

The basis of the Random Forest classification model is decision trees. Decision tree works on Entropy and Information Gain. Decision tree also provides a high accuracy rate. The equation for entropy E(S) simple saying the measure of disorder, p_i is the frequentist probability for class I [8].

$$E(S) = -\sum_{i=1}^{c} p_i \log_2 p_i \qquad (1)$$

Information Gain is where the difference of two entropy values is measured, calculation of the reduction of uncertainty. Greater reduction of uncertainty = Greater Information Gain.

$$Information\ Gain(X,Y) = E(Y) - E\left(\frac{Y}{X}\right) \qquad (2)$$

The output from the machine learning model is then extracted into a CSV file.

3.1.3. Phase Three

In phase three, utilizing the CSV file, chooses 05 random passwords and then presents the password to the user. This security measure is again done to ensure that no two people would get the same password even if the parameters provided into the first algorithm at phase one contains the same data.

The User then chooses one of the following provided passwords and chooses to register with the said chosen password. The password would then be stored in a database as a new entry with the newly provided username. The stored password would be stored as a 256 Hash [9], to ensure stored data security.

3.2. OTP Using Voice Recognition

After a successfully completed registration process the user will submit a voice clip of them saying all the numbers through 0 and 9. The submitted clip will be broken into 10 clips where each different number is broken into different clips. These 10 clips will be stored as the initial voice print which would be updated and used in the authentication process.

The complete biometrics system will be created using two systems which both use machine learning –

1. Voice authentication algorithm.
2. Algorithm to update the voice print.

3.2.1. Phase One: - Voice Authentication

When the user needs to do a transaction, the system will send an automatically generated 6-digit One-Time Password to the user's mobile phone. The user will then be directed to a page with a record button where the user must record a voice clip of them saying the 6-digit One-Time Password with considerably long pauses between the different numbers. The algorithm will then take the user submitted clip and divide it into 6 different clips of the user saying each number of the OTP individually. The algorithm will the compare each user submitted clip with the specific numbers from the voice print stored in the system and if the authentication is successful the user will be able to continue with his transaction.

3.2.2. Phase Two: - Voice Print Update

When the authentication process is completed and while the user is being directed to continue with the transaction the 6 separated voice clips of the 6-digit OTP the user submitted will be put into the 2nd algorithm. The algorithm will use the initial voice prints and the user submitted clip and by using reinforcement learning the algorithm will update the voice print by combining the user submitted voice clips and the stored voice print. The updated voice print will be stored and used in the next authentication process.

3.3 Compliance with Quality Assurance

Apart from the security of each component, there must be an implemented security for the whole application. Having the security for each component will not be enough because after all the components integrated, there are some other sides to be implemented securely as a whole or completed application. Since this is a mobile application, and the rooting country of this application is Sri Lanka as the compliance standard "guidelines on minimum compliance standard for payment related mobile applications" [3] issued by the central bank of Sri Lanka (CBSL) will be used.

As earlier mentioned apart from the security of each component will be focused on the integrated whole application as follows:

- User accounts should be registered with the mobile number
- A login authentication and financial value-based transaction authentication should be in place for each payment related mobile application user account
- Parallel use of the same account should not be allowed from multiple devices
- Authentication should be processed only at the backend except for the authentication methods based on biometric or chip-based authentication
- Short lived access tokens should be implemented to authenticate client requests without transmitting

user credentials

- Multi factor authentication should be implemented
- A configurable account lockout function should be implemented after invalid login attempts
- Authentication attempts should be logged and monitored to detect login anomalies and possible attacks in real time. All transactions should also be monitored for anomalies. Both types of anomalies should be notified to the user
- Access to any internal resource should be properly authenticated
- Principle of least privilege should always be followed
- Privilege escalation controls and URL manipulation controls should be implemented
- Session ID should be randomized
- Mobile application should have automatic user log off functionality after an idle time period
- During the log off all application specific sensitive data stored in all temporary and permanent memories of the mobile device should be erased
- A procedure should be implemented at the server side to detect and communicate simultaneous login attempts to the user
- A procedure should be implemented to centrally disable the access to the mobile application server from a device reported lost or stolen
- Payment related sensitive data should only be stored in an ecosystem approved and regulated by CBSL
- Sensitive information in temporary storages of the device should be secured appropriately
- Data should be sanitized and validated before recorded in databases. Mobile application databases should be hardened for server side and client side
- Mobile application should use cryptographic algorithms and iteration counts that are currently not identified as vulnerable, industry tested and accepted by institutions like Federal Financial Institutions Examination Council (FFIEC)
- Sensitive data should be encrypted while in transit and at rest. Mobile application should use a salt when generating hashes from passwords
- Encryption keys should not be stored in the mobile without appropriate security controls
- Transport layer encryption shall be implemented for all communication
- Mobile application should use valid SSL certificates issued by a trusted certificated authority
- Certificate pinning shall be properly implemented and used with proper exception handling
- Controls to mitigate bypassing of certificate pinning shall be implemented
- Mobile application shall cease operations until SSL certification errors are properly addressed, and certification errors shall not be ignored
- Sensitive data shall be transmitted only through letters, in-app notifications, or email. Only One Time Password (OTP) shall be transmitted using alternate channels such as USSD, SMS, MMS, or other notification channels
- Mobile application shall not allow any third-party to debug the application during runtime.
- Minification and source code obfuscation techniques shall be used in the payment related mobile application

- Mobile applications shall not be allowed to be executed on rooted/jail broken devices
- Mobile application shall acquire only minimum Operating System permissions required for the application to function properly
- Developers shall adhere to secure coding practices and standards, that are inherent to the coding language used
- Mobile application shall not use vulnerable components, protocols, libraries, scripts etc
- Implementations of components, protocols, libraries, scripts shall not lead to any vulnerability
- Mobile application shall be properly patched if any vulnerability is identified
- Sensitive information such as configuration details shall not be hardcoded in the source code
- All input and output data shall be properly sanitized and validated at the server and at the client
- Auto complete feature shall be disabled for password
- Proper error and exception handling shall be implemented throughout the application
- Sensitive information and or hints shall not be disclosed in error or warning messages and notifications
- Mobile application errors shall be logged in the server
- Servers and web services with which the mobile application communicates shall be properly hardened
- Server access controls and audit logs shall be maintained at the server level
- Ports and services which are not used by the payment related mobile application shall be disabled.
- The logs shall be stored in a log serve which is segregated from the application/database servers and protected with appropriate access controls
- The mobile application crash logs shall not be permanently stored in the mobile device and shall be flushed during log in and or log out processes
- The mobile application logs shall not contain any sensitive data
- Security safeguards shall be implemented to protect the logs from unauthorized modification or destruction and only authorized officers shall be provided with access to the logs
- The mobile application shall be hosted only at the relevant platform and or store (Google Play Store, Apple App Store, Microsoft Store, Galaxy Store, etc.) and shall not be hosted for downloading at PSP website, the vendor website or any other third-party website.

The sensitive data of the e cheque will be included in a QR code. The specifications for the symbology known as the QR Code are outlined in ISO/IEC 18004:2015. The properties of the QR Code symbology, data character encoding techniques, symbol formats, dimensional characteristics, error correction guidelines, reference decoding algorithm, production quality standards, and user-selectable application settings are all listed.

- Encode procedure overview
- Data analysis
- Modes
- Data encoding
- Error correction
- Constructing the final message
- Codeword placement in matrix

- Data masking
- Format information
- Version information

3.4. E - Cheque Generation

Electronic cheques can be utilized to make an installment for any exchange that a paper cheque can cover and are represented by the very regulations that apply to paper cheques. The process followed to create a QR E-Cheque has mainly 4 steps. They will be

1. Entering the details related to the Cheque
2. Generating the QR code by encrypting the details
3. Signature recognition to transfer the QR E-Cheque
4. Chat socket to transfer the E-Cheque Safely

3.4.1. Entering the details to the cheque

- The user will have to provide the details related to a cheque transaction as whereas the details already printed in the cheque will be automatically sent to the QR code generator, which are Date, Pay[receiver], amount, amount in words are type of examples which the user will have to enter, while the serial no, bank code, account no will be automatically processed by the back end.

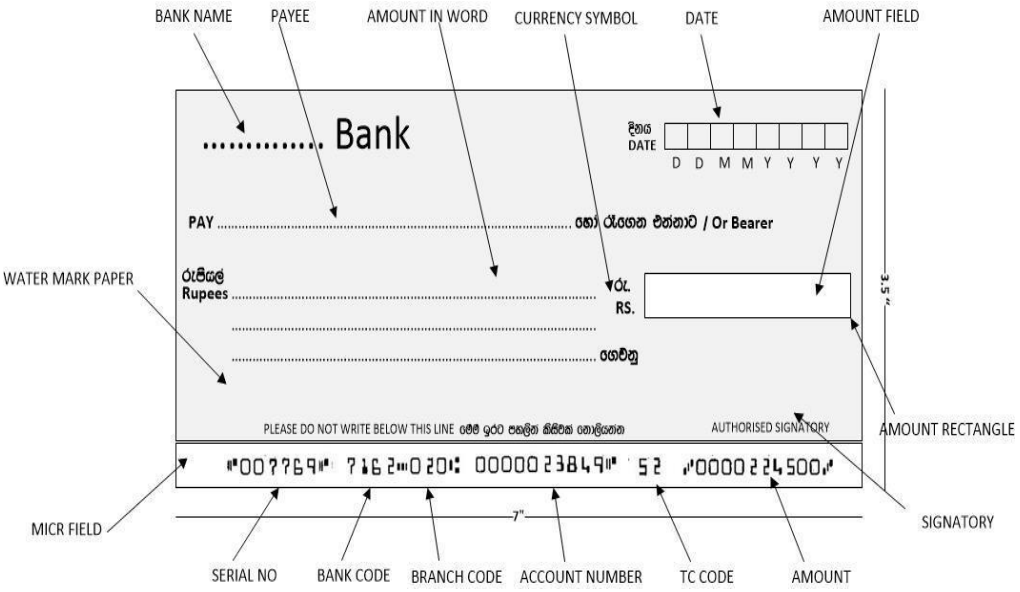

Figure 3.4.1: Cheque Details.

3.4.2. Generating the QR code by encrypting the details

- The data received related to cheque and the data being sent through the back end which are the details already printed on a cheque will be encrypted using an encryption key and finally the generated encrypted data will be used Generate the QR code.[5]

3.4.3. Signature recognition to transfer the E-Cheque

- The produced QR code cannot be shared to the particular person until the E-Cheque provider, has been verified through the process of matching his digital signature where the neural networks algorithm will be used with the support of Image processing technology for the verification process.[4]

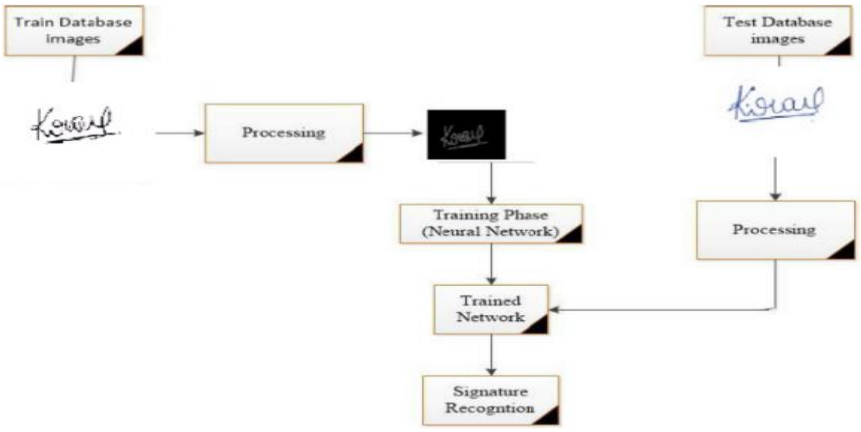

Figure 3.4.3: Image Processing Method.

3.4.4. Chat Socket to transfer the E-Cheque

- Here a Chat socket will be created between the users by the support of primary and secondary key to create an end-to-end encrypted chat socket, whereas the users can have a private conversation to transfer the E-Cheque generated QR code which can be later used in the bank to perform the money transaction.[3].

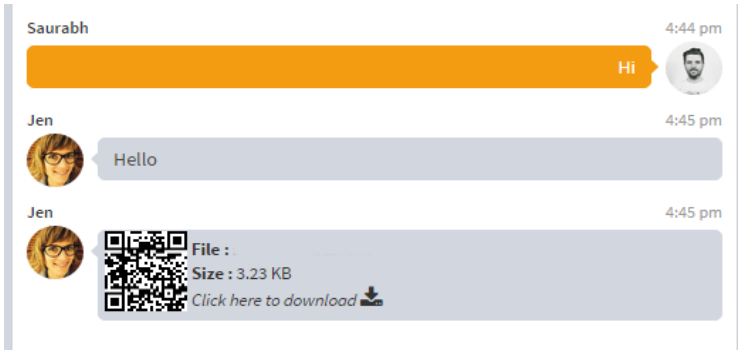

Figure 3.4.4: UI of Chat Socket.

The following process will be more efficient than the manual process followed by the providers and receivers of the Cheque transactions.

4. Results and Discussions

4.1Dynamic Password Algorithm Generation

According to the research conducted, with the algorithm used in phase 01, the ability to brute-force is eradicated, for the technological capabilities existing at the time this research paper was written. Also using the Random Forest machine learning model an accuracy of 98% was achieved as shown in the diagram below.

```
In [20]: from sklearn.metrics import classification_report
         print(classification_report(y_test, y_pred))
```

	precision	recall	f1-score	support
0	0.99	0.93	0.96	18150
1	0.98	1.00	0.99	99050
2	0.99	0.97	0.98	16721
accuracy			0.98	133921
macro avg	0.99	0.96	0.97	133921
weighted avg	0.98	0.98	0.98	133921

Figure 4.1.1: Accuracy Matrix.

More advancements can be done to this component by increasing the efficiency levels of accuracy or by increasing the user friendliness of the passwords being generated.

Limitations for this algorithm is one-fold. The user needs to provide enough data, especially for the fields of user unique data input and numbers field, such as lucky numbers, during the password generation process to create a strong yet customized password. Thus, the randomizing functions in the generator have more and sufficient data to work with and provides a password with the most customized way.

4.2 OTP Using Voice Recognition

The research conducted confirmed that using voice recognition and voice submission of OTPs increases the security level of any application massively due to the added layer of biometric security. Algorithm one which does the voice biometrics section of the project returns an accuracy of 94%. While algorithm 2 makes sure the users voice print is updated in a manner which it would not affect the aforementioned accuracy of algorithm 1. The voice recognition and updating component of the system ran into a few limitations which were mostly limiting the process of updating the process. Due to the fact that the voice updating process is done using machine learning there needed to be a clear and big data set of users submitted voice clips of them saying numbers of zero through nine, there was only one particular data set of the sort, therefore with additional training data the algorithm can be improved massively. While algorithm 1 can also be improved by the addition of machine learning technology for the core aspects of the comparison.

4.3Compliance and Quality Assurance with New Features

In conclusion a check list which is created according to the guidelines provided by the Central Bank of Sri Lanka (CBSL) will be there when the mobile application development. By ensuring all the steps in the checklist is implemented, there security side of the mobile application is implemented. It is more important to conduct the compliance risk assessment at least once a year. By conducting compliance risk assessments, the new vulnerabilities and the potential risks can be identified. Once identify those the mitigation steps can be implemented and securing the application from cyber incidents.

4.4 E-Cheque Generation

As per the researchers conducted, we can identify that the usage of converting the QR code and using the QR code for cheque transactions will provide high security and less amount of manual workload, as per the confirmation of the E-Cheque owner the Neural networks algorithm should be used for the image processing of the customer signature. Which will provide an 85% accuracy while the transaction of the E-Cheque will be done through a chat socket using the E-Cheque application. Then the socket transferring will be done in order to transfer the E-Cheque. Which gave clear results where the transaction of the QR code was complete.

5. Conclusions and Future Advancement

In conclusion, the solution covers an entire application where both the security aspects of Cyber security and compliance policy are integrated together. This application holds a potential value to uplift the use of cheques and use as a main formal form of financial transactions.

For future developments this application could be further modified positively by updating the used technologies to fit with the security boundaries being used then. This application can also be boosted by changing the compliance policy to ensure that the users adhere to up-to-date policies. Furthermore, the algorithms used by this application can be updated using improved algorithms to enhance the accuracy of the application.

References

[1] U. H. H. S. S. D. H. D. W. W. D. D. L.S.Y. Dehigaspege, "Secure Authentication: Defending Social Networks from Cyber Attacks Using Voice Recognition," vol. 06, no. 10, pp. 120-124, 2016.

[2] P. S. Ruby Shukla, "E-Banking : Problems and Prospects," E-Banking : Problems and Prospects, vol. 01, no. 01, pp. 23-25, 2011.

[3] M. H. &. E. Weippl, "QR Code Security: A Survey of Attacks and Challenges for Usable Security," [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-319-07620-1_8.

[4] B. D. P. J. Y. &. K. N. B. P. Kiran, "Offline Signature Recognition Using Image Processing Techniques and Back Propagation Neuron Network System," [Online]. Available: https://link.springer.com/article/10.1007/s42979-021-00591-

y#:~:text=Preprocessing%20of%20signature%20can%20be,to%20reduce%20the%20processing%20time..

[5] G. Peru, "Password Statistics in 2022 – How to Save & Protect Your Data," 13 March 2022. [Online]. Available: https://webhostingprof.com/blog/password-statistics/. [Accessed 30 June 2022].

[6] "Average Business User Has 191 Passwords," 6 November 2017. [Online]. Available: https://www.securitymagazine.com/articles/88475-average-business-user-has-191-passwords.

[7] IBM, "Decision Trees," [Online]. Available: https://www.ibm.com/topics/decision-trees. [Accessed 30 June 2022].

[8] J. M. Lowery, "MD5 vs SHA-1 vs SHA-2 - Which is the Most Secure Encryption Hash and How to Check Them," 26 March 2020. [Online]. Available: https://www.freecodecamp.org/news/md5-vs-sha-1-vs-sha-2-which-is-the-most-secure-encryption-hash-and-how-to-check-them/#:~:text=Probably%20the%20one%20most%20commonly%20used%20is%20SHA-256%2C,considerably%20more%20secure%20than%20either%20MD5%20or%20SHA-1. [Accessed 30 June 2022].

[9] S. T, "Entropy: How Decision Trees Make Decisions," 11 January 2019. [Online]. Available: https://towardsdatascience.com/entropy-how-decision-trees-make-decisions-2946b9c18c8. [Accessed 30 June 2022].

[10] CBSL, "Central Bank Of Sri Lanka," 01 2020. [Online]. Available: https://www.cbsl.gov.lk/sites/default/files/cbslweb_documents/laws/cdg/psd_guideline_no_1_of_2020_e.pdf. [Accessed 28 06 2022].

[11] J.-P. C. Hao Chen, "Security Testing and Compliance for Online," International MultiConference of Engineers and Computer Scientists, vol. 01, no. 01, 2009.