**QRCODE TOKEN: AN IMPROVENT TO ONE TIME PASSWORDS IN TOKEN**

**Introduction**

Token-based authentication is a mechanism that allows users to prove their identity and obtain a unique access one time password(OTP) in exchange. During the OTP’s lifetime, users may access the website or app for which the token was issued, rather than having to re-enter credentials every time they return to the same webpage, app, or other resource secured by that same token. Authentication tokens function similarly to a stamped ticket. As long as the token is valid, the user has access. The OTP is invalidated when the user logs out or exits an app. Token-based authentication differs from standard password- or server-based authentication methods. Tokens provide an additional degree of protection, and administrators have complete control over every activity and transaction (Iam, 2021).

This research, on the other hand, tries to improve and validate the existing hardware token-based authentication. The suggested model is a hardware authentication that acts as an extra layer of security by scanning the user's finger to authenticate user identity before creating the QR-Code necessary for the transaction. Unlike conventional token-based authentication, which generates a set of numbers that are vulnerable to attacks such as shoulder surfing/theft and brute-force, particularly if the numbers are short, this suggested architecture incorporates these types of assaults into the design model (Monica, 2022).

**Statement of Purpose**

Forceful Number Sharing: Giving out phone numbers for logins isn’t considered safe as phone numbers are linked with bank accounts. Another added issue that comes along with sharing phone numbers is the rise in unwanted SMS and spam calls.

Inconvenience: OTPs can also be considered inconvenient as users have to copy the OTP from the token they receive it in, leading to a change in UI and increasing human error in inputting the numbers (Hypersign, 2021).

Impersonation: OPT tokens, most especially ones used by banks in Nigeria does not offer user authentication before producing the OTP. Hence the legitimate user of the token, can be in any part of the world and allow a third party operate the token on their behalf. If this device gets into the wrong hands unfortunately, transactions can be intercepted and the user’s fund can be exploited (Aviad, 2021).

**Aim and Objective**

The aim of this project is to create a Qrcode token as an improvement and a replacement to the traditional token that Generates a one time Qrcode.

The objectives of this project are;

1. To provide a 2-factor authentication as a means of transaction initiator.
2. To bind the owner of the token to its operation to avoid impersonation
3. To eliminate human error produced by copying OTP into transaction App.

Advanced Online Banking Authentication System Using One Time Passwords Embedded in Q-R Code This paper explains implementation details of on-line banking authentication system. Security is associate vital issue for on-line banking application which might be enforced by varied Mobile technologies. Whereas implementing Mobile banking system, secure information transfer want is consummated by exploitation https information transfer and info encryption techniques for secure storage of sensitive info. To eliminate threat of Network interception and to substantiate user identity we have a tendency to use conception of OTP encrypted and embedded into QR-code. QR-code displayed by a hardware token which might be scanned by user mobile device to overcome the weakness of ancient countersign based mostly system. We have a tendency to improve more security by exploitation only once countersign (OTP) that hides within QR- code.

The QR-code Token as a Two-Factor Authentication Using Mobile Banking App provides improved protection, since users are prompted to supply one thing they understand and one thing they need. This methodology delivers a higher-level of authentication assurance, that is important for Mobile banking security.

**PROPOSED SYSTEM**

Disconnected tokens have neither a physical nor logical connection to the client computer. They typically do not require a special input device, and instead use a built-in screen to display the generated authentication data, which the user enters manually themselves via a keyboard or keypad. Disconnected tokens are the most common type of security token used (usually in combination with a password) in two-factor authentication for online identification. One Time Password is a 1D password system which is validate only one time for a valid user within a specific time. Hence each time user will be authenticated with a new passwords .It helps in preventing various types of attack like. replay attack , phishing attack and many more which is using static passwords. It also offers other characteristics like anonymity, portability, extensibility and enable to keep information safe or from being leak [2].There are two approaches for generating an otps: 1. Time based OTP- In these OTP changes at frequent interval of time.

2. Event based OTP- In these OTP will be generated by pressing a button on the OTP device or token.

On the other hand, QR –code is a Quick Response 2D barcode which is used to store information into an image form. It provides more security, more storage for storing information. These are proven by ISO standards that contain information in both horizontal and vertical directions, whereas 1-D barcode contains information only in one direction either horizontally or vertically. It also provides error correction capability. Data can be recovered easily even the some part of the image is damage or distorted. There will be no loss of data. Users can decide what action should be taken or what information are needed to store inside a QR code. QR-CODE can store 7089 characters as compare to bar code which store only 20 digits. We can store URL, text, images , geolocations, and other forms of data. That is why the QR-code has been choosen to contain the OTP with end to end Encrytion between the bank server and the token. Qr-code Only displays only after the owner of the token’s finger print has been authenticated through a finger print scanner on the proposed Qr-token design.

The generated OTP is encrypted using the strong AES algorithm. The generated OTP value is Encrypted with a strong AES algorithm and sent to the user. AES is an iterative asymmetric key block cipher, It uses three key strengths of 128, 192, and 256 bits. AES uses 128 bits as the block for encryption and decryption.

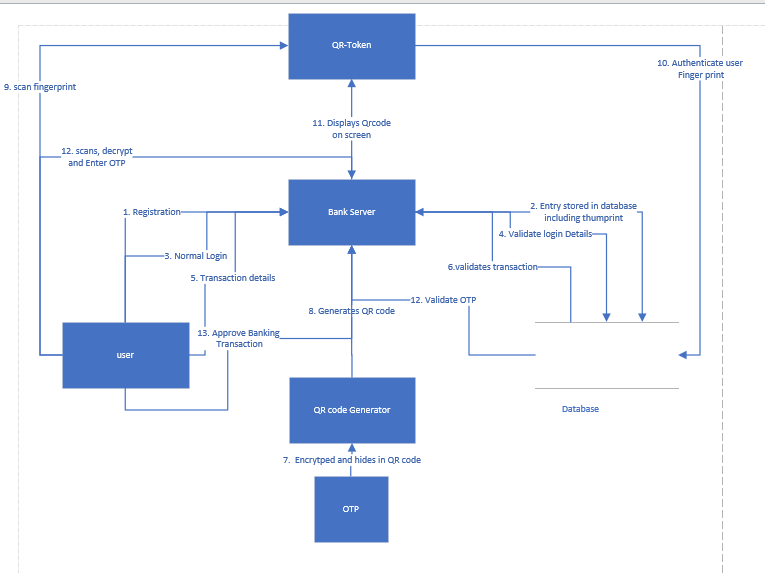
This is one of the perfect encryption algorithms to protect your personal data. The Encrypt AES tool transforms Plaintext to ciphertext in a series of iterations based on the encryption key. The AES decryption method uses the same process of converting ciphertext back to plaintext using the same encryption key. it's so hard to defeat with a brute force attack.

The encrypted OTP is generated when the token is switched on after transaction details have confirmed correct.

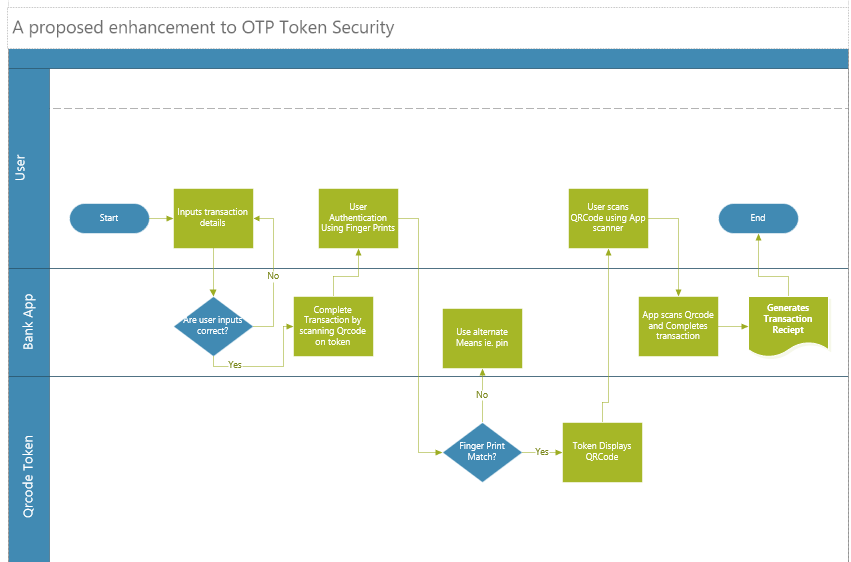
The user’s pre-registered fingerprint will be authenticated by scanning the finger print on the token. The purpose of this step is to avoid impersonation and also, trigger the display of the Qr-code which contains the encrypted OTP. The purpose of using a Qr-code displayed on the screen the token is to make eliminate the stress of copying OTP from the token to the Mobile app. With the QR-code, the user only needs to scan from their mobile app. Doing this will increase the convenience in using hardware tokens and eliminate human error in trying to input OTP manually. Thereby, it lessens the chances of the Qr-code being re-generated. Also, shoulder surfing will be undoable because there is no display of plain text and only QR-code scanner on the mobile app is wired to read and decrypt cypher text (OTP)

The QRcode scanner on the mobile App then scans the qr-code, decrypt the cypher text, inserts the plain text (OTP) and approves the process of transaction tranfer based on the information provided by the user initially

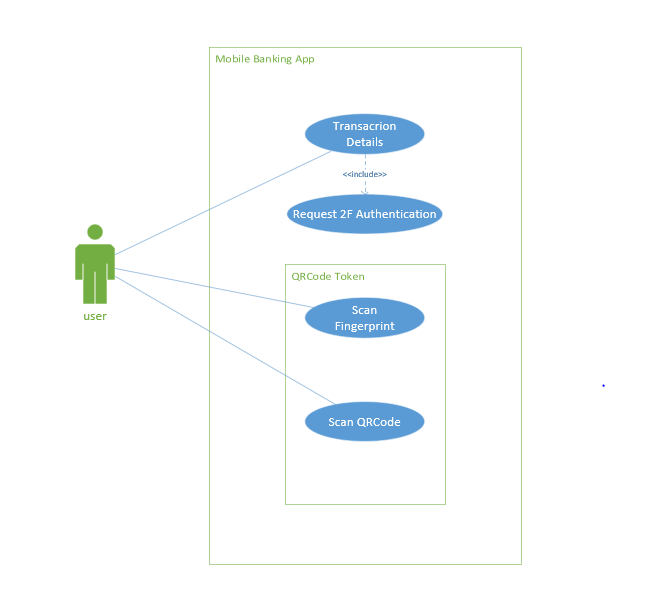
IMPLEMENTATION



Architecture Design



Process Flow Diagram



Uses case diagram

|  |  |
| --- | --- |
| USE CASE ID | CASE-001 |
| USE CASE NAME | Transaction Details |
| DESCRIPTION | Use case allows user to input details needed for a Debit transaction i.e. receivers account number, receivers Bank name and Amount to transfer to receiver. |
| TRIGGER | Login |
| PRIMARY ACTOR | User |
| SECONDARY ACTOR | None |
| PRE-CONDITION | CASE-001 |
| POST-CONDITION | System checks details inserted by user and verifies that it matches a legitimate user of a bank selected and request transaction approval. |

|  |  |
| --- | --- |
| USE CASE ID | CASE-002 |
| USE CASE NAME | Request 2F Authentication |
| DESCRIPTION | Use case redirects user to a hardware token to reverify authentication as a second layer of security |
| TRIGGER | Verified transaction details |
| PRIMARY ACTOR | User |
| SECONDARY ACTOR | None |
| PRE-CONDITION | CASE-001 |
| POST-CONDITION | QR code token verifies User |

|  |  |
| --- | --- |
| USE CASE ID | CASE-003 |
| USE CASE NAME | Scan Fingerprint |
| DESCRIPTION | QRcode scanner needs to authenticate user to make sure the user that initiated the transaction is the same user trying to approve it. |
| TRIGGER | 2f authentication Request |
| PRIMARY ACTOR | User |
| SECONDARY ACTOR | None |
| PRE-CONDITION | CASE-002 |
| POST-CONDITION | System displays QR code |

|  |  |
| --- | --- |
| USE CASE ID | CASE-004 |
| USE CASE NAME | Scan QR code |
| DESCRIPTION | Token displays a Qrcode and user uses mobile App to scan the displayed QR code. The purpose of this, is to identify user and approve transactions on mobile App. |
| TRIGGER | Verified Finger print |
| PRIMARY ACTOR | User |
| SECONDARY ACTOR | None |
| PRE-CONDITION | CASE-003 |
| POST-CONDITION | System Approves transaction and makes Appropriate Debit |

Requirement specification

|  |  |
| --- | --- |
| Req-001 | System shall be verify user thumbprint |
| Req-002 | System shall produce unique qrcodes each time a transaction is triigered |
| Req-003 | System shall allow user scan qrcode |
| Req-004 | System shall approve transaction after qrcode has been scanned |
|  |  |
|  |  |
|  |  |
|  |  |