Journal of Information & Optimization Sciences
ISSN 0252-2667 (Print), ISSN 2169-0103 (Online)
Vol. 47 (2026), No. 1, pp. 273–286
DOI : 10.47974/JIOS-1974

FEDQIM : Fast easy decentralized quantum-proof instant messenger

D. R. Denslin Brabin *
W. Agitha †
K. Kalai Kumar §
P. Swarna Lakshmi ‡
Department of Computer Science and Engineering
DMI College of Engineering
Chennai 600123
Tamil Nadu
India

Abstract

Instant messengers can be used to exchange messages across different platforms like Android, IOS, Linux, Windows and MacOS. In this paper, an instant messenger called as Fedqim is proposed which is a fast, easy to use, decentralized and Quantum-Proof. It enables anyone to host their own instance of Fedqim and connect with other Fedqim users. The network is secured using the Fedqim Cryptographic Protocol. The Fedqim Cryptographic Protocol is inspired by Signal Cryptographic Protocol which is used in Whatsapp and Facebook. But unlike the original protocol, Fedqim Protocol is Decentralized and Quantum Proof public-key cryptosystem. Fedqim's primary focus is user privacy and security. Fedqim ensures that a user is never associated with their real identity like a Phone Number or E-Mail, that means no metadata is leaked.

Subject Classification: *Primary 68M12, Secondary 81P45.*

Keywords: *Cryptography, Communication, Information security, Instant messenger, Quantum computing.*

1. Introduction

Humanity depends on communication. Many communication technologies have emerged in recent years, but instant messaging is the

* *ORCID-ID:* **0000-0002-4377-0617**

* *E-mail:* **denscse@gmail.com (Corresponding Author)**
† *E-mail:* **ajithajerry@gmail.com**
§ *E-mail:* **kalaikumar23@gmail.com**
‡ *E-mail:* **swarnarp@gmail.com**

most sophisticated and practical of them all. For most people in today's world, it is the defacto method of communication. Instant Messaging facilitates almost instantaneous information sharing between two parties. There are several instant messengers available today; some of the more well-known ones are Telegram, Signal, and Whatsapp. Even though instant messaging has become more convenient and available to all, many instant messenger applications do not prioritize the security or privacy of its users. Additionally, the majority of widely used instant messengers are centralized, creating a single point of failure and enabling the collection of user data without user consent.

A central body does not have control over a decentralized digital network. Rather, control is shared by all of its users. There isn't just one server or central authority. Instead, the network is operated peer-to-peer, giving every user equal authority and accountability. The internet is a prime example of a decentralized network since it is not governed by a single entity. Instead, it is divided up among its users. Fedqim makes use of the federated and decentralized Fedqim Cryptographic Protocol. Every user in this digital network therefore has the same authority and accountability. Because you can easily transfer emails from Google to Yahoo, the protocol is quite comparable to electronic mail. Indeed, emails can be sent to Google from your own server. Similar to email, a user can self-host their own Fedqim instance and communicate with other Fedqim users who might be using it. Fedqim is decentralized, but even in the presence of a quantum computer, the cryptographic protocol makes sure that the messages cannot be altered or forgeried.

The Fedqim Protocol includes the Quantum-Proof Double Ratchet Algorithm and Quantum-Proof Triple Diffie-Helman Key Agreement Protocol (X3DH), all of which are derived from the Signal protocol [1][2], which offers the most advanced encryption available for two-way communication. Supersingular Isogeny Key Encapsulation (SIKE), which is deliberated to be immune against quantum computers, is used to encapsulate sensitive information in order to make the protocol quantum-proof while maintaining a modest memory footprint for embedded devices like Android phones. Additionally, the protocol uses digital signatures called Fast-Fourier Lattice-based Compact Signatures (FALCON), which are thought to be immune to attacks by quantum computers.

Fedqim using the Cryptographic Protocol offers privacy, integrity, authentication, participant consistency, destination validation, forward secrecy, post-compromise security (future secrecy), keeping the chain of

events intact, message unlinkability, message repudiation, participation repudiation, asynchronicity, and anonymity.

2. Literature Survey

Numerous security methods and algorithms for user communication have been developed recently. The well-known Peter Shor's Algorithm, which was presented in [3], employs both classical and quantum computers to compute a huge integer's prime factor in a workable amount of time. This makes public-key cryptosystems like RSA, Diffie-Hellman, and ECC easy to crack. Micali, Rabiny and Vadhan [4] introduced Verifiable Random Function (VRF), which offers a proof of the accuracy of its output that cannot be verified interactively. Subsequently, VRFs were used in security protocol design. A simple and effective method for creating a verified random function was provided by Dodis and Yampolskiy [5]. Nevertheless, of the dimension of the input, the proofs and keys of the VRF that is given are always the same size. Proofs are based on selectional bilinear Diffie-Hellman inversion notion.

The first ever off-the-record messaging protocol is proposed by Borisov, Goldberg, and Brewer [6]. They contend that the majority of online social communications ought to possess the exact opposite characteristics of the first two listed above: perfect forward secrecy and repudiability. The authors introduce "off-the-record messaging," a secure communication protocol with features more suited for casual chat than PGP or S/MIME. Additionally, they demonstrate in what way to create off-the-record communication as a Linux GAIM instant communication client plugin. Lastly, they go into how to get comparable privacy for communications with a significant latency, like email. The foundation for an elliptic curve's formal formulation and a quick implementation of Elliptic Curve Cryptography (ECC) is defined by Bernstein et al. [7] and Langley, Hamburg and Turner [8].

The Double Ratchet algorithm, as proposed by Alwen, Coretti and Dodis [9], allows two parties to interchange scrambled interactions using a common secret key. Typically, the parties use a key agreement technique (such as X3DH) to determine the shared secret key. The parties will then use the Double Ratchet to send and receive encrypted conversations. Unger et al. [10] evaluate, classify, and offer an evaluation system for the security, usability, and ease of adoption of existing secure messaging technologies. In addition to taking into account of solutions found in academic literature, they also recognize novel and promising techniques

that are employed in the wild but are not taken into account by it. They outline the design landscape for each of the three primary challenges - transport privacy, conversation security, and trust establishment - that they have identified.

Condensed, stable-time, and quick executions of the ChaCha20 stream cipher, Poly1305-ChaCha20 authenticator, and ChaCha20-Poly1305 method are presented by Santis, Schauer and Sigl [11] for ARM Cortex-M4 processors in an effort to assess the applicability of these algorithms for lightweight, high-speed Internet of Things applications. For example, they can be used to establish quick and safe TLS associations among IoT nodes and distant cloud servers once Advanced Encryption Standard (AES) hardware acceleration proficiencies are unavailable. The ciphertext length of the ChaCha20 stream cipher is the same as the plaintext, making it a great alternative to AES.

A novel implementation FALCON is proposed by Pornin [12]. Long-standing problems with the current reference code are resolved by the new implementation, which is constant-time, uses less RAM, does not require floating-point hardware (though it can use it for better performance), and performs significantly better on both small microcontrollers (ARM Cortex M4) and large systems (x86 with Skylake cores, POWER8,...). Specifically, the Falcon-512 requires fewer than 470k cycles to generate a signature on a Skylake processor (82k cycles for verification alone), and around 21.2 million cycles on an ARM Cortex M4. In terms of traditional security, Falcon-512 is comparable to RSA-2048, whose public keys and signatures require 256 bytes apiece. The author concludes that Falcon can now be said to be constant-time, and will be able to run on small embedded hardware that does not offer a hardware floating-point unit.

By mapping the Montgomery eliptic curves, Curve25519 and Curve448, to twisted Edwards curves, Faz-Hernández, López and Dahab [13] propose a safe and effective software implementation. This allows for the derivation of two new signature cases Ed25519 and Ed448 of the Edwards Digital Signature Algorithm, using SIMD (Single Instruction, Multiple Data) parallel processing. Elliptic curve cryptosystems are thought to be a more effective option than traditional ones like DSA and RSA. Cryptosystems have recently been implemented using the elliptic curves proposed by Montgomery and Edwards. Specifically, the Diffie-Hellman protocols X25519 and X448 were instantiated using the elliptic curves Curve25519 and Curve448. They get to the conclusion that their implementation outperforms cutting-edge implementations of the Diffie-

Hellman functions X25519/X448 and the digital signature schemes Ed25519/Ed448.

In order to provide an proficient assembly accomplishment of Supersingular Isogeny Key Encapsulation for 64-Bit ARM Cortex-A processors, which power almost all Android and iOS devices worldwide, Seo et al. [14] propose the first ever optimized implementation of SIKEp403 and SIKEp610 for 64-Bit ARM Cortex-A processors. The majority of processors used in portable devices are ARM-based. They conclude that their solution outperforms the standard modular multiplication implementation by a factor of 4.76x/5.45 after evaluating and contrasting the assembly approach with the current SIKE implementation.

The Blake3 hashing function is used to hash the user's public identity key, as stated in O'Connor et al. [15]. The resultant bytes are transformed into a hex string, which is then used as the user's unique network ID. No user with the same user ID exists throughout the whole decentralized network since the hashing function makes sure that even a small change in the public keypairs entirely alters the user id string. As a result, a user's user ID string should be 32 bytes long. For the effective and safe sharing of multimedia files, a hybrid technique based on ECC and the AES is presented in [16]. The generation of the sbox and its inverse in AES takes longer execution time. The algorithm that is being discussed uses inverse SBOX and fixed SBOX to shorten the execution time. With the ECC algorithm, the key is encrypted. Other security mechanisms like visual cryptography [17-19] and steganography [20-22] are also used to enhance the security of transmitted messages.

A single photon traveling via polarization is used in quantum cryptography[23]. When using quantum cryptography, the encrypted messages we send across optical fiber channels cannot be copied or altered by an eavesdropper due to the quantum states involved. Shor's algorithm and Grover's algorithm are the two fundamental algorithms in quantum cryptography. A pioneering procedure named as QuantQueryEXP was presented in [24] used for delivering web pages with maximum relevance. It integrates strategic quantum computing practices for skillfully increasing possibility to recommend web pages and contents more relevant to the input queries. The presented QuantQueryEXP procedure produces an overall accuracy of 90.43%.

The confidence of quantum cryptography is dependent completely on the laws of quantum mechanics [25]. A cryptosystem based on Vigenère cipher and using elliptic curve cryptography is porposed in [26]. The factors of the curve, the length of key and the point on the curve determine

the secure level of the presented cryptosystem. Gupta et al. [27] explores an open-source quantum computing simulation tool called Qiskit. They deliberate the essential philosophies of quantum computing and systematically investigates quantum logic gates, including X, Y, Z, CNOT. The behavior of basic blocks in quantum circuit is mimicked using Qiskit simulation tool.

3. Proposed System

The proposed Fedqim has two components, namely Fedqim Chat and Fedqim Core. Fedqim Chat is the client program which can run on multiple platforms, the client program uses the Fedqim Cryptographic Protocol to encrypt every message it sends locally before sending it over to the server, (i.e) the Fedqim Core. The Fedqim Core only acts as a temporary store for the encrypted messages, the encrypted messages do not reveal the sender but only the recipient of the message. Fedqim Core also helps to resolve user public key and other such information.

The Fedqim Chat maintains a local SQLite database to store all messages and contact information. The client program also stores the identify file which is never to be shared with anyone, this identity file is generated at random when the user registers with an Fedqim Core. Fedqim Core implements a Graph Query Language (GraphQL) interface to interact with, it maintains an Relational Database Management System (RDBMS) which stores public key of users. It also has inbox for the encrypted messages for each user that is registered with the Fedqim Core. It also maintains a Signed Prekey bundle for each user that is registered with the host.

A Fedqim User can be registered with any Fedqim Core over the internet, each Fedqim Core can interact with each other using the GraphQL interface. A user in Fedqim has inbox with their registered Fedqim Core, when new message is sent to this user at a specific Fedqim Core, the encrypted message is stored in this inbox. The Fedqim user then polls for new messages using GraphQL subscriptions. The user also updates their signed prekey bundle from time to time. This signed prekey is used for initiating a new session in Fedqim Cryptographic Protocol.

Figure 1 represents the high level architecture of Fedqim, which consists of two components as mentioned. The diagram shows how these two components interacts with each other. Bi-directional arrow denotes bi-directional flow of information from an entity and single directional arrow represents the flow of information in one direction only. The system

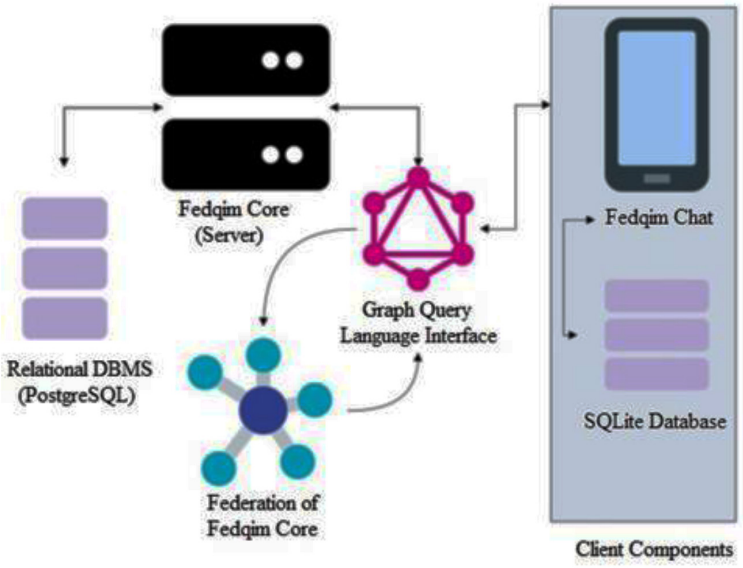


Figure 1

Architecture of Proposed System

has the ability to operate asynchronously. The following are the main components of the Fedqim Cryptogrpahic Protocol,

Fedqim Identity (FID): The identity in Fedqim decentralized network describes the identity of a user and how the public and private keypairs are constructed. It is composed of SIKEp503 keypairs and Falcon-512 keypairs.

Fedqim Transport Messages: Quantum-Proof encrypted messages which is used to transport arbitrary data. These message will expire after certain time limit and has unique identity attached to each message. This type of messages can be used to do authentication and for sending tokens over insecure channel.

Fedqim Token: Quantum-Proof authentication tokens which are similar to Javascript Web Tokens (JWT). Fedqim Token is used to prove identity of a user in a network. Fedqim Token's cannot be tampered or forged and are resistant to attacks from large scale Quantum Computers.

Fedqim Triple Diffie-Hellman Key Agreement (F3DH): Inspired from Extended Triple Diffie-Hellman Key Exchange, E3DH is a Quantum-Proof way to exchange shared secrets. When two parties mutually

authenticate using public keys, E3DH creates a shared secret key between them. E3DH offers cryptographic deniability and forward secrecy.

Fedqim Double Ratchet (FDR): It explains how two parties can exchange encrypted communications using a shared secret key by using the Quantum-Proof Double Ratchet technique. To prevent earlier keys from being estimated from later ones, the parties derive fresh keys for each Double Ratchet transmission. Along with their messages, the parties include Diffie-Hellman public values, which SIKE secures. In order to prevent later keys from being calculated from earlier ones, the outcomes of Diffie-Hellman calculations are mixed into the derived keys. In the event that a party's keys are compromised, these characteristics provide some protection for previously or later encrypted messages.

Identity in Fedqim is decentralized and does not need a centralized database to be identified and authenticated. Both authentication and identity resolution can be done cryptographically. Fedqim identity consist of public and private parts. Fedqim identity is composed of a Curve25519, SIKEp503 and Falcon-512 keypairs combined together. By mapping Curve25519 on Edwards curves, we can derive the signature algorithm Ed25519 of the Edwards Signature Algorithms. To make signatures Quantum-Proof, Falcon-512 is used to sign the same data again. The two level signature validation makes it stronger. Fedqim Public Identity key structure is shown in figure 2. Public identity is composed of the public keypairs of the Fedqim Identity. It has 20 bytes header which denotes the protocol, the type of entity and the protocol version which it belongs to. Fedqim Secret Identity key structure is shown in figure 3. Private identity is composed of the secret keypairs of the Fedqim identity. Similarly it has 20 bytes header which denotes the protocol, the type of entity and the protocol version which it belongs to.

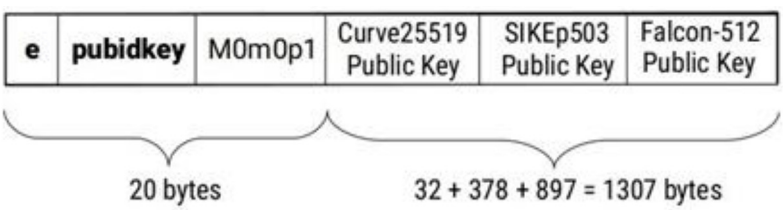


Figure 2

Fedqim Public Identity Key Structure

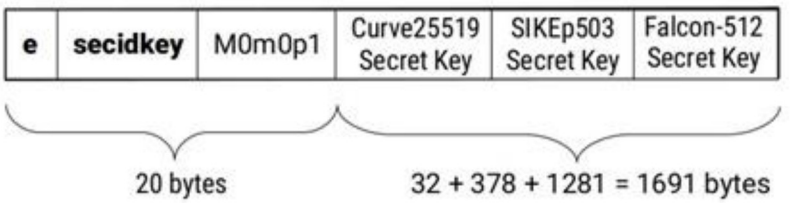


Figure 3

Fedqim Secret/Private Identity Key Structure

4. Results and Discussion

The proposed system is secure than existing systems while being fast as existing systems. Since Quantum Computers are still not easily accessible, benchmarking this is not possible but theoretically existing systems can be cracked within seconds with a Quantum Computer but this is not the case with our proposed system. Fedqim provides 128 bit security even in post-quantum era.

Alex Fedqim Public Identity = .pubidkey.M0m1p0.EthpR4WdLUS1
wutlszvXe7nyow6pnt/dUxHUgdmf2D2SxKWCQdmqIBNag0qp1Kebuqb
5XiZa1AEtKFcUVcKicuK4Xk6ganXrlVFscKCjptCTIv7Ahgv0qJKCID65
GyN2FynJAj+Z3QBUaSBqDo2tzT4eJ8J48FlN7RPWLlPKk5h87+P5xEw
sQz4DX2hwPHfl+hNSEZSG5MjO5pahaST9OrKq3MOM85nrheoRm4pR
YeZY134js62lDq0DYoMSEBgrhLljlFuoAY6tWxw8gxdhN4uJpcMhoLfP
gLbQBMQdt6mQ3sDfpMhuHYbSUOx63UytuOj+1yRGxqnagEmM4N0z
2nQP+K4rW6M2lhpI/ZDgyEUR5UEXN4D/2KbXt929Toif9Vs0nHiw4ud
zVH8e46tyJqbdz6+dMe2YbYu3fLYvVVK6mSRRKa0WUwAa7zojZb1a
9uKNf0zhxY4tcfSPovJJdbAMlHEsi9Z1u==

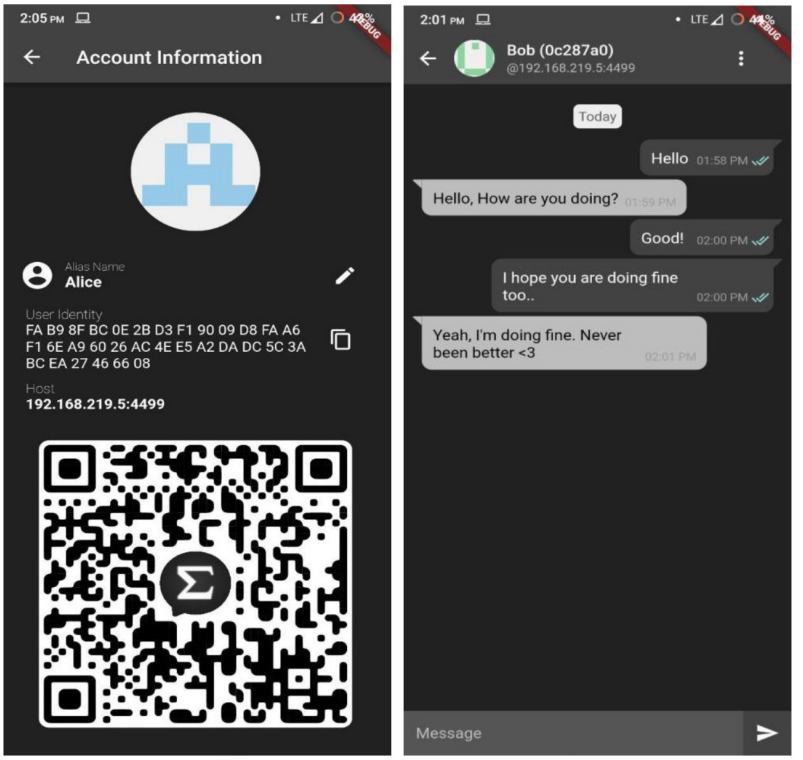
Alex User ID = e.userid.M0m1p0.8679b7d716d491a595d
7877b579e446558984cc6b3e4fd15 d4a41e0f3e7d8c

Blake Fedqim Public Identity = e.pubidkey.M0m1p0.LzpbAs
WTdjFtjKmOEV51k/vXTeRnseoYSj5HxENyuD0oM+1Rytq2Sm7zh
sEewgW/MUpSpJbLaW6YdXuivEsXE2EOJ6Ejl+JCKhQOO1Qpd9cf
hAkcgW8scaFHwwIX4C0mCo+5oEE+u8H04HtXGiV3fEfwXZmVB
2/JDiogDrkKlDKszi1Cy1bb7KN/d538ODwVcB+ihbgU1RjxpLfKdn4
4zv2hdw9VHd6Xf74AZupwpZyL8O4ofVyzeRuLa3IrZIgslU3p7/iTx
mh7wAKwcgvYMM+XoEoGYJ5/EY7lBvh9WhtfyXupechx+uARre
QYk9iI8UsYLkNfe4rZ4oMIl6NMamwwg8HFEaxccelktVHcxvzVGh
yxxmLsd1ZwuUcDkaaEuRD3pOWV5jFzqDLiSUSJoiYd8QOWA1Y
riSUm8l2eAvCCHplLIaczwyN0G8q0DLMfXe/CvmNvOI7MDld+0gt
Rv4aPOnk6LpLWTc5kg==

Blake User ID = e.userid.M0m1p0.9cdf36500345a09294
f64579006cf4822f13eb0df6171510378 7b638 61875101

Figure 4

Basic structure of Fedqim public identity



(a) (b)

Figure 5

Chat Pages Screen Shot (a) Alice's Account (b) Chat page of Bob

Consider two users who is using Fedqim to communicate with each other, Alice and Bob. Alice and Bob generates their pesudo-random Fedqim identity. The basic structure of Fedqim public identity is shown in figure 4. Then Alice gets a random prekey of Bob from an Fedqim Core he is registered with, this prekey is used to initiate a double ratchet for a new session between the users. Alice computes the Fedqim Triple Diffie-Hellman using the prekey of Bob and sends the encrypted session init message to Bob which he does the same computation to derive a shared secret. Bob then decrypts the session init message from Alice and does the same E3DH computation to get the shared secret.

After session init, Alice sends a Fedqim double ratchet message which is the first message in this new session. Then both Alice and Bob send and receive messages securely. Identity of Alice is shown in figure

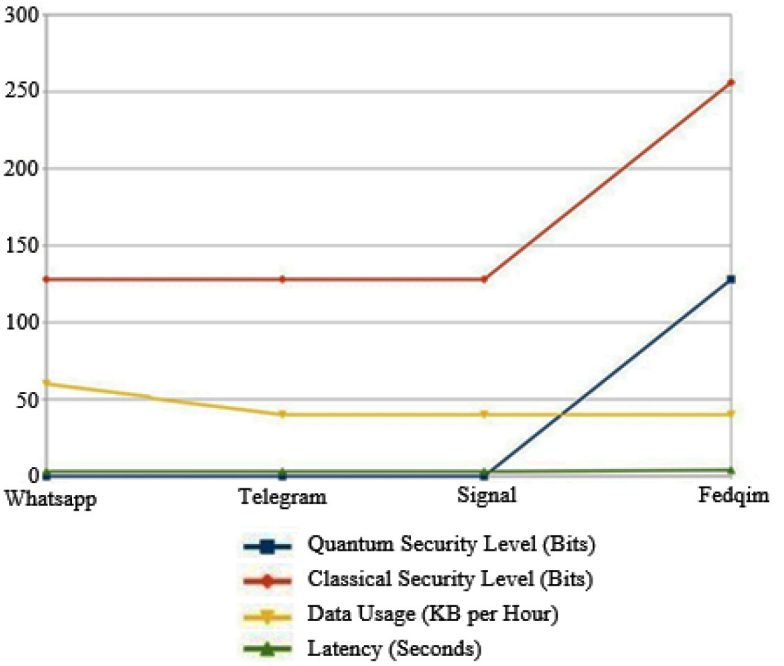


Figure 6

Comparison of Fedqim with Existing Systems

5(a) and Chat page of Bob is shown in figure 5(b) as sample screen shot of Fedqim. Figure 6 compares the security level and efficiency of Fedqim with existing systems. The green line in the graph represents the security level in bits in quantum era, as we can see only Fedqim provides 128 bit security in the quantum era while Signal, Whatsapp and Telegram provides no security in the quantum era. Fedqim is fast as existing system and uses very less internet data.

5. Conclusion

The proposed system is secure and open than existing systems, it is unique from all other systems because it is Decentralized and Quantum-Proof. Fedqim also provides maximum user privacy, anonymity and security unlike any existing systems which are currently in use. Thus Fedqim enables us to send and receive text messages in decentralized and quantum-proof way. The proposed system can be modified and used in different application such as digital payments and file sharing applications.

Future work include introducing truly secure group chats, DNS based Identity resolution and Improving the UI and UX of the Chat application, support to send image, video and other multimedia files to anyone in the Fedqim network.

Acknowledgment

We would like to express our special thanks to DMI College of Engineering for providing the support to carry out the research work.

References

[1] M. Marlinspike and T. Perrin, "The X3DH Key Agreement Protocol," *Signal Blog*. (Nov. 2016), https://signal.org/docs/specifications/x3dh/x3dh.pdf

[2] T. Perrin and M. Marlinspike, "The Double Ratcheting Algorithm," *Signal Blog*. (Nov. 2016), https://signal.org/docs/specifications/doubleratchet/doubleratchet.pdf

[3] P. W. Shor, "Polynomial-time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer," *SIAM ournal of Computing,* vol. 26, no. 5, pp. 1484-1509 (1997), doi: 10.1137/S0097539795293172.

[4] S. Micali, M. Rabiny and S. Vadhan, "Verifiable Random Functions," in *Proceedings of 40th Annual Symposium on Foundations of Computer Science,* pp. 120-130 (Oct. 1999), doi: 10.1109/SFFCS.1999.814584.

[5] Y. Dodis and A. Yampolskiy, "A Verifiable Random Function with Short Proofs and Keys," *Lecture Notes in Computer Science*, vol. 3386, pp. 416–431 (2005), doi: 10.1007/978-3-540-30580-4_28.

[6] N. Borisov, I. Goldberg, and E. Brewer, "Off-the-record Communication, or, Why not to use PGP," in *Proceedings of ACM Workshop on Privacy in the Electronic Society*, pp. 77-84 (Oct. 2004), doi: 10.1145/1029179.1029200.

[7] D. J. Bernstein, N. Duif, T. Lange, P. Schwabe and B. -Y. Yang, "High-speed High-security Signatures," *Journal of Cryptographic Engineering*, vol. 2, pp. 77-89 (2012), doi: 10.1007/s13389-012-0027-1.

[8] A. Langley, M. Hamburg and S. Turner, "Elliptic Curves for Security," *Internet Engineering Task Force; RFC Informational* (2016), http://www.ietf.org/rfc/rfc7748.txt.

[9] J. Alwen, S. Coretti and Y. Dodis, "The Double Ratchet: Security Notions, Proofs, and Modularization for the Signal Protocol," in *Advanc-*

es in Cryptology, Lecture Notes in Computer Science, vol. 11476 (2019), doi: 10.1007/978-3-030-17653-2_5.

[10] N. Unger, S. Dechand, J. Bonneau, S. Fahl, H. Perl, I. Goldberg and M. Smith, "Sok: Secure Messaging," in *IEEE Symposium on Security and Privacy*, pp. 232-249 (2015), doi: 10.1109/SP.2015.22.

[11] F. De Santis, A. Schauer and G. Sigl, "Chacha20-poly1305 Authenticated Encryption for High-speed Embedded IoT Applications," in *Design, Automation Test in Europe Conference & Exhibition*, pp. 692-697 (2017), doi: 10.23919/DATE.2017.7927078.

[12] T. Pornin, "New Efficient, Constant-Time Implementations of Falcon," in *IACR Cryptology ePrint Archive*, 2019/893 (2019), https://ia.cr/2019/893.

[13] A. Faz-Hernández, J. López and R. Dahab, "High-performance Implementation of Elliptic Curve Cryptography using Vector Instructions," *ACM Transactions on Mathematical Software*, vol. 45, no. 3, Art. no. 25, pp. 1-35 (2019), doi: 10.1145/3309759.

[14] H. Seo, P. Sanal, A. Jalali and R. Azarderakhsh, "Optimized implementation of sike round 2 on 64-bit arm cortex-a processors," *IEEE Transactions on Circuits and Systems*, vol. 67, no. 8, pp. 2659–2671 (2020), doi: 10.1109/TCSI.2020. 2979410.

[15] J. O'Connor, J. P. Aumasson, S. Neves and Z. Wilcox-O'Hearn, "Blake3: One Function, Fast Everywhere," (2022), https://raw.githubusercontent.com/BLAKE3-team/BLAKE3-specs/master/blake3.pdf.

[16] Riddhi Somaiya, Atul Gonsai, Rashmin Tanna "Implementation and evaluation of EMAES –A hybrid encryption algorithm for sharing multimedia files with more security and speed", *International Journal of Electrical and Computer Engineering Systems*, vol. 14, no. 6, pp. 401-409 (2023), doi: 10.32985/ijeces.14.4.4.

[17] A. Yildizhan and N. Topaloglu, "Chaotic Encryption and Privilege Based Visual Secret Sharing Model for Color Images," *Computing and Informatics*, vol. 38, no. 3, pp. 701–727 (2019), doi: 10.31577/cai.2019.3.701

[18] D. R. Denslin Braja and V. S. Dharun, "Identification of Person or Data using Modified Square Blockwise Approach," *International Journal of Reasoning-based Intelligent Systems*, vol. 11, no. 2, pp. 103-108 (2019), doi: 10.1504/ IJRIS.2019.099845.

[19] Peng Li, Jianfeng Ma, Liping Yin and Quan Ma, "A Construction Method of (2, 3) Visual Cryptography Scheme," *IEEE Access*, vol. 8, pp. 32840-32849 (2020), doi: 10.1109/ACCESS.2020.2973659.

[20] S. Dhawan, C. Chakraborty, J. Frnda, R. Gupta, Rana Ak and Pani SK. "SSII: secured and high-quality steganography using intelligent hybrid optimization algorithms for IoT," *IEEE Access*. vol. 9, pp. 87563-87578 (2021), doi: 10.1109/ACCESS.2021.3089357.

[21] G. Peter, A. Sherine, Y. Teekaraman, R. Kuppusamy and A. Radhakrishnan, "Histogram shifting-based quick response steganography method for secure communication," *Wireless Communications and Mobile Computing*, Art. no. 1505133 (2022), doi: 10.1155/2022/1505133.

[22] D. R. Denslin Brabin, Christo Ananth and Sriramulu Bojjagani, "Blockchain based security framework for sharing digital images using reversible data hiding and encryption," *Multimedia Tools and Applications*, vol. 81, pp. 24721–24738 (2022), doi: 10.1007/s11042-022-12617-5.

[23] B. Muruganantham, P. Shamili, S. Ganesh Kumar and A. Murugan, "Quantum cryptography for secured communication networks," *International Journal of Electrical and Computer Engineering*, vol. 10, no. 1, pp. 407-414 (2020), doi: 10.11591/ijece.v10i1.pp407-414.

[24] I. S. Kaushik, G. Deepak, and A. Santhanavijayan, "QuantQueryEXP: A Novel Strategic Approach for Query Expansion Based on Quantum Computing Principles," *Journal of Discrete Mathematical Sciences and Cryptography,* vol. 23, no. 2, pp. 573–584 (2020), doi: 10.1080/09720529.2020.1729506.

[25] Christopher Portmann and Renato Renner, "Security in quantum cryptography," *Reviews of Modern Physics*, vol. 94, Art. no. 025008 (2022), doi: 10.48550/arXiv.2102.00021.

[26] Mai Manh Trung, Le Phe Do, Do Trung Tuan, Nguyen Van Tanh and Ngo Quang Tri, "Design a cryptosystem using elliptic curves cryptography and Vigenère symmetry key," *International Journal of Electrical and Computer Engineering*, vol. 13, no. 2, pp. 1734-1743 (Apr. 2023), doi: 10.11591/ijece.v13i2.pp1734-1743.

[27] Sumit Gupta, Mayank Namdev, Ankur Goyal, Srinivas Samala, Deepika Dave, and Dheresh Soni, "Exploration of Quantum Computing and Communication Blocks with IBM Qiskit," *Journal of Discrete Mathematical Sciences and Cryptography,* vol. 27, no. 7, pp. 2041–2052 (2024), doi: 10.47974/JDMSC-2078.

Received September, 2024