Quantum-Assisted Framework for Secure File Transfer using MFA

Dharshini K
School of Artificial Intelligence
Amrita Vishwa Vidyapeetham
Coimbatore, India
cb.ai.u4aim24111@cb.students.amrita.edu

Esha R
School of Artificial Intelligence
Amrita Vishwa Vidyapeetham
Coimbatore, India
cb.ai.u4aim24112@cb.students.amrita.edu

Harsshitha S
School of Artificial Intelligence
Amrita Vishwa Vidyapeetham
Coimbatore, India
cb.ai.u4aim24115@cb.students.amrita.edu

Vaishnavi P
School of Artificial Intelligence
Amrita Vishwa Vidyapeetham
Coimbatore, India
cb.ai.u4aim24149@cb.students.amrita.edu

Abstract -- Cryptography was introduced to prevent a third party from accessing and learning the contents of private messages sent during a communication process. Quantum Cryptography looks promising to provide a new level of secure communication by applying quantum mechanics concepts to cryptography.

This project implements a hybrid quantum-classical cryptographic system for secure naval communications. It combines AES-256 encryption, BB84 quantum key exchange (simulated), Quantum Random Number Generation (QRNG), and Multi-Factor Authentication (MFA) to create a post-quantum resistant file transfer system. The system ensures confidentiality through quantum-enhanced key exchange and provides audit trails through comprehensive logging.

Keywords—BB84, Multi-Factor Authentication, Quantum Key Distribution, AES-256

1. INTRODUCTION

The development of quantum computers poses an existential threat to modern secure communications due to their ability to process data exponentially quicker than conventional computers. This new threat is the largest threat posed to modern communications since the invention of the first computers during the Second World War [2]. Quantum computing also offers a solution. Through the secure generation of truly random one-time pads achieved by Quantum Key Distribution (QKD) methods, such as BB84, military communications can be future-proofed [2].

With the advent of quantum computing, classical cryptographic protocols face unprecedented challenges due to the potential of quantum computers to break widely used encryption schemes. Hybrid quantum-classical cryptographic protocols present a promising approach to mitigating these challenges by leveraging the strengths of both quantum and classical cryptographic techniques [9]. In this project, we have created two military channels: the sender end (Admin) and the receiver end (User). At the receiver end, user authentication is verified by "username," "password," and "MFA token"; the TOTP (Time-based One-Time Password) generated by the authenticator app serves as an MFA (Multi-Factor Authentication) token. Password verification is securely verified by a quantum-resistant algorithm - Secure Hash Algorithm (SHA-256) - using PBKDF2, which compares the user-given password with the stored password. The hash algorithm PBKDF2-SHA2 (Password-Based Key Derivation Function 2, Secure Hash Algorithm 2), presenting an optimized brute force cracking methodology through high-speed implementation techniques based on CPU (Central Processing Unit)/GPU, and to establish performance benchmarks for it [8]. In the receiver or admin end, the channel displays file transfer records, data retrieval, and decryption processes. The secret code generated in this channel serves as a passkey to the user channel for viewing the encrypted file. File exchange is securely transferred from one channel to the other using a classical encryption algorithm, AES-256, which works by symmetric key exchange method.

The AES-256 key algorithm is resistant to many classical attacking algorithms, such as brute-force, which helps to encrypt the file. The file is transferred securely using BB84, a quantum algorithm that secures the algorithm by preventing quantum attacks through basis selection and comparison. In our extensive simulation study, the grouped BB84 protocol with 300 qubits comparison guarantees at least 99.92% accuracy in eavesdropping detection under rapidly varying quantum channel conditions [7].

Quantum technologies for military applications introduce new capabilities, improving effectiveness and increasing precision, thus leading to "quantum warfare," wherein new military strategies, doctrines, policies, and ethics should be established [10]. Quantum-enhanced communication channels provide the military with safe and secure data transformation by preventing cyber-quantum attacks. In this project, we have built a sender channel (Admin) and receiver channel (User) where the file is transferred from admin to user with quantum-enhanced algorithms, ensuring security and protection from cyber-attacks.

2. LITERATURE REVIEW:

2.1 Quantum Key Distribution for Secure Encryption in Underwater Networks, 2024 [10]

This project addresses security challenges in underwater acoustic networks (UANs) by integrating Quantum Key Distribution (QKD) with classical cryptographic methods to enable secure naval communications. This system employs the 3-state BB84 protocol for quantum key exchange, after validating through field tests in the Bacchiglione River, where it achieved a 2.2 Mb key generation rate in 30 minutes with low Quantum Bit Error Rates (QBER: ~2.3%). The architecture has two types of stations: surface and underwater. Surface stations provide data transmission and acoustic modems deliver cryptographic information. One-Time pad (OTP) encryption is used for confidentiality, while the integrity of the message is preserved with the addition of cryptograms. The framework incorporates a hybrid network protocol stack (DESERT Underwater Framework) for managing acoustic communication layers and synchronizing quantum key updates via autonomous surface vessels. This work is set up for post-quantum secured underwater networks that link up quantum key distribution and practical maritime communications restrictions.

2.2. An Effective Data Security Mechanism for Secured Data Communications Using Hybrid Cryptographic Technique and Quantum Key Distribution, 2024 [12]

This project utilizes a hybrid cryptographic technique for data security with AES and RSA encryption along with Quantum Key Distribution (QKD). The method begins with an admin generating quantum states. Qubits are sent to authorized users via a quantum channel. AES is utilized for effective symmetric encryption, whereas RSA protects the AES key in the ciphertext, and counteracts cloud environment vulnerabilities. Designed and tested with Python, the mechanism under consideration attained a security level over 99.9%. This is due to the synergy between hybrid cryptography and QKD. Experiments proved that AES encryption had quick processing times, and QKD ensure interaction between legitimate users by verifying distributed keys through photon polarization. This hybrid system increases data confidentiality and reliability, making a foundation for future improvement, including encryption algorithms together with quantum cryptography.

2.3. Quantum Encryption in Military Communication, 2023 [13]

This study advocates for the use of post-quantum encryption in military communications to mitigate the threat posed by quantum computers, using historical examples such as Project Ultra and Room 40 to highlight the significance of secure communication. It looks at current cryptographic systems, quantum risks, and industry remedies, such as AWS's use of hybrid encryption. The paper proposes quantum cryptography as a solution and covers quantum key distribution (QKD) approaches such as BB84, their limits, and the engineering issues associated with maritime applications. While advocating for traditional post-quantum algorithms for maritime units due to the fragility and high cost of quantum technology, it also suggests quantum-ready networks connecting shore bases to improve decentralization and redundancy.

2.4. Implementation of Secure Key Distribution Based on Quantum Cryptography, 2009 [14]

This work studies into the use of quantum cryptography for safe key distribution, with a focus on the BB84 protocol as a solution to the limitations of traditional cryptographic systems like RSA. The study contains Java simulations that compare RSA and BB84 under various scenarios. Results show that RSA is vulnerable to computational attacks, notably from quantum computers, whereas BB84 successfully blocks unauthorized access to shared keys. The protocol has two phases: quantum transmission and public discussion, during which bases are compared to yield a raw key. Despite its efficiency, the study points out drawbacks such as the lack of authentication processes and dependency on certain materials, recommending future research to fill these gaps and improve the practical applications of quantum cryptography in secure communications.

2.5. Hybrid Quantum-Classical Cryptographic Protocols: Enhancing Security in the Era of Quantum Supremacy, 2025 [15]

In this work we present the design and implementation of hybrid quantum-classical cryptographic protocols that can contribute to strengthening security in the period of quantum supremacy. This research tackles the weaknesses of existing classical cryptography systems including RSA and ECC against quantum attack by suggesting that hybrid solutions be formed out of combining Quantum Key Distribution (QKD) with classical encryption schemes such as AES-256. These protocols have dual-layered security, relying on quantum mechanics for safe key exchange but the practicality and efficiency of traditional encryption for data integrity. High-speed fiber-optic networks and special quantum hardware were used to experiment on three configurations: classical encryption (RSA-4096 and AES-256), strictly quantum encryption (QKD with one-time pad), and hybrid encryption (QKD with AES-256). Results showed that hybrid protocols balance security and performance, providing improved quantum attack resistance against classical approaches while minimizing computational overhead and network latency of purely quantum systems.

2.6. Enhanced BB84 quantum cryptography protocol for secure communication in wireless body sensor networks for medical applications, 2021 [16]

This paper presents the Enhanced BB84 Quantum Cryptography Protocol (EBB84QCP) for mitigating the risks in Wired Body Sensor Networks (WBSN) and specially protecting sensitive medical information over wireless transmissions. Traditional encryption methods such as DES and RC4 are susceptible to eavesdropping and man-in-the-middle attack due to the insecure key distribution methods. In contrast, EBB84QCP applies bitwise operations together with quantum mechanics to create secure keys capable of being generated without direct sharing, significantly reducing interception vulnerability. Results from experiments conducted through simulations in Java showed that EBB84QCP surpasses traditional algorithms in key generation rates without increasing susceptibility to wormhole, quantum, spoofing, blackhole and DoS attacks.

2.7. Experimental underwater quantum key distribution, 2021 [17]

This work examines the feasibility and performance of underwater Quantum Key Distribution (QKD) using the BB84 protocol and decoy-state methods for secure sea communication. Experimental demonstration was conducted in a 10-meter water tank with an attenuation coefficient of 0.08/m to simulate Jerlov Type II seawater conditions. The experiments demonstrated that underwater QKD maintained a security key rate of 563.41 kbits/s with a Quantum Bit Error Rate (QBER) of 0.36%, while decoy-state QKD improved the security key rate to 711.29 kbits/s and a QBER of 0.95%. These results show the potential of underwater QKD for applications such as secure naval communications and oceanic exploration, circumventing problems of attenuation and environmental noise while being demonstrated to scale with decoy-state methods for enhancing security and distance of transmission.

2.8. Quantum Key Distribution: Modelling and Simulation through BB84 Protocol Using Python3, 2022 [18]

This paper presents a Python3 implementation of the BB84 Quantum Key Distribution (QKD) protocol for secure communications in IoT environments. This research presents growing threats from quantum computing onto traditional cryptography and the potential applicability of QKD for further securing communications based on quantum mechanics principles. The result demonstrates that an eavesdropper is detectable: the interference caused by the quantum measurement coming into play, as per Heisenberg's uncertainty, and allegedly inapplicable to cloning. The expected theoretical values show that there is a 50% key bit discard rate and a 0.11% error threshold. The simulation has established the applicability of QKD for communication security and leaves avenues for further work such as its power consumption analysis and deployment in a resource constraint device.

2.9. A Review of Security Evaluation of Practical Quantum Key Distribution System, 2022 [19]

This article discusses the insecurity and security issues in Quantum Key Distribution (QKD) systems due to the imperfections of practical devices. It stresses that as QKD provides information-theoretic security, actual implementations of QKD diverge from theoretical models and hence become vulnerable to quantum hacking techniques like photon-number-splitting attacks, Trojan horse attacks, and side-channel attacks. The research focuses on the security requirements concerning different modules of QKD, including the source, encoder, quantum channel, decoder, and detector, with regard to stable photon-number distribution, random encoding and secure detection mechanisms. Measurement-Device-Independent QKD and Device-Independent QKD are techniques proposed to deal with various threats.

2.10. Quantum Cryptography: A Pathway to Secure Communication, 2022 [20]

This paper discusses the progress in quantum cryptography as a countermeasure to the dangers that quantum computing poses to conventional cryptographic systems. The research focuses on quantum key distribution and is primarily interested in the BB84 protocol that uses polarized photon states for encoding and decoding messages and for eavesdropper detection. In addition, the paper outlines post-quantum cryptography algorithms such as lattice-based, multivariate, and hash-based cryptography that are set to resist quantum computer attacks. The findings highlight the imperative nature of QKD in the development of secure communication channels and imply that quantum encryption developments in the future will improve data protection across applications including banking and secure communications.

2.11. Advancements in Secure Quantum Communication and Robust Key Distribution Techniques for Cybersecurity Applications, 2025 [21]

This paper offers detailed treatment of the evolution of quantum secure communication and key distribution techniques with a particular emphasis on quantum communication in security. It outlines the salient aspects of quantum communication in the form of quantum states and QKD protocols such as BB84 and E91, and challenges such as quantum state loss and noise. The piece also describes the applications of quantum communication in future-proof secure settings, including secure multi-party computations and IoT. It discusses current trends in quantum computers, QKD networks, and cybersecurity, including a variety of homomorphic encryption approaches, secure key management for IoT, and CNN199 intrusion detection systems. These findings show that in spite of threats, such as the sensitivity of quantum states, continued effort in quantum error correction, quantum repeaters, and tried-and-true cryptographic algorithms is what quantum computing solely requires towards its application and secure implementation.

3. PROPOSED WORK

3.1 User Authentication and MFA Enrollment

The user authentication and MFA enrollment initializes with static credential verification where users log in via React.js using a username and password. These passwords are hashed using PBKDF2 with SHA-256 and are compared with stored hash in PostgreSQL user table. For added security, Multi-Factor Authentication (MFA) is incorporated using Time-Based One-Time Passwords (TOTP). During MFA enrollment for new users, a unique secret key is generated and is shown as QR code, which users scan with an authenticator app like Google Authenticator, which are validated using HMAC-SHA1 within ±2 step (±30 seconds) within time window.

3.2 File Transfer (Drop Off)

File Transfer allows users (Admin or standard user) to securely transfer files or messages to authorized recipients. It begins when a user provides three essential inputs: the recipient's username, a secret code and the file or message to be securely transferred. This initiates QRNG, encryption and logging.

3.2.1 Quantum Random Number Generation (QRNG)

To make encryption secure, IBM's Qiskit is used to generate a 256-bit quantum random key. Quantum circuit is built where Hadamard gates are applied to create a state of superposition, simulating the randomness of real quantum particles [4]. After measuring the qubits, the outcome is turned into a binary string and then converted into a secure key in Base64 or hex format. Using Qiskit's AerSimulator provides a high level of randomness making it more secure than traditional random number generation methods.

3.2.2 AES-256

Files and messages are encrypted using AES-256 in Cipher Block Chaining (CBC) mode [22]. Before encryption, the plaintext data is padded using PKCS#7 to ensure proper alignment with AES block sizes allowing processing of files and messages of varying lengths. Randomly generated Initialization Vector (IV) is included to ensure that even if the same message is encrypted more than once, it results in different outputs each time adding an extra layer of protection [5].

3.2.3 Plaintext Padding

Let:

P be the plaintext

B be the AES block size (16 bytes)

The Padded plain text P' is computed as

$$P' = P||\big(B - (|P|mod\ B)\big).chr(B - (|P|modB))$$

Where:

|| denotes concatenation

|P| is the length of plaintext in bytes

Chr(x) returns a byte with ASCII value x.

The result P is appended with padding bytes.

3.2.4 CBC (Cipher Block Chaining) Encryption

Given:

K: 256-bit AES key

IV: 128-bit Initialization Vector (randomly generated)

P_i : i-th block of padded plaintext

C_i : i-th block of ciphertext

E_K : AES encryption with key K.

The encryption proceeds as:

$$C_0 = E_K(P_o \oplus IV)$$

$$C_i = E_K(P_o \oplus C_{i-1}) \qquad \text{for i} \geq 1$$

Where:

$\oplus$ denotes bitwise XOR

Each ciphertext block depends on the previous ciphertext, enhancing diffusion and making pattern analysis difficult

3.2.5 BB84 Protocol Simulation (Quantum Key Exchange)

The BB84 protocol is used to simulate a secure key exchange. Both the sender and the receiver generate random bits and bases. When their bases match, they derive a shared secret key from those matched bits. All the data from this simulated exchange is stored in the KeyLog table creating a secure trail for auditing and transparency [6].

Sender:

- Random bits: B= $[b_1,b_2,\ldots,b_n]$, $b_i \in \{0,1\}$
- Random bases: $\Theta=[\theta_1,\theta_2,\ldots,\theta_n]$, $\theta_i \in \{+, \times\}$

Receiver:

- Random bases: $\Theta'=[\theta_1',\theta_2',\ldots,\theta_n']$

Basis Matching:

Shared key is formed only when $\theta_i = \theta_i'$:

$$K_{shared} = \{b_i \mid \theta_i = \theta_i'\ '\}$$

Expected number of shared bits:

E [|K_{sha} |] = $\frac{n}{2}$t

Audit Logging:

All bit, basis, and match information are saved in the KeyLog table:

KeyLog={ (b_i, θ_i , $\theta_i{}'$ $match_i$)}

Logging

Once encryption and key exchange are complete, the system logs the encrypted file, recipient username, secret code, and other details into the FileTransfer table in PostgreSQL. Finally, a response is sent to the frontend indicating whether the transfer was successful or if any error occurred.

3.3 File Transfer-Receive

Recipients retrieve the file or message by entering the secret code provided by the sender. The backend queries the FileTransfer table to locate the encrypted data, then decrypts it using the stored AES key and IV. The decrypted content is returned with sender details and a timestamp. Errors are shown for invalid codes or decryption failures.

3.4 Key Management & Audit Logs

Admins access all logs while standard users view only their transactions. KeyLog and FileTransfer store encryption keys, BB84 parameters, timestamps and user details. The frontend displays logs ensuring transparency and auditability.

3.5 Data Structure and Mathematical Relevance

3.5.1 Hash Tables

Hash tables act as temporary storage for user credentials and multi-factorial authentication (MFA) tokens during active sessions. It provides constant-time complexity (O(1)) on average for insertion, lookup, and deletion operations.

This efficiency is essential for real-time authentications. Compared to linear structures like lists, hash tables scale seamlessly as the number of users increases.

3.5.2 Strings and Lists

Strings handle textual data like usernames, passwords and time-based one-time password (TOTP) tokens while lists handle ordered sequences including TOTP validation windows and BB84 protocol parameters.

3.5.3 Binary/Byte Arrays

Binary data (byte arrays) represent files and messages in their raw format allowing direct compatibility with AES encryption and decryption processes. It has linear time complexity of O(n)which aligns with the sequential nature of encryption and decryption ensuring predictable performance for larger files.

3.5.4 Base64 Encoding

Base64 encoding converts encrypted binary data into ASCII strings for safe transmission over text-based protocols like JSON or HTTP.

3.5.5 Hashing Algorithms

PBKDF2-HMAC-SHA256 is used to securely hash user passwords changing sensitive plaintext into irreversible cryptographic digests [23].

Equation:

DerivedKey=PBKDF2(Password, Salt, c, dk_len)

where:

- c: Iteration count (key stretching)
- dk_len: Desired key length
- Each iteration computes HMAC-SHA256 (Salt || Counter)

3.5.6 Entropy & Randomness

AES encryption keys are generated using QRNG.These keys have high entropy, ensuring high standards for resisting cryptographic attack.[24]

Shannon Entropy:

$$\boldsymbol{H(x) = -\sum_{i=1}^{n} P(x_i)\, log_2\, P(x_i)}$$

or a random key X, H(X) should approach n bits (max entropy).

3.5.7 Modular Arithmetic(TOTP Tokens)

Modulo operations underpin time-based one-time password tokens by synchronizing token generation with time intervals [25].

TOTP = HMAC-SHA1(K,T) mod 10^6

Where T=$\left\lfloor\frac{unixTime}{t_0}\right\rfloor$

3.5.8 Quantum Key Distribution(BB84)

By mimicking photon polarization and basis mismatches, the simulation shows how eavesdropping alters the transmitted bits showing quantum cryptography's security guarantees.

$$|+> = \frac{|0>, +|1>}{\sqrt{2}}(Diagonal\ basis)$$

|0⟩=Rectilinear basis

3.5.9 Symmetric Encryption

AES-256 encrypts data using substitution-permutation networks (S-boxes) and finite field arithmetic in Galois Field GF(2^8).

These enable AES to scramble data by byte substitution, row shifting and column mixing [26].

AES Round Function is given by
State=SubBytes(State)
→ShiftRows(State)→Mixcolumns(State)→
AddRoundKey(State,Ki)

3.5.10 Base64 Encoding

Base64 converts encrypted binary data (e.g., ciphertext) into ASCII strings using a 64-character alphabet.
Encode 3 bytes (24 bits) into 4 Base64 characters:
Output=Lookup(6-bit chunks of 24-bit input)

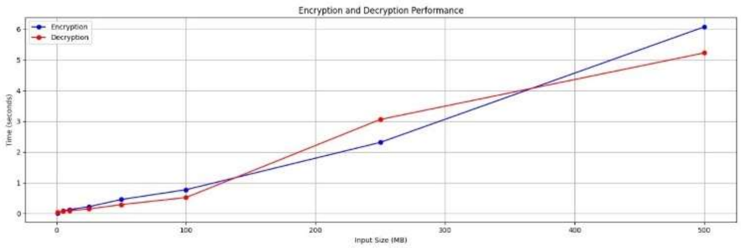
4. RESULTS AND DISCUSSION

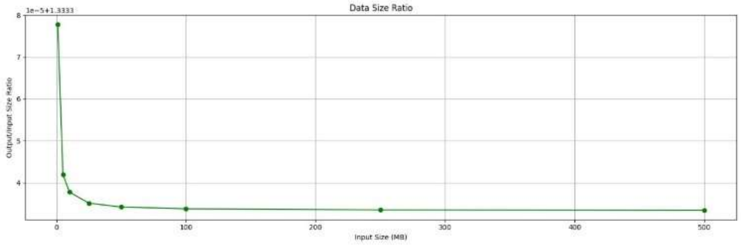
The developed cryptographic communication framework combines quantum random number generation, AES-256 encryption, BB84 quantum key distribution(QKD) protocol and graphical user interface (GUI) to enable secure file transfer and retrieval.

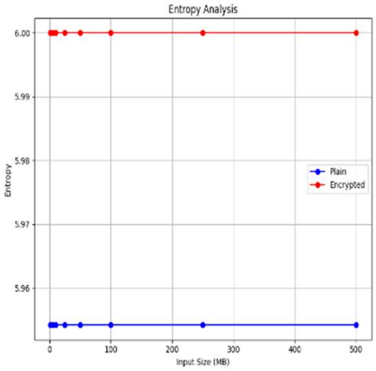
Quantum-generated AES-256 keys underwent evaluation, which confirmed that high entropy and true non-reproducibility are critical attributes for ensuring secure and unpredictable key generation.

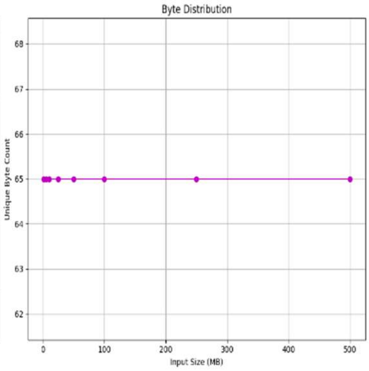
AES-256 encryption in Cipher Block Chaining (CBC) mode was also assessed by comparing entropy levels before and after encryption which consistently showed an improvement from approximately 5.95 to 6.00 underscoring the algorithm's capability to obfuscate data patterns effectively. Along with this, the BB84 quantum key exchange simulation played a vital role in securely establishing shared keys between sender and receiver, achieving an expected 50% basis match rate.

Size(in kb)	Encryption time(in seconds)	Decryption Time(in seconds)	Output size(in MB)	Plain Entropy	Encrypted Entropy	Entropy Increase
1	0.01	0.03	1.33	5.95	6	0.05
5	0.08	0.08	6.67	5.95	6	0.05
10	0.12	0.09	13.33	5.95	6	0.05
25	0.22	0.14	33.33	5.95	6	0.05
50	0.45	0.29	66.67	5.95	6	0.05
100	0.77	0.52	133.33	5.95	6	0.05
250	2.31	3.06	333.33	5.95	6	0.05
500	6.07	5.22	5.22	5.95	6	0.05
1000	Memory limit reached					









5. CONCLUSION

This project introduces a forward-thinking security system that combines classical encryption with quantum-resistant methods. By combining quantum random number generation and the BB84 protocol, it produces highly secure AES-256 encryption keys with high entropy (approximately 5.95) and ensures that keys cannot be reproduced. In simulations, the BB84-based quantum key exchange reached a 50% basis match rate consistent with theoretical expectations for security performance.

AES-256 in CBC encryption was tested, showing consistent performance with increased entropy (~6.00) and smooth scalability by successfully encrypting and decrypting 500 KB files in roughly 6.07 and 5.22 seconds, respectively. In order to enable the system to be used by anyone, GUI was built by allowing users to securely transfer and retrieve files with ease.

Altogether, this layered approach provides a powerful defense against both current and emerging cybersecurity threats which ideal for high-stakes environments like navy communications, where data confidentiality and integrity are mission-critical. The framework proves to be not just secure, but also practical and scalable for real-world use.

6. REFERENCES

[1]Sivakumar, J., Ganapathy, S. An Effective Data Security Mechanism for Secured Data Communications Using Hybrid Cryptographic Technique and Quantum Key Distribution. Wireless Pers Commun **133**, 1373–1396 (2023).

[2] Colbeck, M (Royal Navy), Quantum Cryptography and Military Communications, 2023-11-28, 11070

[3] M. Jofre, M. Curty, F. Steinlechner, G. Anzolin, J. P. Torres, M. W. Mitchell, and V. Pruneri, "True random numbers from amplified quantum vacuum," Opt. Express 19, 20665-20672 (2011)

[4] Vaishnavi Kumar, John Bosco Balaguru Rayappan, Rengarajan Amirtharajan, Padmapriya Praveenkumar,Quantum true random number generation on IBM’s cloud platform,Journal of King Saud University - Computer and Information Sciences,Volume 34, Issue 8, Part B,2022,Pages 6453-6465,SSN 13191578,

[5] Recommendation for Block Cipher Modes of Operation: Methods and Techniques(NIST SP 800-38A): https://www.ibm.com/docs/en/zos/2.4.0?topic=rules-pkcs-padding-method

[6] Quantum Key Distribution (QKD) and Quantum Cryptography (QC): https://www.nsa.gov/Cybersecurity/Quantum-Key-Distribution-QKD-and-Quantum-Cryptography-QC/#:~:text=Quantum%20key%20distribution%20utilizes%20the,over%20a%20dedicated%20communications%20link.

[7] C. Lee, I. Sohn and W. Lee, "Eavesdropping Detection in BB84 Quantum Key Distribution Protocols," in IEEE Transactions on Network and Service Management, vol. 19, no. 3, pp. 2689-2701, Sept. 2022, doi: 10.1109/TNSM.2022.3165202.

[8] D. Kim and S. C. Seo, "Efficient Optimization of MS Office 2013+ Password Cracking and PBKDF2-HMAC-SHA2 on GPUs," in IEEE Access, vol. 12, pp. 96436-96448, 2024, doi: 10.1109/ACCESS.2024.3426605.

[9] Saiyed, A. I. (2025). Hybrid Quantum-Classical Cryptographic Protocols: Enhancing Security in the Era of Quantum Supremacy. Spectrum of Research, 5(1)

[10]Krelina, M. Quantum technology for military applications. EPJ Quantum Technol. 8, 24 (2021). https://doi.org/10.1140/epjqt/s40507-021-00113-y

[11] Lazzarin, Jacopo, et al. "Quantum Key Distribution for Secure Encryption in Underwater Networks." Oceans2024, Department of Information Engineering, University of Padova, 2024.

[12] Sivakumar, Jananya, and Sannasi Ganapathy. "An Effective Data Security Mechanism for Secured Data Communications Using Hybrid Cryptographic Technique and Quantum Key Distribution." Springer Science, vol. 2024, no. 1, 30 Jan. 2024.

[13] Colbeck, M.J.L. "Quantum Encryption in Military Communications." Conference Proceedings of EAAW, 28-29 Nov. 2023

[14] Elboukhari, M., A. Azizi, and M. Azizi. "Implementation of Secure Key Distribution Based on Quantum Cryptography." 2009 International Conference on Multimedia Computing and Systems, Ouarzazate, Morocco, 2009, pp. 361-365. IEEE, doi:10.1109/MMCS.2009.5256673.

[15] Saiyed, Asif Iqbal. “Hybrid Quantum-Classical Cryptographic Protocols: Enhancing Security in the Era of Quantum Supremacy”. Spectrum of Research, vol. 5, no. 1, Jan. 2025

[16] V AD, V K. “Enhanced BB84 quantum cryptography protocol for secure communication in wireless body sensor networks for medical applications”. Pers Ubiquitous Comput. 2023;27(3):875-885. doi: 10.1007/s00779-021-01546-z. Epub 2021 Mar 18. PMID: 33758585; PMCID: PMC7971400.

[17] Feng Z, Li S, Xu Z. “Experimental underwater quantum key distribution.” Opt Express. 2021 Mar 15;29(6):8725-8736. doi: 10.1364/OE.418323. PMID: 33820314.

[18] Adu-Kyere A, Nigussie E, Isoaho J.”Quantum Key Distribution: Modeling and Simulation through BB84 Protocol Using Python3.” Sensors (Basel). 2022 Aug 21;22(16):6284. doi: 10.3390/s22166284. PMID: 36016045; PMCID: PMC9413261.

[19] Sun S, Huang A. “A Review of Security Evaluation of Practical Quantum Key Distribution System.”Entropy (Basel). 2022 Feb 10;24(2):260. doi: 10.3390/e24020260. PMID: 35205554; PMCID: PMC8870823.

[20] I. Giroti and M. Malhotra, "Quantum Cryptography: A Pathway to Secure Communication," 2022 6th International Conference on Computation System and Information Technology for Sustainable Solutions (CSITSS), Bangalore, India, 2022, pp. 1-6, doi: 10.1109/CSITSS57437.2022.10026388.

[21] Singh, Sunil K., et al. "Advancements in Secure Quantum Communication and Robust Key Distribution Techniques for Cybersecurity Applications." Cyber Security and Applications, vol. 2025, 100089, 2025, https://doi.org/10.1016/j.csa.2025.100089.

[22] Recommendation for Block Cipher Modes of Operation: Methods and Techniques (NIST SP 800-38A): https://csrc.nist.gov/pubs/sp/800/38/a/final

[23] @misc{pkcs5,
title={PKCS \#5: Password-Based Cryptography Standard},author={RSA Laboratories}, year={2012}, url={https://tools.ietf.org/html/rfc8018}

[24] Yeliz Karaca, Majaz Moonis,
Chapter 14 - Shannon entropy-based complexity quantification of nonlinear stochastic process: diagnostic and predictive spatiotemporal uncertainty of multiple sclerosis subgroups,
Editor(s): Yeliz Karaca, Dumitru Baleanu, Yu-Dong Zhang, Osvaldo Gervasi, Majaz Moonis,Multi-Chaos, Fractal and Multi-Fractional Artificial Intelligence of Different Complex Systems,Academic Press,2022,
Pages 231-245,ISBN 9780323900324,https://doi.org/10.1016/B978-0-323-90032-4.00018-3.(https://www.sciencedirect.com/science/article/pii/B9780323900324000183)

[25]Time-based One-time Password (TOTP): https://www.twilio.com/docs/glossary/totp

[26] National Institute of Standards and Technology (2001) Advanced Encryption Standard (AES). (Department of Commerce, Washington, D.C.), Federal Information Processing Standards Publication (FIPS) NIST FIPS 197-upd1, updated May 9, 2023. https://doi.org/10.6028/NIST.FIPS.197-upd1

APPENDIX

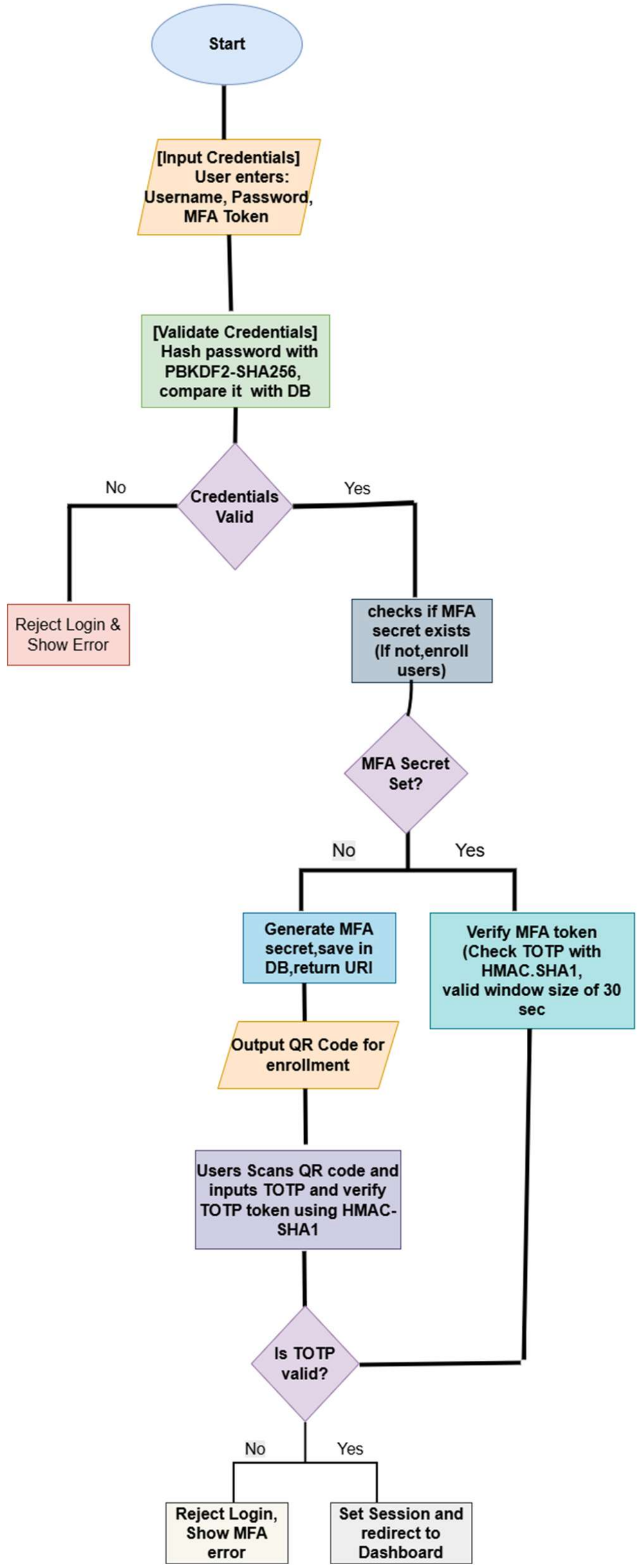


Figure 1: Data Flow Diagram of User Authentication and MFA Enrollment

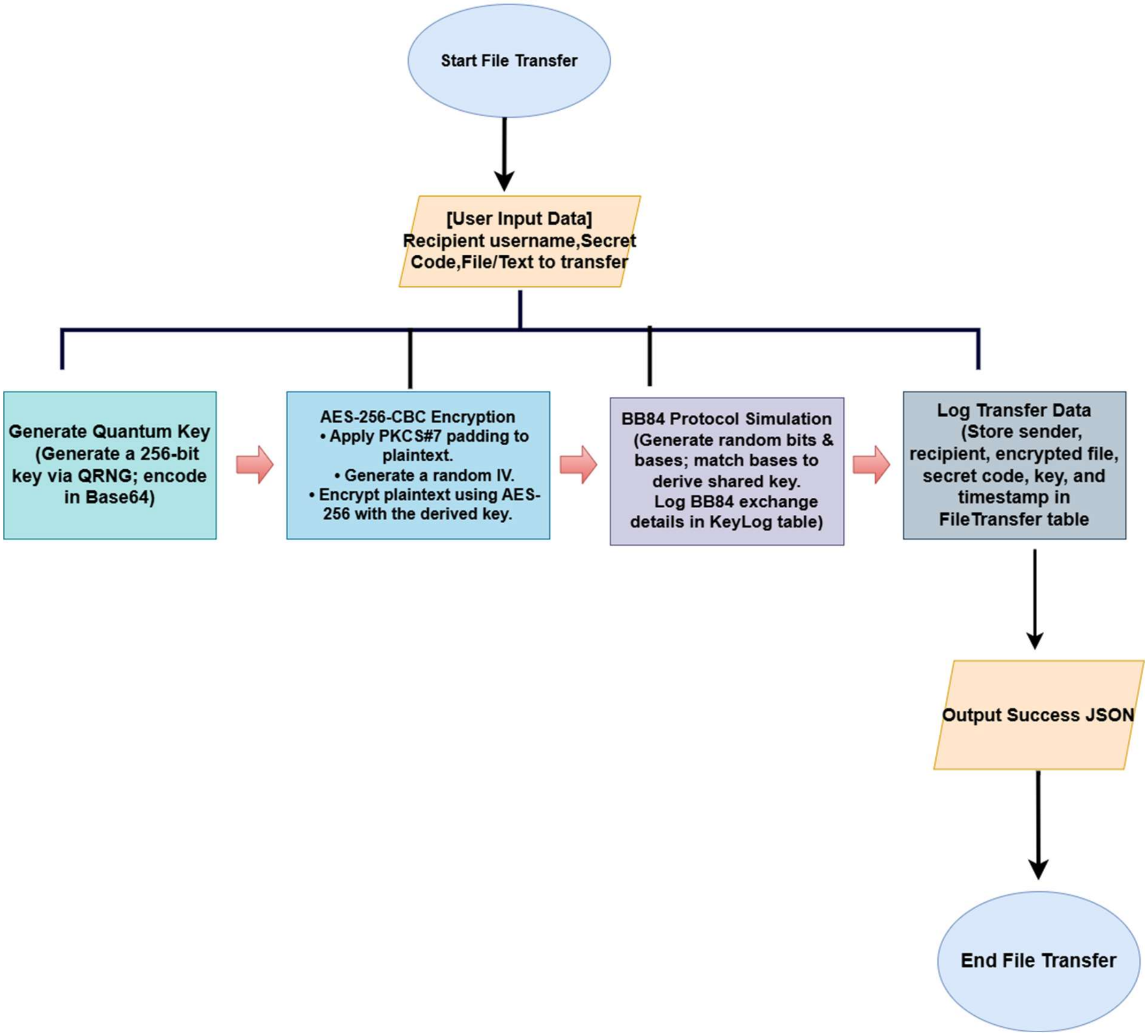

Start File Transfer
[User Input Data]
Recipient username,Secret Code,File/Text to transfer
Generate Quantum Key (Generate a 256-bit key via QRNG; encode in Base64)
AES-256-CBC Encryption
• Apply PKCS#7 padding to plaintext.
• Generate a random IV.
• Encrypt plaintext using AES-256 with the derived key.
BB84 Protocol Simulation (Generate random bits & bases; match bases to derive shared key.
Log BB84 exchange details in KeyLog table)
Log Transfer Data (Store sender, recipient, encrypted file, secret code, key, and timestamp in FileTransfer table
Output Success JSON
End File Transfer

Figure 2: Data Flow Diagram of Drop-Off

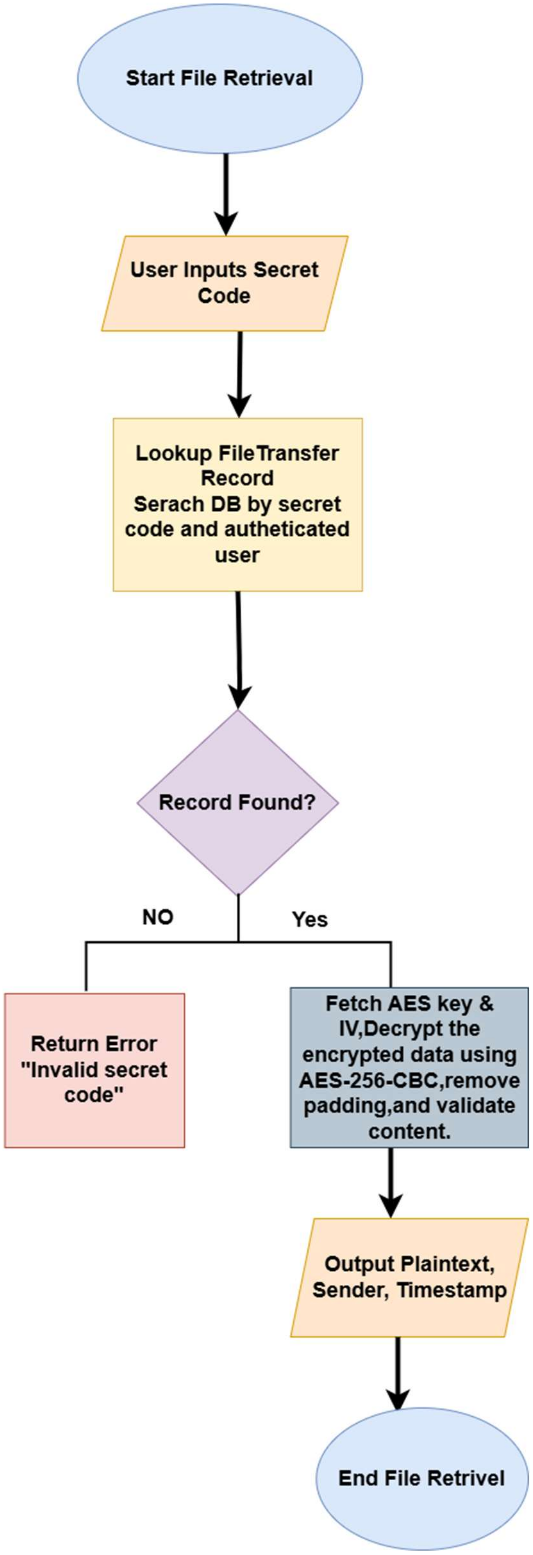

Start File Retrieval
User Inputs Secret Code
Lookup FileTransfer Record
Serach DB by secret code and autheticated user
Record Found?
NO
Yes
Return Error "Invalid secret code"
Fetch AES key & IV,Decrypt the encrypted data using AES-256-CBC,remove padding,and validate content.
Output Plaintext, Sender, Timestamp
End File Retrivel

Figure 3: Data Flow Diagram of Receive

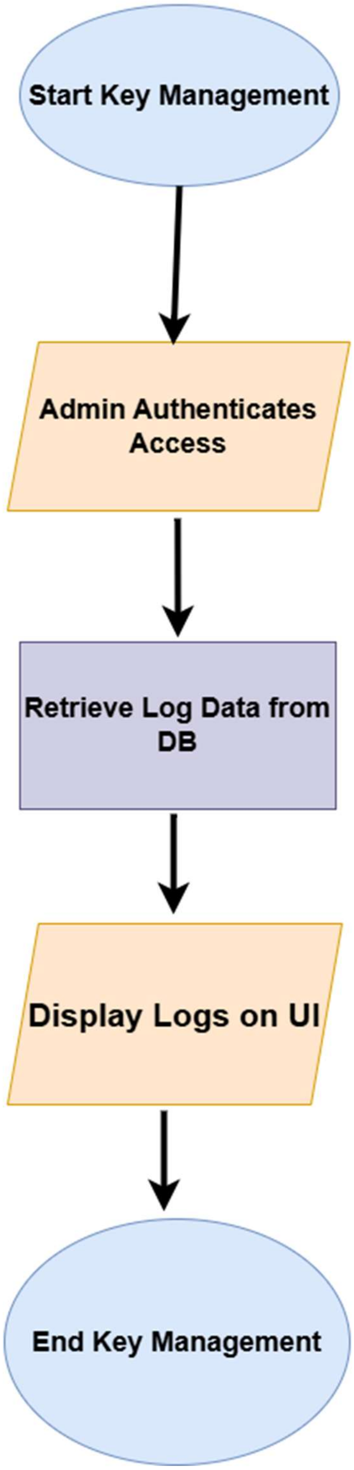
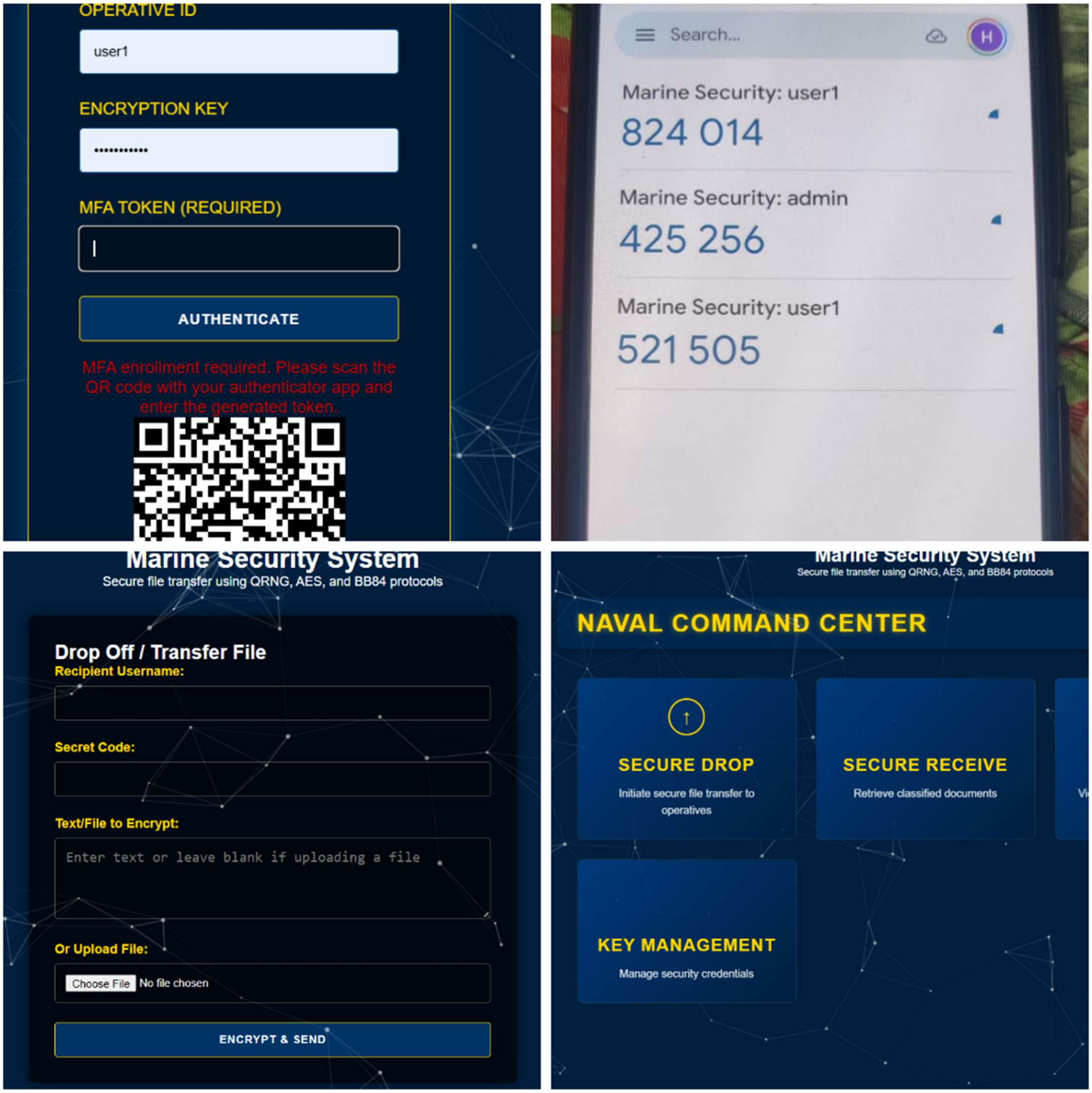


Figure 4: Data Flow Diagram of Key Management and Logs



OPERATIVE ID

user1

ENCRYPTION KEY

•••••••••••

MFA TOKEN (REQUIRED)

MFA enrollment required. Please scan the QR code with your authenticator app and enter the generated token.

Marine Security: user1
824 014

Marine Security: admin
425 256

Marine Security: user1
521 505

Marine Security System

Secure file transfer using QRNG, AES, and BB84 protocols

Drop Off / Transfer File

Recipient Username:

Secret Code:

Text/File to Encrypt:

Enter text or leave blank if uploading a file

Or Upload File:

No file chosen

Marine Security System

Secure file transfer using QRNG, AES, and BB84 protocols

NAVAL COMMAND CENTER

SECURE DROP

Initiate secure file transfer to operatives

SECURE RECEIVE

Retrieve classified documents

KEY MANAGEMENT

Manage security credentials

Figure 5: User Interface Dashboard