UPDATED October 14, 2023

Polynom™ Security Whitepaper

Code Siren, LLC ("Code Siren") has developed the world's first post-quantum cryptography (PQC) collaboration application platform that is both CNSA Suite 2.0-compliant[1] and provides complete data sovereignty for individuals and enterprises that want more control of their data and reduced vendor-dependency. The platform's secure architecture was designed for privacy-focused users and technical professionals/teams concerned with network integrity, unauthorized interception, and intellectual property theft. We have created a suite of proprietary cryptographic technologies for Polynom™ with specific applications with input from the military, law enforcement, intelligence, and hacker communities.

This whitepaper will explain Code Siren's technical ethos and demonstrate current and future technologies that threaten modern and future society. While Polynom™ was built for rugged use over dirty networks with the world's most advanced cryptography, it is also designed to be fun to use, leveraging Code Siren's gaming background.

Contents Page

[1] On September 7, 2022, the US National Security Agency ("NSA") released the Commercial National Security Algorithm Suite 2.0 ("CNSA 2.0"). See: https://media.defense.gov/2022/Sep/07/2003071834/-1/-1/0/CSA_CNSA_2.0_ALGORITHMS_.PDF

© Code Siren, LLC. All rights reserved. Unauthorized reproduction is prohibited.

Introduction to Cryptography

The first known use of cryptography dates back to 1900 BC, when an Egyptian scribe used unusual hieroglyphic symbols in an inscription not to hide the message but to make it appear more dignified. However, this inscription is the oldest known example of a text that has been transformed in some way and is considered the first use of cryptography.

Cryptography has been used by many different cultures throughout history. In India, for example, the classic work on statecraft "Arthshashtra" describes the use of "secret writing" by spies. In the Roman Empire, Julius Caesar used a simple substitution cipher to send secret messages to his generals. This cipher, the Caesar Cipher, is one of the most well-known historical ciphers.

Cryptography has evolved, and modern ciphers are much more complex than the Caesar Cipher. However, the basic principles of cryptography have remained the same.

Cryptography is a critical technology we use daily to protect our digital lives. Cryptography works in the background to secure our private data, identities, and digital footprints when we unlock our smartphones, make online purchases, start up our cars, or browse the web. Cryptography is an essential part of the security process for computing systems, and it helps to keep our information safe from unauthorized access and misuse.

Data encryption, which achieves integrity, authenticity, and confidentiality, transforms data into an unreadable format using an algorithm and a decryption key. The encryption algorithm is a mathematical formula used to scramble the data. The decryption key is a piece of information that is used to control the algorithm. This mechanism makes the encrypted data secure and only accessible to authorized users with the decryption key. End-to-end encryption ("E2EE") applies this process to all communications between two devices, from one endpoint to another. In other words, this means that even if an attacker intercepts the encrypted data, they cannot read it without the decryption key. Data encryption can protect a wide variety of data, including files, emails, and even entire hard drives.

Server-Side Encryption

In many messaging services, third parties store messages and other data. This data is encrypted only while in transit, meaning it is protected from unauthorized viewers while being sent from one device to another. However, once the data reaches the third-party server, it is stored in an unencrypted format, meaning all the messages can be viewed and stored on its servers.

End-to-End Encryption

End-to-end encryption (E2EE) is a different approach to encryption. With E2EE, the data is encrypted on the sender's device and only decrypted on the recipient's device. This means the data is never stored in an unencrypted format on third-party servers. As a result, only the sender and recipient can view the data, even if an attacker intercepts the encrypted data in transit.

E2EE is often considered more secure than server-side encryption because it prevents third-party servers from accessing the data. This can be important in cases where data privacy is of the utmost importance, such as when communicating with sensitive information or when conducting whistleblowing activities.

Asymmetric Encryption and Symmetric Encryption

There are two main types of data encryption: asymmetric and symmetric. Asymmetric encryption uses two different keys, a public key and a private key, to encrypt and decrypt data. The public key is shared with everyone, while the private key is kept secret. This type of encryption is often used for sending secure messages, as the recipient can use the public key to encrypt the message, and only the sender can use the private key to decrypt it.

© Code Siren, LLC. All rights reserved. Unauthorized reproduction is prohibited.

Symmetric encryption uses only one key to encrypt and decrypt data. This type of encryption is often used for encrypting data that must be shared with multiple people, as everyone can use the same key to encrypt and decrypt the data.

Which type of encryption is better?

Asymmetric encryption is generally considered more secure than symmetric encryption, as it is more difficult to crack. However, symmetric encryption is faster than asymmetric encryption, as it does not require exchanging two keys. The best type of encryption for a particular application will depend on the specific security needs of that application. If security is paramount, then asymmetric encryption is the best choice. However, symmetric encryption may be better if speed is more critical.

RSA is a popular algorithm that uses a mathematical operation called modular exponentiation to encrypt and decrypt data. Until recently, RSA and elliptic curve cryptography ("ECC") were considered two of the most secure encryption methods. Public key infrastructure ("PKI") is a system that manages and distributes public keys. PKI is used in various applications, including secure email, online banking, and digital signatures.

Quantum Computing

A quantum computer ("QC") is a computer that uses the principles of quantum mechanics to perform calculations and allows QCs to solve problems intractable or infeasible for classical computers across the lifetime of the Universe, such as breaking encryption codes, simulating complex molecules, and finding the optimal solutions to optimization problems. Quantum computers are still in their early stages of development, but they can potentially revolutionize many fields, including finance, medicine, and materials science.

A cryptanalytically relevant quantum computer ("CRQC") powerful enough to break public-key cryptography is possible within a few years. China and other industrialized countries are already working on developing these machines, so Western democracies and their enterprises need to start planning now. We must transition to quantum-resistant (QR) algorithms to protect national security (NSS) and our society's intellectual property. This adaptation will require careful planning and budgeting, but it is essential to do so before it is too late.

Shor's Algorithm

In 1994, Peter W. Shor developed a quantum algorithm[2] that can factor large integers in polynomial time, which is asymptotically faster than classical algorithms. Shor's algorithm enables QCs to speed up calculations needed to break specific cryptographic algorithms.

The security of our online transactions and most cryptocurrencies rests on the assumption that factoring large integers is computationally intractable. However, Shor's algorithm challenges this assumption by showing that factoring integers can be broken efficiently on a CRQC.

Shor's algorithm is arguably the most dramatic example of how the quantum computing paradigm has changed our perception of which problems should be considered tractable. In this section, we will briefly summarize some basic facts about factoring, highlight the main ingredients of Shor's algorithm, and illustrate how it works using a toy factoring problem.

[2] Peter W. Shor. Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer. https://arxiv.org/pdf/quant-ph/9508027.pdf. 1994.

© Code Siren, LLC. All rights reserved. Unauthorized reproduction is prohibited.

Grover's Algorithm

In 1996, Lov Grover published a quantum algorithm[3] that can search an unsorted database much faster than classical algorithms. The algorithm uses quantum superposition to create a state where all possible solutions are equally probable. This state is then amplified using amplitude amplification, which increases the probability of finding the desired solution. The overall effect is a quadratic speedup over classical algorithms, which can only search an unsorted database in linear time.

Grover's algorithm can be used in various applications, such as finding patterns in data, breaking encryption, and simulating physical systems. In the context of CRQCs, Grover's algorithm could find defects in products or processes more quickly and efficiently than traditional methods. For example, it could scan many products for a specific defect, such as a missing component or a manufacturing flaw. Grover's algorithm could help improve product quality and manufacturing costs by reducing the time it takes to find defects.

Quantum Attacks

Most public key ciphers, such as RSAs ("RSA" is a public-key cryptosystem from Rivest–Shamir–Adleman); DSAs (digital signature algorithms), DH (Diffie-Hellman); ECC (elliptic curve cryptography), and their variations, are vulnerable to CQRCs in a quantum attack scenario. These ciphers rely on complex problems for classical computers to solve, but CQRCs can solve them much more quickly. For example, RSA relies on the difficulty of factoring large numbers, which can be solved in polynomial time on a quantum computer using Shor's algorithm. ECC relies on the difficulty of finding the discrete logarithm of a number in a finite field, which can also be solved in polynomial time on a quantum computer using Shor's algorithm.

This means that Grover's algorithm could break symmetric ciphers with relatively short keys, such as AES-128, which has a 128-bit key. However, it would not be able to break symmetric ciphers with longer keys, such as AES-256, which has a 256-bit key.

When securing symmetric ciphers against quantum computers, it is necessary to use keys longer than the size that Grover's algorithm can search. For example, AES-256 would be secure against Grover's algorithm if used with a 256-bit key.

In recent years, IBM has significantly improved the number of qubits in its quantum computers. In 2019, the company's largest quantum computer had 27 qubits. By 2021, this had increased to 127 qubits, and in 2023, IBM unveiled a 433-qubit processor called Osprey. IBM has also announced plans to build a 1,000-qubit computer by the end of 2023.

This progress is significant because it brings IBM closer to the threshold of quantum supremacy, which is the point at which a quantum computer can perform a computation that would be impossible for a classical computer. Once IBM achieves quantum supremacy, it will be able to begin to develop quantum algorithms that can break current cryptographic systems.

However, it is essential to note that IBM's quantum computers are still in their early stages of development. The qubits in these computers are not yet as reliable as those in classical computers, and the errors that occur can make it challenging to run complex quantum algorithms. As IBM continues improving its qubits' reliability, it will get closer to the point where quantum computers can be used for practical applications.

Post-Quantum Cryptography

Post-quantum cryptography is a subset of cryptography designed to be secure against quantum computers. Some currently used cryptography is quantum-safe, meaning it is secure even if adversaries can use quantum computers.

[3] Lov K. Grover. A Fast Quantum Mechanical Algorithm for Database Search. https://arxiv.org/pdf/quant-ph/9605043.pdf. 1996

© Code Siren, LLC. All rights reserved. Unauthorized reproduction is prohibited.

However, new post-quantum algorithms (especially Latticed-based cryptography) are being developed specifically designed to be quantum-safe.

Most symmetric encryption schemes are considered quantum-safe if they use sufficiently large key sizes. The same is said for most hash functions (e.g., AES-256 bit and SHA-512 are largely considered quantum-safe[4]). However, it is essential to note that no cryptography can ever be guaranteed to be secure forever. Cryptography is constantly being tested and analyzed to make reliable assumptions about its security.

The following section will focus on the post-quantum algorithms submitted and accepted by the National Institute of Standards and Technology (NIST) standardization process[5]. We will explain why we focus on these algorithms and believe they are promising long-term candidates for post-quantum cryptography. We will also discuss the security level of these algorithms and how they can be used to implement hybrid encryption schemes.

Lattice-Based Cryptography

Lattice-based cryptography is a rapidly growing field of post-quantum cryptography. It is known for its efficiency, versatility, and heavy scrutiny. NIST has selected[6] a collection of lattice-based algorithms through the PQC Standardization Process that we believe are the most secure, stable, and quantum-proof available algorithms.

Here are some of the advantages of lattice-based cryptography:

- Speed: Lattice-based algorithms are some of the fastest post-quantum cryptographic algorithms available and are often faster than even elliptic curve cryptography, which is currently considered one of the fastest classical cryptographic algorithms.
- Reasonable key sizes: The key sizes of lattice-based algorithms are small enough to be used in standard protocols and more practical than PQC with huge key sizes.
- Heavy scrutiny: Lattice-based cryptography has been heavily scrutinized for many years, which has helped to ensure its security and has given researchers a high degree of confidence in its ability to withstand future attacks.
- Diverse uses: Lattice-based cryptography can be used to solve a diverse set of security challenges. This includes efficient constructions such as key agreements and signature schemes, as well as more elaborate constructions such as identity-based encryption ("IBE") and fully homomorphic encryption ("FHE").
- Mathematical foundation: Lattice-based cryptography has a solid mathematical foundation, which gives researchers high confidence in its security and makes it easier to analyze and verify its correctness.
- Understandability: Lattice-based cryptography is relatively easy to understand, even for those without a strong background in mathematics, which makes it a good choice for applications where security is essential, but ease of use is also a priority.

Overall, lattice-based cryptography is a promising area of post-quantum cryptography. It offers many advantages over other post-quantum cryptographic algorithms, including speed, reasonable key sizes, heavy scrutiny, diverse uses, a solid mathematical foundation, and understandability, which makes it a good choice for various applications, from everyday security to high-security applications, such as protecting government secrets and private industry's most valuable intellectual property.

[4] NIST. (2022, March 8). Post-Quantum Cryptography. Retrieved from https://csrc.nist.gov/projects/post-quantum-cryptography

[5] NIST Announces First Four Quantum-Resistant Cryptographic Algorithms. https://www.nist.gov/news-events/news/2022/07/nist-announces-first-four-quantum-resistant-cryptographic-algorithms. July 5, 2022

[6] Post-Quantum Cryptography (PQC) Selected Algorithms 2022. https://csrc.nist.gov/Projects/post-quantum-cryptography/selected-algorithms-2022. June 29, 2023.

© Code Siren, LLC. All rights reserved. Unauthorized reproduction is prohibited.

Polynom™

Polynom is our reference implementation of the Code Siren/Polynom PQC engine. It is a proof-of-concept application created by a team of game developers, white hat hackers, and cryptologists to make an engaging, fun collaboration platform founded on cutting-edge, modern, and strong cryptography. It is the convergence of our two worlds - fun collaboration and secure collaboration. Polynom is the first secure and reliable team collaboration platform to utilize the requirements of the US NSA's Commercial National Security Algorithm Suite 2.0 ("CNSA 2.0") for all encryption.

The architecture provides for a decentralized, self-hosting model that promotes data sovereignty. The CNSA 2.0-compliant collaboration platform includes AES-256 for symmetric encryption, CRYSTALS-Kyber-1024 for key establishment, SHA-512 for secure file hashing, and CRYSTALS-Dilithium5 for digital signatures.

Key features of Polynom:

Secure: All DMs, Quantum Rooms, and Social Encryption communication are E2EE using CNSA 2.0 algorithms, NIST Security Level V.

Reliable: Polynom is self-hosted, so users have complete control over their data.

Easy to use: Fun and engaging UI with intuitive design.

Compliant: FIPS 197, FIPS 180-4, FIPS 203, FIPS 204, NIST SP-208

Polynom utilizes a modular PQC engine, which is easy to use and extends throughout the collaboration ' 'app's platform. Each PQC algorithm component can be replaced as standards improve and alternative encryption systems emerge. Polynom is a robust platform that can protect data against various threats, including man-in-the-middle (MiTM) and quantum attacks. It is a valuable tool for developers and teams who must protect their data in the post-quantum era.

Polynom Clients

The desktop application for Windows, Linux, and OSX (the "Polynom Client") must be connected to the Internet. The Polynom Client allows users to create shared workspaces that bring your workflow, VoIP, and files under one PQC-secured roof into a fun and gamified environment with a modern and advanced UI/UX. The Polynom Mobile app is available on Android and iOS.

Polynom Server

Polynom Server connects Polynom Clients, allowing them to interoperate and share information from anywhere in the world. Polynom Server supports various network functions that Clients can use, such as messaging, collaboration, screen sharing, and audio/video calls in a PQC environment.

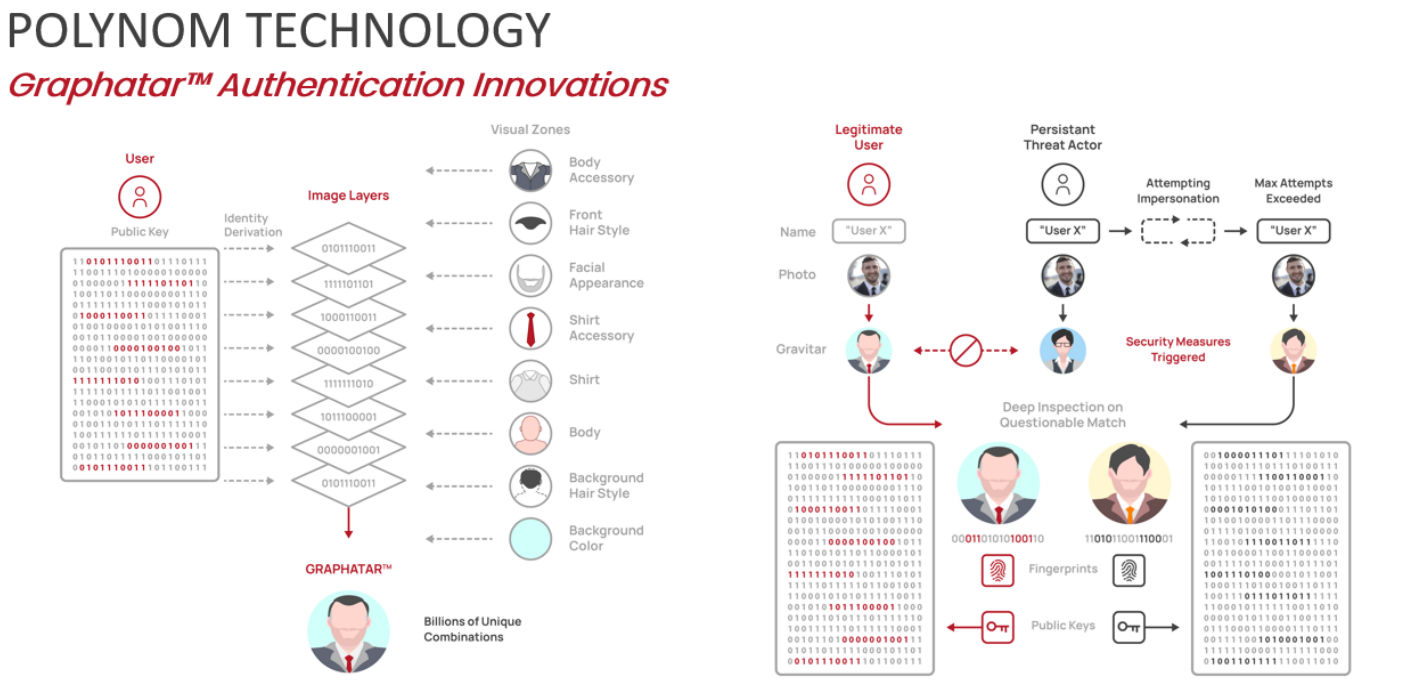
Graphatar™ and Role-Based Access Control

A Graphatar is a visual representation of the public keys of an identity. Graphatars are rendered as layered images that allow users to quickly determine if the person they are speaking to is who they think it is. Graphatars are the first line of defense against phishing and identity theft by minimizing the likelihood that a user can pretend to be another user, a vulnerability commonly found on other platforms, such as Telegram. Because there are over 15 Decillion (15,576,890,575,604,480,000,000,000,000,000,000) unique combinations, Graphatars make it mathematically very difficult to trick users into believing they are somebody else. If a Graphatar does not seem right, such as the colors or characteristics of the image do not appear as expected, further analysis can be done by examining the Graphatar's hash and the identifier. The Graphatar hash is a 64-byte SHA-512 hash of the Graphatar public keys. The Graphatar identifier

© Code Siren, LLC. All rights reserved. Unauthorized reproduction is prohibited.

is an 8-byte checksum of the 64-byte hash. These values help analyze for possible threat actors, especially when a malicious individual attempts to impersonate a legitimate user.

Since the identity is owned by the individual (and not a centralized store), every Polynom server a user connects to will render the Graphatar the same across all servers. This process allows for an identity federation across disparate servers since the same public key(s) will verify all messages sent to those servers. Graphatar technology provides for authentication and non-repudiation. Not only does this allow all users across different servers to perceive a Graphatar identically, but it also allows messages sent to different servers to be verified to have originated from the same place, i.e., the person with access to the Graphatar's private keys.



Due to the proliferation of identity theft and SIM card cloning, most collaborative platforms and communication apps are highly vulnerable. Proprietary Graphatar™ technology users to generate a distinct image close to their physical appearance. This Graphatar is a visual version of their unique public key (i.e., digital fingerprint) that minimizes the possibility of collisions (over one in ten billion).

Polynom™ allows users to catch threat actors before they can infiltrate conversations, data exchange, and other sensitive applications. Polynom's innovations in quantum-aware Graphatar™ security protocols make impersonation unfeasible and impractical even with significant user (target) information.

Hiding in Plain Sight™ (HiPS) Technology

Hiding-in-Plain Sight technology allows Polynom to operate securely (i.e., equivalent to a Type 1 Cryptographic device) on compromised networks, servers, and hostile countries. Polynom achieves this by using various techniques to disguise its traffic and make it difficult to block or intercept.

For example, Polynom disguises its traffic by adding random noise to the lengths of requests and responses, which makes it challenging to write firewall rules based on the fixed lengths of different messages. Polynom also uses varying techniques for masquerading traffic as different Internet-standard byte patterns. HiPS makes it even more difficult for inspection techniques (e.g., deep packet inspection) to identify and block Polynom traffic.

In addition to the techniques described above, Polynom uses various other techniques to protect its traffic, including PQC CRYSTALS Kyber-1024 exchanges resulting in AES 256-bit keys. These techniques make Polynom one of the most secure

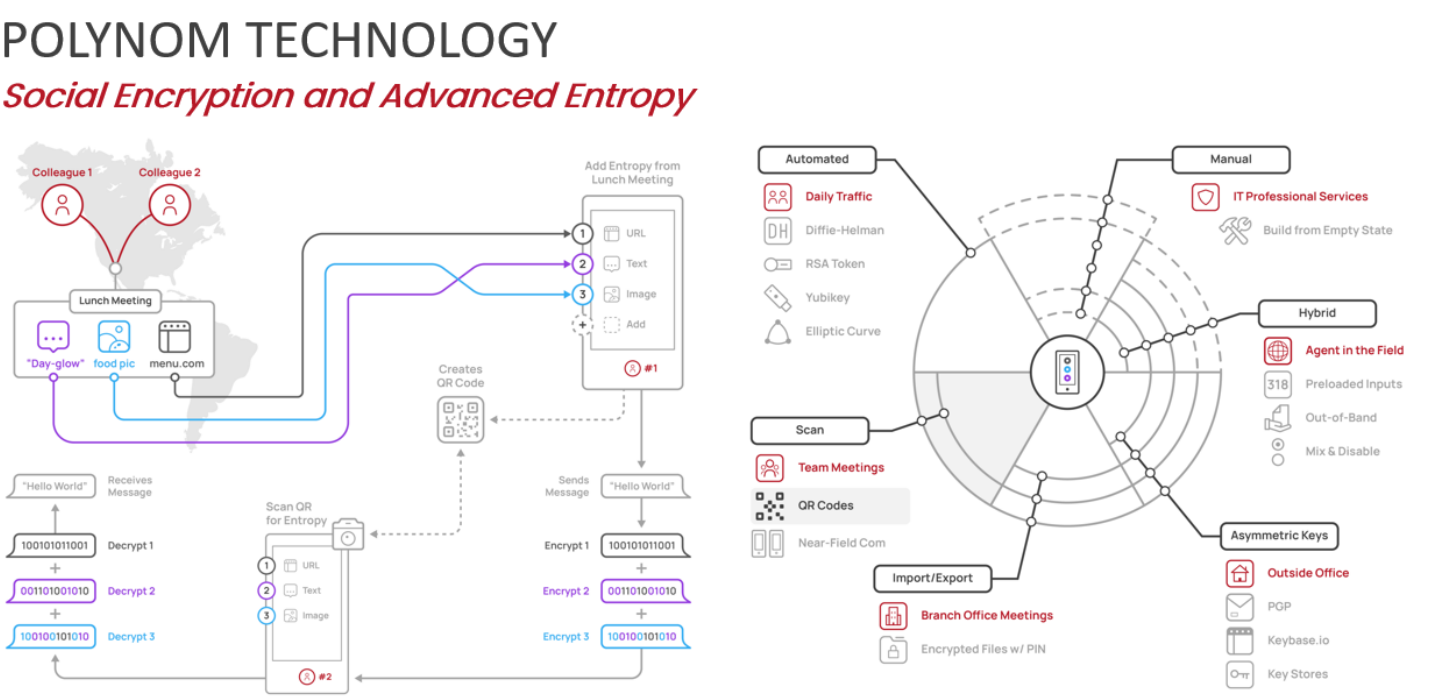
© Code Siren, LLC. All rights reserved. Unauthorized reproduction is prohibited.

communications platforms available. For further information, see: "How the Great Firewall of China Detects and Blocks Fully Encrypted Traffic."[7]

Social Encryption™ ("SE")

Social Encryption is a quantum-resistant technology that allows 1:N people to communicate E2EE across a network without manually exchanging keys. It does this by replacing network key exchanges with a symmetric key exchange from already known sources of entropy that can be acquired from anything, including passphrases, contents of files/images, or any offline/online source of data. By using shared experiences and knowledge, there is no need to perform traditional network handshakes to exchange shared keys. SE technology leaves no network trace, i.e., the network does not know that any data has been exchanged when multiple parties communicate securely in an E2EE fashion.

SE is a "rolling cipher" algorithm, meaning that each entropy line is derived from a separate source and is compounded into the final result, an AES 256 shared key. This final AES 256-bit key is used for E2EE encrypting all content between parties using AES 256-bit GCM mode. SE keys are identified across the network using an SHA-512 hash derived from an HMAC (hash-based message authentication code). The technology allows parties involved to know which SE key to use for the decryption of content without revealing any of the entropic data and other secrets used to derive the SE keys.



Social Encryption™ allows individual users to secure their communication and group collaborative efforts by adding layers of entropy using information only known to the group or subject users. Those extra layers of encryption are added to the existing cryptography for additional security.

Polynom's versatile, user-controlled cryptographic protocols provide unrivaled security in the team/channel collaborative software space. Users can configure a combination of NIST-approved AES 256-bit variant cryptography to preempt threats at the nation-state level and beyond.

Quantum Rooms™

Quantum Rooms are based on quantum entanglement. Quantum entanglement is a physical phenomenon that occurs when two particles are linked together to share the same fate, regardless of how far apart they are. In other words, if you measure the state of one particle, you will instantly know the measurement of the other particle, even if billions of miles

[7] Wu, Mingshi, et al. "Towards a Framework for Secure and Efficient Data Sharing in Mobile Edge Computing." USENIX Security Symposium. 2023. https://www.usenix.org/system/files/sec23fall-prepub-234-wu-mingshi.pdf

© Code Siren, LLC. All rights reserved. Unauthorized reproduction is prohibited.

separate them. Similarly, Quantum Rooms are workspaces only for individuals with a common SE key. Just like the principles of Quantum Entanglement, the content within QRs only exists as long as 1:n individuals view it.

For this reason, QRs are both temporal and ephemeral. The moment the last user leaves the QR, the content is permanently deleted, and the room ceases to exist. QRs merge disappearing content and private rooms without manually inviting users and/or managing authorization.

Quantum Rooms are unique in that they are never created, nor are users ever invited to join the rooms. A QR only exists if 1:n users possess an SE key and happen to have entered the room. Users who have the same SE key(s) will be able to communicate securely with each other via E2EE encryption. Since QRs are temporal, individuals can discuss matters of sensitivity and/or confidentiality without worrying about content management, such as manually setting message expiration rules as required in apps like Signal.

The content in a QR can be kept-alive, i.e., prevented from auto-deletion, by having at least one person remain in the room. QR technology allows the users to manually control content expiration by choosing to remain in the QR. The content will persist forever as long as at least one person remains in a QR.

Since a given QR is only visible to individuals with a specific SE key, deleting an SE key will make entrance into a previously accessible one impossible. They also allow individuals with duplicate SE keys to revisit the same rooms simultaneously without configuring anything within the server. Users who do not have a given SE key will have no way of knowing if any QRs even exist on the server. QRs allow someone to host a server without channels or rooms, i.e., "Empty Server"). The benefit of an empty server is that nobody can create channels or rooms, but only those with SE keys will see which QRs are available to them.

Cryptography-as-a-Service™ (CaaS)

- CaaS is an API that allows two or more computer programs or applications to communicate with each other cryptographically. It is a way for programs to share data and functionality with encryption.
- CaaS provides a library that enables core identity management features, which enable the foundational identity and access management core and generate key pairs equivalent to the Polynom app's identities.
- CaaS is the foundational implementation of PQC cryptographic algorithms and routines, which serve as the basis of identity and access management, enabling encryption, decryption, message signing and verification, Graphatar composition, etc.

Polynom Encryption Algorithms

CRYSTALS-Kyber-1024 (FIPS 203)

CRYSTALS-Kyber-1024 ("Kyber") is a type of key encapsulation mechanism ("KEM") designed to resist attacks by future quantum computers. It is one of the NIST PQC standards, which means that it has been evaluated by government, industry, and academic experts and found to be secure. Kyber was invented by a team of cryptographers led by Léo Ducas of CWI Amsterdam. The team also included Vadim Lyubashevsky of IBM Research, Peter Schwabe of Radboud University, and Gregor Seiler of IBM Research.

Kyber has several vital features. First, it is IND-CCA2 secure, which means that an attacker cannot decrypt a ciphertext even if they have access to the public key and the ciphertext and can modify the ciphertext. Second, Kyber is quantum-safe, which means it is designed to resist attacks by future quantum computers. Third, CRYSTALS-Kyber 1024 is efficient, meaning it is relatively fast and does not require much memory.

© Code Siren, LLC. All rights reserved. Unauthorized reproduction is prohibited.

Kyber can be used in a variety of applications. For example, it can establish secure communication channels between two parties, encrypt data so that it can only be decrypted by the intended recipient, or authenticate two parties to each other. CRYSTALS-Kyber-1024 is considered NIST Security Level V.

CRYSTALS-Dilithium5 (FIPS 204)

CRYSTALS-Dilithium5 ("Dilithium5") is a digital signature scheme designed to resist attacks by future quantum computers. It is one of the NIST PQC standards, which means that it has been evaluated by experts and found to be secure. Dilithium5 is considered NIST Security Level V.

Dilithium5 was invented by a team of cryptographers led by Vadim Lyubashevsky of IBM Research. The team also included Roberto Avanzi of ARM, Joppe Bos of NXP, Leo Ducas of CWI, Eike Kiltz of RUB, Tancrede Lepoint of SRI, John Schanck of the University of Waterloo, Peter Schwabe of Radboud University, Gregor Seiler of IBM Research, and Damien Stehle of ENS Lyon.

Dilithium5 has many key features. First, it is IND-CCA2 secure, meaning an attacker cannot forge a digital signature even if they can access the public key. Second, Dilithium5 is quantum-safe, which means it is designed to resist attacks by future quantum computers. Third, Dilithium5 is efficient, meaning it is relatively fast and does not require a lot of memory.

Dilithium5 can be used in a variety of applications. For example, it can be used to sign documents, verify software, or authenticate users. Dilithium5 is based on the hardness of the learning with errors (LWE) problem. The LWE problem is a mathematical algorithm used in cryptography to create secure encryption algorithms. It is based on representing secret information as a set of equations with errors. The LWE problem is also a lattice problem, which means that it is related to the problem of finding the shortest vector in a lattice. Lattice problems are widely believed to be too hard for both classical and quantum computers to solve, which makes the LWE problem a popular candidate for PQC.

Advanced Encryption Standard (AES-256)

The Advanced Encryption Standard-256 ("AES-256") is a secure and efficient encryption algorithm that uses a 256-bit key to encrypt and decrypt data. It is one of the world's most widely used encryption algorithms, and various applications, including secure communication, data encryption, file encryption, and password storage, use it.

AES-256 was invented by two Belgian cryptographers, Joan Daemen and Vincent Rijmen. They submitted their proposal for the algorithm in the AES selection process in 1997. The algorithm was named Rijndael after their names, and it was adopted as the Advanced Encryption Standard (AES) by NIST in 2001.

AES-256 is a block cipher that encrypts data in blocks of 128 bits. It is a symmetric cipher, meaning the same key is used to encrypt and decrypt data. The 256 in AES-256 refers to the length of the key used in the encryption process, not the size of the blocks. AES-256 is considered one of the most secure encryption algorithms used today. AES-256 uses a substitution-permutation network (SPN) to transform the data. The SPN is a series of rounds that perform a different data transformation. The number of rounds in AES-256 is 14, each using a different key.

The strength of AES-256 comes from the size of the key. A 256-bit key is very large, and it would take an attacker an extremely long time to brute force it. It is estimated that it would take an attacker billions of years to crack a 256-bit AES key. AES-256 is a very efficient algorithm that can be used on various devices, including computers, smartphones, and tablets. It is also a very secure algorithm, and it is considered to be one of the best choices for encrypting data.

Secure Hash Algorithm (SHA-512)

The Secure Hash Algorithm (SHA) family of hash functions was initially designed by the National Security Agency (NSA). However, the specific SHA-2 algorithms were designed by a team of cryptographers led by Dr. Ronald Rivest at MIT.

© Code Siren, LLC. All rights reserved. Unauthorized reproduction is prohibited.

SHA-2 is a revision of the SHA-1 algorithm, published in 2001. SHA-2 is a family of hash functions that includes SHA-224, SHA-256, SHA-384, and SHA-512. SHA-2 is considered to be a more secure revision of the SHA-1 algorithm. The SHA-512 cryptographic hash function produces a 512-bit hash value.

SHA-512 is a one-way function, meaning reversing the hash function to recover the original input data is impossible, which makes SHA-512 useful for various applications, including digital signatures, file authentication, message digests, and password hashing.

SHA-512 is a very secure hash function that has yet to be broken. However, it is essential to note that no hash function is completely unbreakable, and it may be possible that SHA-512 could be broken in the future.

Leighton-Micali Signature (LMS)

The Leighton-Micali Signature ("LMS") algorithm is a digital signature scheme based on the learning with errors (LWE) problem. It is one of the NIST PQC standards. LMS was invented by Tom Leighton and Silvio Micali in 1992. It is a digital signature scheme based on the Learning with Errors (LWE) problem.

Leighton is a computer scientist and electrical engineer known for his work in cryptography and computer networks. He is currently a professor at MIT. Micali is a computer scientist and cryptographer known for his work on secure multi-party computation and zero-knowledge proofs. He is currently a professor at the Weizmann Institute of Science in Israel.

The LMS algorithm was first published in a paper titled "A New Paradigm for Digital Signatures" in the Journal of Cryptology. The paper was co-authored by Leighton and Micali, as well as by Shafi Goldwasser and Charles Rackoff.

The LMS algorithm is a one-time signature scheme, meaning a new signature must be generated for each signed message. The signer generates the signature, which uses the private key to create a signature verified by the recipient using the signer's public key. The LMS algorithm is efficient, relatively fast, and does not require much memory. It is also secure, meaning it is believed to resist cryptanalytic attacks.

Key Storage

Polynom Client

The only keys that are stored for Polynom Client (for Windows, Linux, Android, iOS, OSX) are the CRYSTALS-Dilithium5 ("Dilithium5") key pairs that are used for identity. Each identity has a single Dilithium5 key pair generated and stored locally in the SQLite database. The local database is encrypted using an AES 256-bit variant in GCM mode.

Polynom Server (Linux)

A single Dilithium key pair is created for the server and stored in the SQLite database (Community edition) or the MariaDB (Pro & Enterprise editions). The local database is encrypted using AES 256-bit in GCM mode.

Polynom Gateway (Linux)

A single Dilithium key pair is created for the gateway and stored in the SQLite database. The local database is encrypted using AES 256-bit a GCM mode.

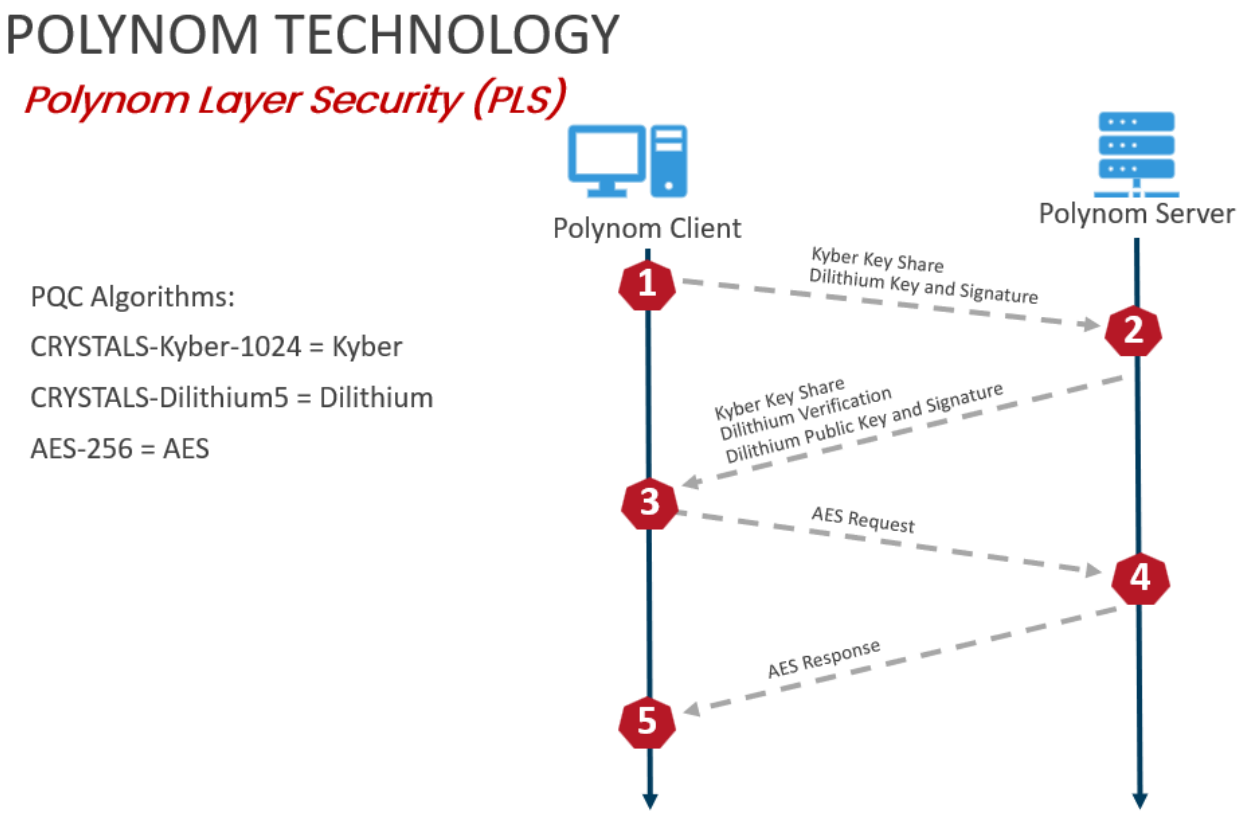
© Code Siren, LLC. All rights reserved. Unauthorized reproduction is prohibited.

Threat Model

The security goals of the Polynom platform are:

- Neither Code Siren, LLC nor any of the Polynom servers you connect will ever collect personally identifiable information ("PII").
- Protect messages' authenticity, confidentiality, and integrity while in transit.
- Protect the authenticity, confidentiality, and integrity of messages while in storage.
- Guarantee non-repudiation of all messages sent by a given user
- Verify the identity of the users on the network.

Polynom Layer Security (PLS)



Protocol

PQC Handshake

The Polynom Client performs a "PQC Handshake" before communicating with the Polynom Server. A PQC Handshake involves a PQC Handshake request and a PQC Handshake response. The purpose of the handshake is to derive a shared AES 256-bit key via CRYSTALS-Kyber-1024 KEM. The originating party's CRYSTALS-Dilithium5 ("Dilithium5") private key signs the handshake request and response messages. The Dilithium5 signature is verified at the receiving end for authenticity, integrity, and non-repudiation. A Polynom Client can further choose to trust a Polynom Server or only communicate with a Polynom Server only if the SHA-512 hash of its public keys matches what the Client was expecting beforehand, thereby avoiding the potential for a man-in-the-middle ("MiTM") attack.

© Code Siren, LLC. All rights reserved. Unauthorized reproduction is prohibited.

Sending a Message/Request

Once a PQC Handshake has been completed, the Client can begin communicating with the server by sending any requests. The request is encrypted with AES 256-bit in a GCM mode to ensure the authenticity and confidentiality of data.

Receiving a Message/Request

Once a PQC Handshake has been completed, the Client can communicate with the server. The server will send optional responses back to the Client. The responses are encrypted with AES 256-bit in GCM mode to ensure the authenticity and confidentiality of data.

Signature

The originating party's Dilithium5 private key signs the PQC Handshake request and response messages. The Dilithium5 signature is verified at the receiving end for authenticity, integrity, and non-repudiation.

© Code Siren, LLC. All rights reserved. Unauthorized reproduction is prohibited.