ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 11, Issue 5, May 2024

Implementing FourQ Elliptic Curve Cryptography for Secure Chats

[1] Divi Mahendra Sai*, [2] Gowtham R, [3] C S Darshan, [4] Deep Narayan Mahapatra
[1] [2] [3] [4] Department of Computer Science & Engineering, MVJ College of Engineering, Bangalore, Karnataka, India
Corresponding Author Email: [1] Mahendra.doak@gmail.com

Abstract*— The need for robust cryptographic systems is emphasized by the increasing cybersecurity threats. Efficiency and security in communication are guaranteed with a practical implementation of Elliptic Curve Cryptography (ECC) using FourQ curve. This system will allow two parties to communicate securely over the internet using ECC. Key generation is performed using FourQ, AES is used as the encryption and decryption algorithm, while Diffie-Hellman Key Exchange algorithm is applied. The ECC-FourQ implementation has been evaluated by the paper to showcase its usage in real world scenarios, highlighting its effectiveness, efficiency and high security. The research seeks to improve knowledge on ECC using FourQ and its application in securing communication channels. Cyber threats can be effectively addressed through this system's efficiency and security levels that have been demonstrated here.

Index Terms: Diffie-Hellman Key Exchange, Elliptic Curve Cryptography (ECC), FourQ curve, Secure Communication Performance Evaluation.

I. INTRODUCTION

Nothing in this digital age is as important as the need for secure communication. The way cryptography has evolved over the ages is remarkable, from Caesar Cipher to up-to-date models of AES and Public Key Cryptosystem illustrate the fight against insecurity to protect vital information. It is through cryptography (Fig. [1]) that military communications, financial transactions, personal data and other uses have confidence and integrity.

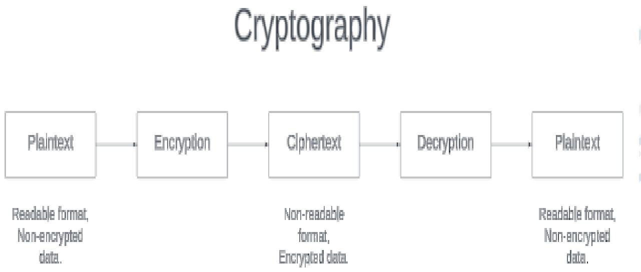
Cryptography
Plaintext
Encryption
Ciphertext
Decryption
Plaintext
Readable format, Non-encrypted data.
Non-readable format, Encrypted data.
Readable format, Non-encrypted data.

Fig 1. Cryptography Outline

Compared to traditional systems, Elliptic Curve Cryptography (ECC) represents a major advance in cryptography offering greater security with smaller key sizes. ECC employs elliptic curves over finite fields' algebraic structures for efficient cryptographic operations while maintaining their security. One interesting feature of ECC is FourQ curve which is distinguished by its performance and security properties.

The second most important aspect of modern-day cryptography is AES which enables secure and quick symmetric key encryption. AES frequently comes into use in applications such as secure messaging or encrypting data.

The Diffie-Hellman key exchange algorithm performs a crucial function in creating secure communication channels over unsecured networks. It allows two bodies to decide on a common secret key in the absence of prior communication, thereby ensuring privacy.[1][6][7]

Furthermore, Secure Hash Algorithm (SHA)-256 provides hashing and data integrity services. It produces a fixed-sized hash value that is unique for each input data which can be used to verify if transmitted messages have been tampered with.

In this digital era where online communication is prevalent, cryptographic technologies must be implemented into chat applications to enhance security and privacy. This research paper titled "Implementing FourQ Elliptic Curve Cryptography for Secure Chats" explores practical implementation of ECC with FourQ, AES, Diffie-Hellman and SHA-256 on chat application. Its aim is to show how these methods can improve security and privacy by focusing on their role in enhancing online communications. [1] [2] [6] [7]

For the elliptic curve (Fig [2]) FourQ which is a specific version of Q-curves based on recent designs, it has an endomorphism φ that can be calculated with ease accompanied by low degree. Also, the researchers created a library for ECC that functions within E(Fp^2) [N], which is focused on 128-bit security level curves in other words, it uses prime N = 246 bit.[2]

FourQ and its library have several advantages over existing curves and implementations. The use of FourQ allows scalar multiplications to be performed much more efficiently than all known software implementations of curve-based cryptographic primitives due to the existence of endomorphisms ψ and φ. Being easily implemented correctly used in cryptographic protocols for various purposes supporting fast fixed-base and variable-base scalar multiplications for the ephemeral Diffie-Hellman key exchange and Schnorr-like digital signatures.[2]

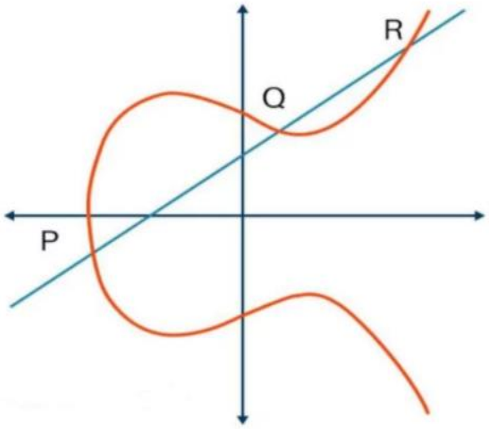

Fig 2. Elliptic Curve

In line with our study, we carried out an extensive performance analysis on scalar multiplication operations on FourQ elliptic curve with an aim to determine the performance advantage over other implementations running on ARM Cortex-A processors. It is found that for variable base scalar multiplication on Cortex-A8 requires 235000 cycles and 132000 cycles for Cortex-A15. These outcomes are higher than those of the top performing genus 2 Kummer and Curve25519 designs by a factor between 1.3 - 1.7 times and between 2.1 - 2.4 times in the same platforms respectively. Moreover, our approach achieves a speedup factor of more than four times and five point five times compared to NIST standard curve K283. [4]

The paper discusses benchmark results for a vectorized FourQ implementation. It specifically looks at the efficiency improvements achieved through fixed-base scalar multiplications with minimal precomputations. Please note that this result indicates that FourQ can be used in signature verification for most target platforms such as those requiring less than 15% incremental cost beyond single variable-base scalar multiplications as one double-scalar multiplication.[4]

TABLE 1. Performance results (in terms of thousands of cycles) of core scalar multiplication operations on FourQ with protection against timing and cache attacks on various ARM Cortex-A processors with NEON support.[4]

Scalar Multiplication	Cortex-A7	Cortex-A8	Cortex-A9	Cortex -A15
[k]P, variable base	373	235	256	132
[k]P, fixed base	204	144	145	84
[k]P + [l]Q	431	269	290	155

II. RELATED WORKS

A. The cryptographic security of FourQ

The FourQ curve is considered cryptographically secure due to its resistance to attacks, including ECDLP in E(Fp2) [N). Its security relies on its design considerations and properties, including Pollard's Rho Algorithm, endomorphisms, and complexity comparison. The ECDLP on FourQ is similar to other curves in speed-record literature. The curve's security and field choice are largely due to its field speeds at 128-bit level and four-dimensional scalar decompositions. The curve's low discriminant and rational 2-torsion make it a suitable choice for discrete logarithm problems. [2]

B. FourQ on Embedded Devices with Strong Countermeasures Against Side-Channel Attacks

The study demonstrates the very first realizations of FourQ elliptic curve based on microcontrollers with a high level of performance ever, which is the best in comparison with other curves for curve-based scalar multiplication, Diffie-Hellman key exchange and digital signatures. The Elliptic Curve Diffie Hellman key exchange computations take less than one second for an 8MHz clocked AVR microcontroller at the 128-bit security level. SchnarrQ signings are also discussed in this paper were signing computations are more than two times faster than those of μKummer, previously the quickest and almost four times as fast as fastest Ed25519 implementation known. The paper further makes suggestions about protection from various side-channel attacks by giving secure algorithms.[3]

III. METHODOLOGY ADOPTED

A. Authentication

Our approach begins with ensuring that the people who access secure chat application are genuine. This is done using Advanced Encryption Standard (AES) in conjunction with SHA-256 hashing algorithm. In this case, AES is used to encrypt credentials such as passwords and usernames of users so that they can be securely transmitted via a network.SHA-256 then hashes these credentials generating fixed size output which can be used to uniquely identify a user. This hash would be compared with the hash stored in the database, which contains hashes of authorized users. Using this combination of cryptographic techniques, we create a reliable and secure process for verifying the identities of users before allowing them into the chat application. The authentication flow is show in Fig. 3. [8]

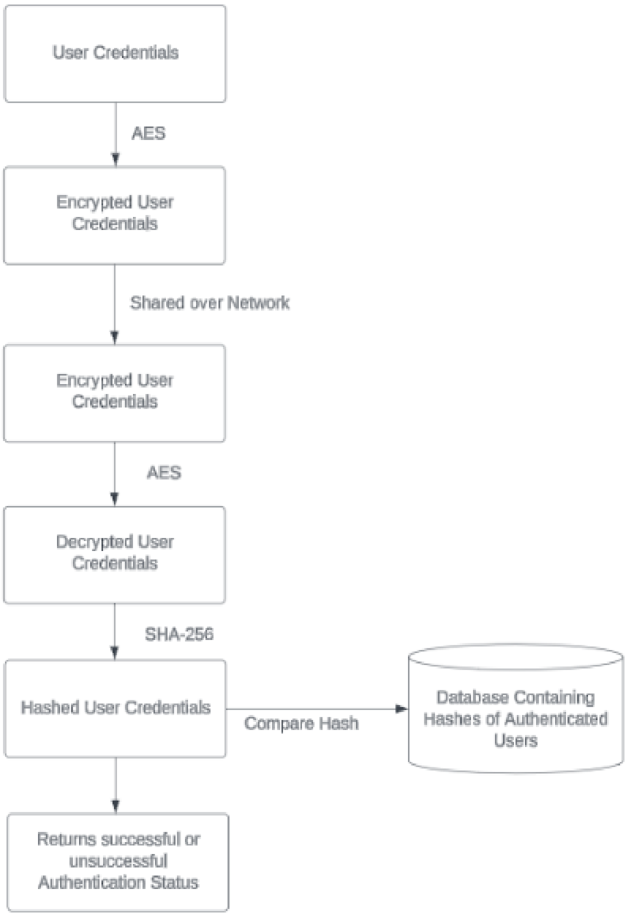
User Credentials
AES
Encrypted User Credentials
Shared over Network
Encrypted User Credentials
AES
Decrypted User Credentials
SHA-256
Hashed User Credentials
Compare Hash
Database Containing Hashes of Authenticated Users
Returns successful or unsuccessful Authentication Status

Fig 3. Authentication Process

B. Generating Shared Keys

As soon as authentication is completed and clients are ready to chat with others who are online, we will begin with shared key generation using Diffie-Hellman with FourQ ECC. The second phase of our approach involves generating shared keys for secure communication between users. We utilize Diffie-Hellman key exchange algorithm to generate a shared secret key which enable two parties to communicate secretly even on an insecure channel. Our implementation employs elliptic curve cryptography (ECC) library FourQ which generates public and private keys that are needed for Diffie-Hellman algorithm. This method guarantees that the shared keys can be created efficiently and maintained secure against any eavesdropping or interception attempts. Through combining FourQ ECC with Diffie-Hellman, we create a strong framework to securely send messages within the chat application. The Key generation is shown in Fig. 4.[6][7][8]

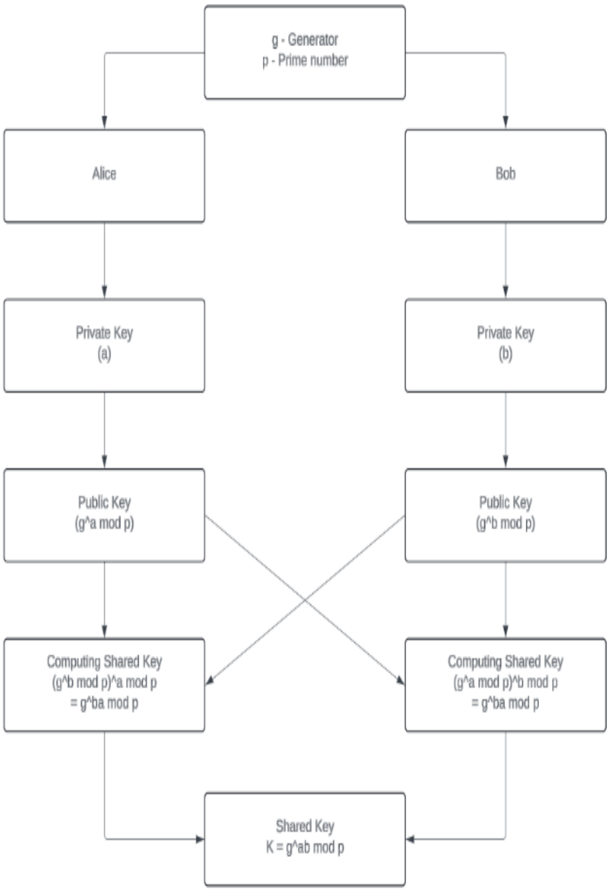
g - Generator
p - Prime number
Alice
Bob
Private Key (a)
Private Key (b)
Public Key (g^a mod p)
Public Key (g^b mod p)
Computing Shared Key (g^b mod p)^a mod p = g^ba mod p
Computing Shared Key (g^a mod p)^b mod p = g^ba mod p
Shared Key K = g^ab mod p

Fig 4. Generating shared key using Diffie-hellman

C. Encrypting and Decrypting Messages using Shared Key

Subsequent to the authentication phase, wherein users are verified and granted access, the communication process advances to the encryption and decryption stage. In this phase of secure message exchange between authenticated users, unique shared keys serve as a basis for it. Before sending, messages are encrypted by employing the strong AES-256 encryption algorithm. The recipient's designated shared key is used in this encryption process thereby ensuring that messages remain private and resistant to unauthorized access through network during transmission. Using their corresponding shared keys, on receiving; the receiver can decrypt a message back into its original form for understanding purposes. This iterative process of encryption and decryption using shared keys enhances confidentiality and integrity of all conversations within chat sessions thus meeting tight security requirements for protection sensitive data. The Encrypting and Decrypting Messages using Shared Key is shown in Fig. 5.[7] [8]

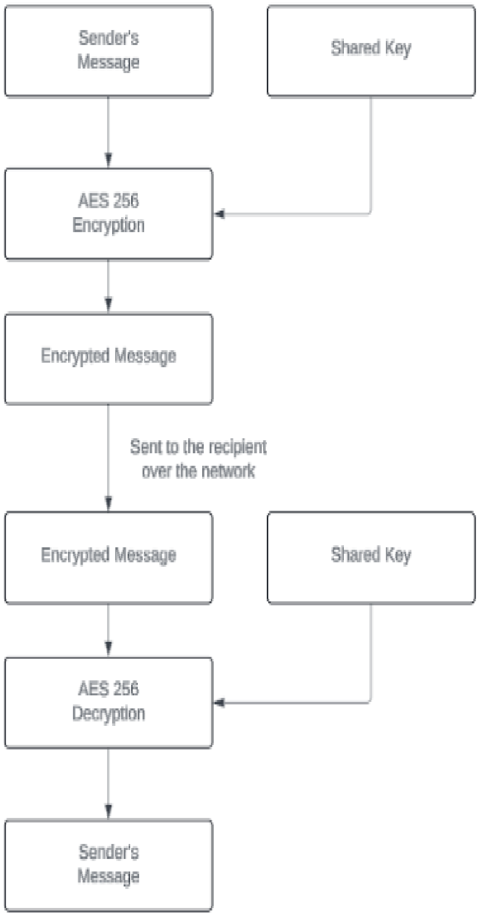

Fig 5. Encrypting and Decrypting Messages using Shared Key

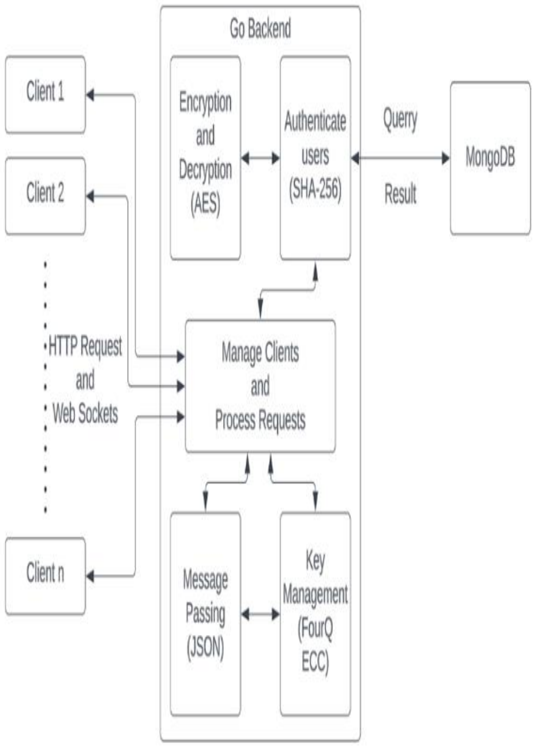
Go Backend
Client 1
Client 2
HTTP Request and Web Sockets
Client n
Encryption and Decryption (AES)
Authenticate users (SHA-256)
Query
Result
MongoDB
Manage Clients and Process Requests
Message Passing (JSON)
Key Management (FourQ ECC)

Fig 6. Architecture Diagram

IV. RESULTS

The project focusing on the implementation of FourQ Elliptic Curve Cryptography (ECC) for secure chat applications has produced compelling results, showcasing ECC's practicality in enhancing communication security. Through the integration of ECC, the project achieved robust user authentication, ensuring that only authorized users could access the chat system. By utilizing AES and SHA-256 for authentication, a secure foundation for communication was established. The project's implementation of the Diffie-Hellman algorithm for key exchange further fortified security, facilitating the secure generation of shared keys between communicating parties.

Furthermore, the project effectively employed the FourQ elliptic curve for message encryption, guaranteeing the confidentiality and integrity of the exchanged data. This implementation underscored the efficiency and efficacy of FourQ ECC in securing chat communications. The project's outcomes not only validate the practical application of ECC but also highlight its potential to enhance the security of various communication systems. In essence, the project's success offers valuable insights into the practical implementation of ECC, paving the way for further advancements in secure communication technologies.

V. FUTURE ASPECTS

Performance Optimization: Optimize the implementation further to decrease computational overhead and enhance efficiency to be able to support resource-constrained devices like IoT ones. Quantum computing could even speed up cryptographic operations hence a much faster performance of IoT devices.[5]

Enhanced Security Features: New security features may include post-quantum cryptography algorithms that can protect the chat app on IoT against emerging quantum attacks.[5][9]

User Experience Improvements: In order to allow for faster processing and responsiveness, speak about making the user experience better on IoT by including features like message delivery notifications, typing indicators, and message timestamps through which communication is allowed through quantum computing.[9][10]

Cross-Platform Compatibility: To make it usable and popular among different devices (including those of IoT platforms), the application should work with most of them in various operating systems.[5]

Integration with Existing Systems: Allow for the integration of this secure chat app to be done with present messaging platforms or through enterprise systems that utilize quantum computing features to enhance the security and efficiency of IoT gadgets.

Scalability: This scalability design is one that makes use of quantum computing technology to do parallel processing hence making the application more scalable on IoT devices.

Compliance and Regulations: Ensure that the application also adheres to other laws or regulations on data protection including those regarding IoT devices on quantum safe cryptography.

Research and Development: The Application should be frequently updated with the latest advancements in quantum computing and quantum safe cryptography for IoT devices, thus we need to keep track of recent trends continuously.

User Education and Awareness: Users utilizing these devices can benefit from being informed about what quantum computers mean, therefore providing them knowledge-based resources to understand it.[9]

Continuous improvement and refinement: Gather feedback from users and stakeholders to pinpoint areas for enhancement and iterate on the application. This process includes addressing any issues or considerations related to quantum computing and IoT devices.[8][9][10]

VI. CONCLUSION

To sum up, the adoption of this technology in communication protocols of chat applications is a great advancement in enhancing privacy and security. This research method encompassed user authentication through AES and SHA-256 algorithms, shared key creation using Diffie-Hellman algorithm and message encryption / decryption by means of AES-256. Future improvements should focus on optimization of performance, strengthening security features and ensuring compatibility with new systems like quantum computers. We must steadfastly follow this path to improve secure chat application, as they are the leading companies in providing secure communication mediums. In our ever-increasing digital-and-globalized world, this progressive move is vital for meeting evolving demands made by users.

REFERENCES

[1] Shamsher Ullah, Jiangbin Zheng, Nizamud Din, Muhammad Tanveer Hussain, Farhan Ullah, Mahwish Yousaf, Elliptic Curve Cryptography; Applications, challenges, recent advances, and future trends: A comprehensive survey,Computer Science Review,Volume 47,2023,100530, ISSN1574-0137,https://doi.org/10.1016/j.cosrev.2022.100530.

[2] @misc{cryptoeprint:2015/565, author = {Craig Costello and Patrick Longa}, title = {FourQ: four-dimensional decompositions on a Q-curve over the Mersenne prime}, howpublished = {Cryptology ePrint Archive, Paper 2015/565}, year = {2015}, note = {\url{https://eprint.iacr.org/2015/565}}, url = {https://eprint.iacr.org/2015/565} }

[3] Z. Liu, P. Longa, G. C. C. F. Pereira, O. Reparaz and H. Seo, "Four$\mathbb {Q}$Q on Embedded Devices with Strong Countermeasures Against Side-Channel Attacks," in IEEE Transactions on Dependable and Secure Computing, vol. 17, no. 3, pp. 536-549, 1 May-June 2020, doi: 10.1109/TDSC.2018.2799844.

[4] @misc{cryptoeprint:2016/645, author = {Patrick Longa}, title = {FourQNEON: Faster Elliptic Curve Scalar Multiplications on ARM Processors}, howpublished = {Cryptology ePrint Archive, Paper 2016/645}, year = {2016}, note = {\url{https://eprint.iacr.org/2016/645}}, url = {https://eprint.iacr.org/2016/645} }

[5] W. Zhang, D. Lin, H. Zhang, X. Zhou and Y. Gao, "A Lightweight FourQ Primitive on ARM Cortex-M0," 2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/ 12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE), New York, NY, USA, 2018, pp. 699-704, doi: 10.1109/TrustCom/BigDataSE.2018.00102.

[6] Kenji Imamoto, Kouichi Sakurai, Design and Analysis of Diffie-Hellman-Based Key Exchange Using One-time ID by SVO Logic, Electronic Notes in Theoretical Computer Science, Volume 135, Issue 1, 2005, Pages 79-94, ISSN 1571-0661, https://doi.org/10.1016/j.entcs.2005.06.003.

[7] Lein Harn, Changlu Lin, Efficient group Diffie–Hellman key agreement protocols, Computers & Electrical Engineering, Volume 40, Issue 6, 2014, Pages 1972-1980, ISSN 0045-7906, https://doi.org/10.1016/j.compeleceng.2013.12.018.

[8] T. Melo, A. Barros, M. Antunes and L. Frazão, "An end-to-end cryptography based real-time chat," 2021 16th Iberian Conference on Information Systems and Technologies (CISTI), Chaves, Portugal, 2021, pp. 1-6, doi: 10.23919/CISTI52073.2021.9476399.

[9] K. -S. Shim, Y. -h. Kim, I. Sohn, E. Lee, K. -i. Bae and W. Lee, "Design and Validation of Quantum Key Management System for Construction of KREONET Quantum Cryptography Communication," in Journal of Web Engineering, vol. 21, no. 5, pp. 1377-1417, July 2022, doi: 10.13052/jwe1540-9589.2151.

[10] D. Henriyan, Devie Pratama Subiyanti, R. Fauzian, D. Anggraini, M. Vicky Ghani Aziz and Ary Setijadi Prihatmanto, "Design and implementation of web based real time chat interfacing server," 2016 6th International Conference on System Engineering and Technology (ICSET), Bandung, Indonesia, 2016, pp. 83-87, doi: 10.1109/ICSEngT.2016.7849628.