Scientific Research Publishing

Journal of Information Security, 2025, 16(4), 528-543
https://www.scirp.org/journal/jis
ISSN Online: 2153-1242
ISSN Print: 2153-1234

Entangled Cyclical Encryption Architecture: The Paradigm Cipher for a Fractured World

Jacob Koch

Perfect Computing Solutions, Inc., Madison County, IL, USA
Email: jako@easypcs.us

How to cite this paper: Koch, J. (2025) Entangled Cyclical Encryption Architecture: The Paradigm Cipher for a Fractured World. *Journal of Information Security*, **16**, 528-543.
https://doi.org/10.4236/jis.2025.164027

Received: September 22, 2025
Accepted: October 21, 2025
Published: October 24, 2025

Abstract

The accelerating pace of quantum computing has drastically shortened the expected timeline for breaking classical encryption—surpassing predictions by a factor of 500 or more. Lattice-based encryption, once considered a post-quantum solution, is now increasingly vulnerable due to its susceptibility to key vector mapping, high-level dictionary attacks, rainbow table optimizations that exploit weaknesses in static key structures, as well as misestimated quantum capabilities. The Entangled Cyclical Encryption Architecture (ECEA) introduces a paradigm shift: a 12-phase, self-evolving encryption architecture where the key and data are codependent and transform recursively. Rather than treating the key as a static unlock mechanism, this method uses the longer of the key, or the data to encrypt and reshape the data 1000's of times, producing emergent, entangled ciphertext that cannot be decrypted without fully reversing the entire cycle. Additionally, the cyclical nature of the process wraps around the existing data, minimizing extra storage requirements and keeping the encrypted output close in size to the original. This adaptive structure eliminates common cryptanalytic attack vectors, ensures resistance against quantum-based brute force and pattern-seeking algorithms, and future-proofs data protection by enabling dynamic complexity growth alongside computing advancements. 3 new algorithms have been created that utilize aspects of this technology. Unlike traditional methods, no predictable rules, reduction paths, or reusable structures exist, rendering all 3 ECE (Entangled Cyclical Encryption) algorithms fundamentally resistant to both classical and quantum decryption attack strategies. This methodology also allows the encryption architecture to expand with technology for decades.

Keywords

Post Quantum Encryption, AI, Quantum Computing, Encryption, Cybersecurity, Cellular Security, Encryption Architecture

1. Introduction

The threat landscape has evolved: quantum computing, advanced AI codebreakers, and distributed intrusion methods now challenge even the strongest encryption standards. Entangled Cyclical Encryption (ECE) is a next-generation encryption architecture that fuses the concepts of key/data co-dependency, cyclically transformative phases, and multi-layer encryption.

Unlike static encryption like AES-256 or RSA, ECE leverages both the data and key as symbiotic agents in a 12-phase recursive encryption cycle. Each phase reshapes both entities, producing emergent, non-reversible ciphertext embedded in dynamically entangled frameworks. The encryption has to be unwound before the data can be decrypted.

Recent public challenge stress testing had over 3000 cryptographic enthusiasts and students from colleges like Harvard and MIT, of which none of the participants could break even the early-stage implementations of ECE. Unlike static encryption, ECE creates emergent, non-reversible ciphertext through entangled, shifting structures. Decryption is impossible without traversing the full encryption cycle in reverse—phase by phase. Designed for post-quantum security, ECE represents a major evolution in information defense, aiming to future-proof data for the next 150 years.

2. Manifesto: A Cipher for a Fractured World

2.1. The Current State

Data breaches. State surveillance. Post-quantum uncertainty. The encryption protocols of yesterday are bleeding at the edges—and worse, they're predictable. The structures that once secured our secrets are being unraveled by the very architectures that empowered them. We don't need more fixes and patches; we need a paradigm. *Entangled Cyclical Encryption Architecture* (*ECEA*) is that paradigm—a multi-phase, key-driven, dynamically entangled encryption framework that doesn't merely conceal data. It embeds data into cycles of data transformation.

This is not just encryption. It's information architecture as defense.

ECEA is born from the belief that encryption must no longer be passive.

It must be entangled, self-evolving, and mathematically fluid.

This architecture does not obscure information. It embeds it in motion.

2.2. The Problem

The pace of advancement in quantum and post-quantum computing has been nothing short of remarkable. A decade ago, it was widely believed that breaking modern encryption algorithms would take millions of years. Today, with the advent of quantum computing, some of those same encryption schemes can be compromised in a matter of months—and more likely, days.

Lattice-based cryptography, often touted as a cornerstone of post-quantum security, is not the impenetrable fortress many assume it to be. While it does offer resistance to conventional quantum attacks, the structure of lattice-based encryp-

tion—its high-dimensional data matrices—can reveal patterns or heuristics that reduce the complexity of an attack. Once an approximate location within the lattice is identified, the number of potential solution vectors drops significantly. From there, reverse-engineering a key becomes a far less computationally intensive task than initially estimated.

With quantum systems already achieving computational outputs equivalent to billions of years of classical computation in mere minutes, traditional assumptions about cryptographic time horizons—such as AES or RSA holding strong for millennia—are no longer reliable. What once required theoretical eons may now be reduced to hours or days. The landscape of cybersecurity is shifting rapidly, and so must our assumptions and defenses. With quantum systems already achieving computational outputs equivalent to billions of years of classical computation in mere minutes, traditional assumptions about cryptographic time horizons—such as AES or RSA holding strong for millennia—are no longer reliable. What once required theoretical eons may now be reduced to hours or days. The landscape of cybersecurity is shifting rapidly, and so must our assumptions and defenses.

2.3. Our Solution

Instead of data being transformed by the key, the data and key interweave over 12 transformational phases with potentially 1000's of cycles to protect data. The longer of the two (key or data) drives the process. Every bit in each phase modifies the next. Each phase reshapes both entities—producing a final cipher that is uniquely shaped by its own construction. The data changes as different parts of the key is used, as demonstrated in **Figure 7**. Encryption can no longer be static. Evolution has become more important than ever due to the miscalculations of quantum computing capabilities in the past. Privacy controls and cloud security are becoming significant concerns for many reasons such as separation of key and user or more advanced side channel attacks. There are still some concerns about how secure quantum LWE and RING-LWE actually are [1] [2].

3. Breaking the Assumptions: The Whole Picture

The times of a single static transformation are gone. It is weak and vulnerable.

3.1. Existing Cryptographic System

- Rely on static keys or key hierarchies.
- Do not evolve as encryption progresses.
- Fail to scale dynamically with data length or complexity.
- Are not inherently quantum-resilient.
- Use a combination of public and private keys to come up with the transformation factors/vectors.

3.2. Entangled Cyclical Encryption Architecture (ECEA)

ECEA flips the existing model on its head. Emerged from a simple observation:

no existing system fully uses both the key and data as entangled cryptographic agents. In most systems, one dominates. With increasing quantum capabilities on the horizon, even robust schemes like RSA and ECC show surface cracks under lattice-based assaults.

What if encryption wasn't static—but cyclical?

- What if each stage of encryption evolved based on the output of the last?
- What if the structure of the key and data formed a living vector in a dynamic lattice?

ECEA was born not as an improvement—but as a total transformation to an antiquated, breaking down system. Many of the encryption systems today have been in existence for decades. There are comparisons between existing systems and ECEA as demonstrated in Figures 1-4. Recently, advancements have been made to make the existing systems capable of withstanding attacks. Our system was conceptualized in part by AI, to make any kind of attack irrelevant. Multiple AI Engines had the same cyclical mechanism for defense. We built it for the future of security.

AI Assessment of Entangled Cyclical Encryption Architecture		
Attack Type	**Effectiveness vs ECE**	**Notes**
Shor's / Grover's	⃠ Ineffective	ECE avoids vulnerable math types
AI/ML-guided attacks	⃠ Ineffective	No consistent pattern to learn
Heuristic optimization	⃠ Ineffective	No fitness feedback or approximable terrain
Rainbow/dictionary/vectors	⃠ Ineffective	No reuse or structure to target

Figure 1. AI Assessment of ECE weaknesses (ChatGPT).

Algorithmic Attack Effectiveness on Existing Encryption Types				
Attack Method	**RSA / ECC**	**AES / Symmetric**	**Lattice-Based Crypto**	**Notes**
Shor's Algorithm	Breaks it completely	Not applicable	*Emerging threat*	Cracks factoring/log-based systems instantly with sufficient quantum qubits.
Grover's Algorithm	*Halves key strength*	*Reduces AES-256 to 128-bit*	*Could aid lattice search*	Doubles brute-force speed; significant for small or medium key sizes.
Machine Learning (ML)	*Can infer usage patterns*	*Can aid side-channel attacks*	*Helps lattice vector prediction*	Not a direct break, but speeds up attack prep and vector guessing.
Genetic Algorithms	Not effective alone	*Possible in side-channel key recovery*	*May narrow lattice search space*	Useful in approximations when clues or feedback loops are available.
Rainbow Tables	*Effective on poor RSA hashing*	*Deadly to unsalted hashes*	Not useful	Precomputed attacks rely on deterministic hash outputs.
Dictionary Attacks	*Common on password-encrypted keys*	*Effective on weak passphrases*	*Potential if patterns exist*	Easy to apply where human-generated inputs are used.
Side-Channel Attacks	*Very effective in real-world*	*Proven with power/timing analysis*	*Still under research*	Physical-layer exploits bypass math entirely.
Classical Brute Force	Impractical for RSA >2048-bit	Impractical for AES-256	Infeasible for strong lattice sizes	Still used on weak implementations or stolen hash lists.

Figure 2. Algorithm attack effectiveness (ChatGPT).

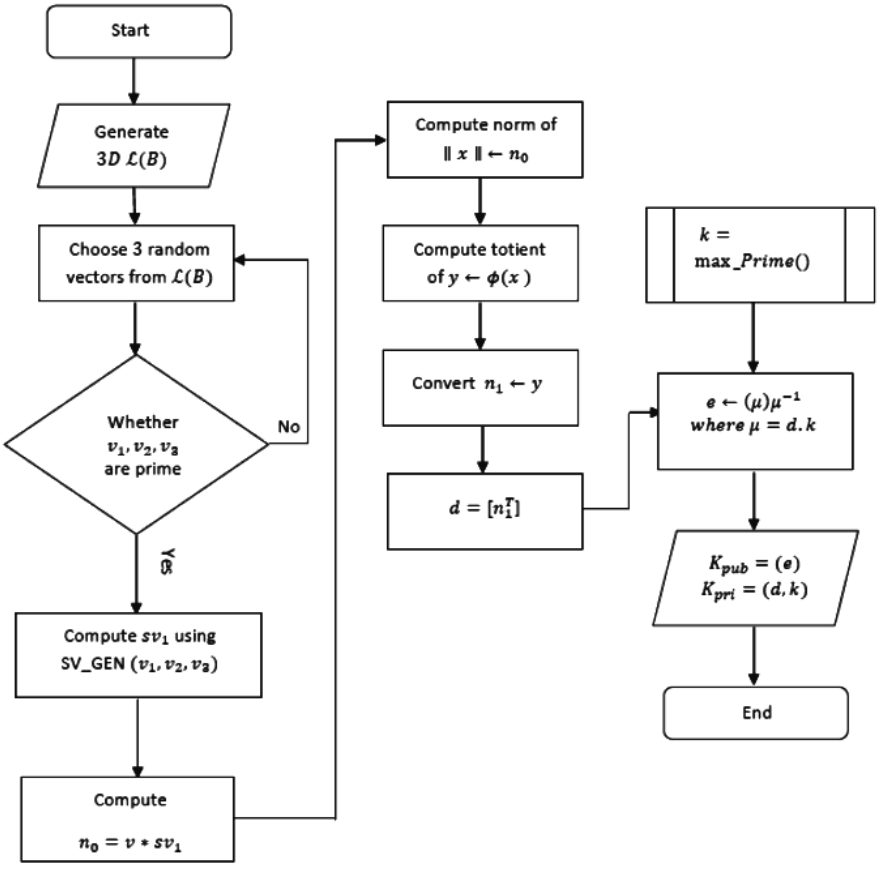

Figure 3. Lattice RSA based encryption data flowchart ([1] Google Search).

Estimated ECE Encryption Times for 250 MB (Baseline = 4 minutes)

System Class	Example CPU / Specs	AES-NI / Acceleration	Speed Gain vs You	Estimated Time
TEST Laptop	2.5 GHz Dual/Quad Core, 16 GB RAM	✗ Possibly No AES-NI	1× (baseline)	**4 minutes (240 sec)**
High-End Laptop	i7-12700H / Ryzen 7 7840HS	✓ Yes	5×–10× faster	**24–48 seconds**
High-End Desktop	i9-14900K / Ryzen 9 7950X	✓ Full AES-NI	15×–25× faster	**10–16 seconds**
Server / Workstation	AMD EPYC / Threadripper 64-core	✓✓ Multi-core + SIMD	30×–50× faster	**5–8 seconds**
GPU-Accelerated	CUDA/OpenCL + AES-optimized libs	⚠ Depends on compatibility	50×–100× faster	**2.5–4.8 seconds**

Figure 4. Benchmark comparison between computers (ChatGPT).

3.3. Architecture Overview

Components

- Key/Data Resolver: Determines dominant input stream.
- Phase Engine (Variable depending on Algorithm selection): Recursive trans-

form phases applying:

- Bit-phase rotation—individual bit manipulation in every phase.
- Riffle scrambling—Algorithm 3 permutations quickly flipping some of characters around.
- XOR transformations.
- Some phases use a reversed key or reversed data, or both.
- Entropy tunneling—reflects how difficult it is to infer or predict any component of the encrypted data. Tunneling refers to burying transformations within layers of other transformations. This is described in more detail in Section 4.1

- Entanglement Mechanism: Uses the key to encrypt the data by cycling through the data (or the key) to the end of the object file. It performs all twelve phases with each phase completely cycling to the end of whichever is longer encrypting the object file more each time.
- Symmetric Key: At this time, only symmetric keys are being used, but the ability to use asymmetric keys could be utilized.
- No limit to bit strength: Unlike normal encryption methods which require an 8-bit base and are limited to a maximum of 512 characters (AES-256 is 32 characters), there is no static requirement like that. A key of any size can be used. The strongest encryption performed has been 10 million bits or 1.25 million characters.
- Scalable with Technology: Also differing from standard encryption architectures, the ECE architecture grows and adapts with technology. It adapts to stronger technology. It literally grows with our innovations. As an added bonus, the cyclical nature of the encryption keeps file close to the original file's size, saving immeasurable petabytes of storage space.
- Distributed storage.
- Obfuscation of vector boundaries.
- Redundant reassembly logic.

This is a whole new architecture like some of the encryption mechanisms were in their early stages. Many improvements and mods can be utilized with this technology, including lattice encryption using ECEA methodologies. Although untested, this could be applied to ECEA.

3.4. Architecture Additions

While the ECEA encryption system is in its infancy, it is stronger than other encryption systems; some of the things used to make antiquated encryption architecture more viable could be applied. This would increase the complexity and security of the encrypted data.

3.5. What the Future Looks Like

Modern attacks are no longer linear or brute-force alone—they're smart, multi-dimensional, and adaptive. Traditional encryption relies on fixed inputs and pre-

dictable structures, making them increasingly vulnerable in an era of:

- Distributed computing networks (e.g., botnets or GPU farms).
- Accelerated learning loops (AI iteratively training on encryption patterns).
- Quantum supremacy (real-time vector discovery in lattice systems).
- Vector Mapping and key shortcuts.

3.6. Current Encryption/ECEA Comparison Attack Theories

A. Quantum Algorithms (Shor's, Grover's)

Threat to static encryption like RSA, ECC, AES

- Shor's targets factor-based systems (not relevant to ECE).
- Grover's speeds up brute-force key search, but since this encryption evolves per data/key and isn't directly reliant on symmetric key patterns, Grover's impact is severely limited.
- ECE's recursive entanglement means even having the key doesn't provide immediate decryption—Grover's can't shortcut the 12-phase evolution.

B. Machine Learning & AI-Guided Attacks

Threat to ciphers with patterns, repeatability, or key leakage

- AI needs repetition and statistical structure to learn patterns.
- Since ECE's ciphertext is emergent, non-repeating, and fully dependent on specific input + key evolution, there's no stable pattern to train on.
- This system's lack of reversible transformations and deterministic output makes it a moving target for ML-based models.

C. Heuristic Optimization (Genetic, Simulated Annealing, etc.)

Threat to systems with known bounds, approximable states, or scoreable fitness

- These methods require feedback loops (fitness functions) and solvable search spaces.
- ECE's encryption, if properly constructed, has no smooth gradient or scoring path to follow.
- If each phase transforms data without a consistent traceable trail, these algorithms become effectively blind.

D. Dictionary, Rainbow Table, Vector Libraries

Threat to systems with known hash behavior or lattice structure

- These only work on predictable hash functions or fixed key behaviors.
- ECE uses key/data co-dependency and recursive evolution—meaning every encryption path is unique.
- If even the same input + key yields different outputs due to evolving state or external entropy, precomputed tables are useless.

3.7. Use Cases for ECEA

Transforming digital security across critical global sectors.

A. National Defense & Intelligence

Military communications and classified intelligence rely on absolute data integrity. ECE's non-reversible, multi-phase encryption ensures that even intercepted

files remain inaccessible, resisting quantum decryption and cyber-espionage.

B. Financial Services

Banks, fintechs, and payment processors handle vast volumes of sensitive data. ECE provides robust protection against both traditional and quantum attacks, eliminating the risk of credential replay, transaction tampering, and identity theft.

C. Healthcare & Genomics

Patient records and genetic data are among the most personal—and valuable—forms of information. ECE ensures HIPAA-compliant, end-to-end encryption with self-evolving protection that adapts as medical technologies advance.

D. Cloud Infrastructure & SaaS

Cloud-hosted applications and storage systems are prime targets for breach. ECE's compact ciphertext and data-key interdependence minimize storage overhead while eliminating common attack vectors, making it ideal for multitenant, scalable environments.

E. Critical Infrastructure (Energy, Water, Transportation)

Nationwide systems must defend against cyber-sabotage. ECE's deterministic resistance to known attacks and AI-enhanced resilience make it suitable for embedded systems in grid management, logistics, and public safety.

F. Cryptocurrency & Blockchain

Private keys are the Achilles' heel of most blockchain platforms. ECE introduces encryption that binds keys and transaction data into recursive, obfuscated structures, deterring theft and enhancing the cryptographic layer beyond standard elliptic curves.

G. Artificial Intelligence Systems

As AI models become proprietary and strategic assets, protecting both model weights and inference data is critical. ECE safeguards training data, internal weights, and decision logs, even in federated learning environments.

H. Autonomous Vehicles & IoT

Connected devices must verify identity and exchange trusted information in real-time. ECE enables lightweight, recursive encryption models that support constrained hardware without sacrificing future-proof protection.

I. Telecom & Mobile Networks

Mobile carriers and telecom providers operate vast, high-speed networks transmitting voice, video, and data. ECE's minimal overhead and adaptive encryption cycles are optimized for dynamic bandwidth conditions, securing 5G and edge communications without latency spikes.

J. Legal, Compliance & Digital Forensics

Law firms, regulatory bodies, and e-discovery platforms handle sensitive and legally binding information. ECE provides tamper-evident encryption that protects case files and audit trails—even under court-mandated data sharing.

4. Figures and Details

4.1. Architecture's Inner Workings

The Entangled Cyclical Encryption Architecture (ECEA) is a next-generation

cryptographic framework designed to eliminate predictability, disrupt pattern recognition, and render techniques like rainbow table attacks obsolete. At its core, ECEA employs two distinct algorithms that leverage engineered manipulations in different sequences and complexities—each offering robust, built-in resistance to quantum attacks. This resilience comes from ECEA's unique entangled cyclical methodology, which fosters high entropy—a cryptographic measure of randomness and unpredictability. In this context, entropy reflects how difficult it is to infer or predict any component of the encrypted data, making unauthorized decryption exponentially more complex.

A key innovation within ECEA is entropy tunneling: a process of deeply embedding encrypted data across multiple transformation layers, each influenced by independent entropy sources or key fragments. Even if one layer is compromised, the deeper layers remain secure and undecipherable, maintaining the integrity of the entire structure, where as other mechanisms such as Non-linear Indeterminate Equations and other lattice-based reduction methods yielded the public keys and decryptable functionality of 100 keys in only 40 seconds using Gentry's Attack which deals with homomorphic mechanisms for usage without actual decrypting the file, as it decrypts into memory for usage [3]. ECEA is also designed to integrate with advanced techniques such as Layered Obfuscation, where entropy tunneling enhances concealment by ensuring that each obfuscation layer is independently secured and insulated from reverse analysis. This also makes side-channel attacks like on NTRU, useless as there is no methodical way to reduce the inherent relationship between the key and the applied vectors nor can it be reduced by 50% on existing systems [4].

Together, these innovations position ECEA as a cutting-edge foundation for future-proof encryption standards. Designed to combat any adaptive AI threat. This architecture has no patterns or predictable methods. This essentially makes it impervious to any kind of AI attack as discussed in some recent works covering Adaptive attacks with AI [5]. The entanglement is accomplished using 12 phases of modification. After the user selects the folder/files to be encrypted, the byte data is formulated into byte arrays for modifications. After the data has been configured for use, the determination of which is longer (data or key) has to be completed. The longer of the two is the determinant in the number of cycles. The cycles are determined by the number of iterations it takes to fully complete the longer set of data. Each iteration uses the shorter of the two parts (data or key) to manipulate a portion of the longer part. The process does this for each phase of the encryption as the shorter of the two parts is used to manipulate the longer. It can also be used over a network where both users have the private key [6].

Quantum computers are here and with the recent in-depth understanding of quantum entanglement and developments in the quantum realm, the race is on for something better [7]. When a quantum computer meets existing cybersecurity, many things will falter as the technology greatly surpasses existing possibilities [8]. With the advent of commercial computing from top companies, the enter-

prise relationship between data security and protecting the data has never been more crucial [9].

The first phase creates a byte array from the key file so that bytes can be used against bytes. It then performs the XOR function on the data character value compared to the key character value. The resulting answer is the adjusted byte values for each byte of the data. for each byte of the data. This cycled as many times as required to fully utilize the shorter of the two pieces to manipulate the longer data. The fully completed data packet has now been encrypted multiple times. This encrypted data packet is the starting point for phase 2 to begin.

Phase 2 utilizes rotations of the encrypted data. The rotations are kept anchored to ASCII values by using mod 65535. When the value ends up being above 65535, 256 is subtracted from it until it is under 65535. This process is used to rotate the value using the byte information for the data and the key. The calculation uses the value of each to make the rotation. This process is cycled as many times as it takes to fully utilize the shorter data to manipulate the longer data. The fully completed encrypted packet from this phase is used to start phase 3.

Phase 3 utilizes the XOR function again on the recently passed Phase 2 data packet. Cycling through data and having the resulting encrypted data byte/character value of the data added to byte/character value for the key. It repeats this until all of the cycles have completed. The resulting encrypted packet is then passed into Phase 4.

Phase 4 is the area where the XOR function is used and the resulting values are then added to the previously adjusted data byte values to come up with the rotation. The rotation must be tied to ASCII via mod 65535. The resulting value must have 256 subtracted from it until it is under 65535. The resulting character/byte value is then appended to the final string that is passed to Phase 5 upon completion.

In Phase 5, the previously encrypted data packet is once again modified with XOR function on the new data from Phase 4. The resulting answer is appended to the final packet that gets passed into Phase 6.

Phase 6 utilizes addition to add the byte values from each part of the data and the key together. However, the values being passed from Phase 5 are reversed and the following steps happen on the reversed data/byte values. When the end of the shorter data piece is reached, it starts over adding the starting byte value of the shorter data piece to the next byte data piece of the longer data piece.

Phases 7, 8, and 9 are all repeats of the previous phases with different options such as reversed data and keys. The remaining phases are reversing the byte array, converting to base 64 data and saving the data to a file.

Data reversals and key reversals happen at designated parts of the process. Sometimes it is only data that is reversed, sometimes just the key. They can also happen together. Phases 1 through 3 are repeated two more times with varying arrangements of reversed data, reversed keys or both.

The stronger algorithm uses phases 1 through 3, three times. The easier (but

still inherently quantum attack-proof) algorithm only does the process once, and is therefore 33% as secure as one using all 12 phases. Salt can be implemented to sprinkle disinformation throughout the data and the locations are metadata in the encrypted byte information. After all is complete whether algorithm one or two, salt or unsalted, the data is transformed into byte 64 data. This converted data/byte value is the final encrypted product and is the starting point for the decryption process.

The decryption process works the exact opposite of the encryption process, using XOR functions and character rotations with mixed reversed data and reversed key through the process yields the unencrypted data in its decrypted form.

This process allows for easier faster encryptions where the key and the data are close to the same length. A ratio where they are close to equal is very weak and only allows for one cycle to be completed in each phase. This is very weak and not recommended. The additional capacity for well above one-million-bit strength keys comes from the cyclical nature. The more cycles, the more times the process wraps the data with each cycle being another layer of encryption. More cycles equal a deeper entropy tunnel, whereas fewer cycles make shallower, weaker entropy.

4.2. The Future and New Classifications Required

This methodology can grow with technology for hundreds of years with additional pieces being added throughout time. Using a Ratio of 1:1 being the weakest and 10,000:1 being close to the highest point necessary for Tier 4 quantum security, the number of cycles is directly related to strength of the encryption. **Figure 6** shows initial categories of these Tiers in Section 4.3.

All emerging standards should adopt this methodology through a structured and regulated framework, ensuring responsible growth as it evolves into a foundational technology.

The third algorithm uses permutation to transform parts of the data, converts the data to byte arrays and then proceeds to generate a key by using the square root of the data length and the square root of the key length. The Riffle Key is an integer that is calculated by taking the square root of the text length/square root of the key length. If the character is prime, the letter used is r. If the Riffle Integer is not prime, the letter used is b. The permutation happens to every other letter. Once the permutation cycle has completed, the shorter of the data pieces (the data or the key) is cycled during the rotation mechanism where the byte value of the data is added to the byte value of the key and then anchored into the range under 65,535. If the rotated value is above 65,535, 256 is subtracted until the calculation is below 65,535. This happens as many times as it takes to reach the end of the longer part. The only manipulation performed in this algorithm aside from the permutation is the rotation as previously described. Additionally, this method uses a reversed key for the calculations. The benchmark times for this new architecture are shown in **Figure 5**. The recommended Tier system is illustrated in **Fig-**

ure 6. An example of a key being cycled around a smaller dataset 6 times is demonstrated in Figure 7. Figure 8 demonstrates the flowchart of the Shufflemaster algorithm. This technology is patent pending and excerpts from the patent can be found in Figure 9 and Figure 10. The quantum-resistant comparisons between different types can be found in Figure 11.

4.3. Internal Calculation Results and Data

FILE TIMES (TEXT)								
Bit Strength	Char Count - Key	File Type	Total Time (h:m:s)	Start File size	Encrypted File Size (BYTES)	Selected Algorithm	Cycle Count	Iteration Time (seconds) Avg
4096	512.00	TXT	0:11:00	140,748.00	142MB	ShuffleMaster	275	0:00:02
1,000,000	125,000.00	TXT	0:15:00	830 KB	840 KB		6.6	0:02:16
8192	1,024.00	TXT	0:01:00	32KB	36KB		3.9	0:00:15
4096	512	TXT	0:07:35	400 kb	410 KB		781	0:00:01
8192	1,024.00	TXT	0:02:00	64 kb	70 KB		7.81	0:00:15
FOLDER TIMES (BYTE)								
4096			0:00:50	13 MB				
4096			0:08:00	250 mb			488.00	
4096	512.00	70 FILES 15MB +	1:00:00	1.05 GB	1,137,111,332	ShuffleMaster	2,200,000.00	0:00:00
8192	1,024.00	TXT	0:30:00	1.05 GB				#DIV/0!
4096	512	200 FILES	0:38:00	1.05 GB		Twexty Tango	2,200,000.00	0:00:00
8192	1,024.00	TXT	0:18:00	1.05 GB				#DIV/0!
1477824	184,728.00	70 files	0:00:35	1.05 GB		Twexty Tango		#DIV/0!

Figure 5. Benchmark times.

Tier	Cycle Range	Security Classification
Tier 0	≤ 200 cycles	Basic Security
Tier 1	201 – 1,000 cycles	Low-Mid Security
Tier 2	1,001 – 2,000 cycles	Moderate Security
Tier 3	2,001 – 5,000 cycles	Medium Security
Tier 4	5,001 – 10,000 cycles	Mid-High Security
Tier 5	10,001 – 20,000 cycles	High Security
Tier 6	20,001 – 50,000 cycles	Very High Security
Tier 7	50,001 – 70,000 cycles	Advanced Security
Tier 8	70,001 – 90,000 cycles	High-Extreme Security
Tier 9	90,001 – 100,000 cycles	Near-Ultimate Security
Tier 10	> 100,000 cycles	Ultimate Security

Figure 6. Tier classification.

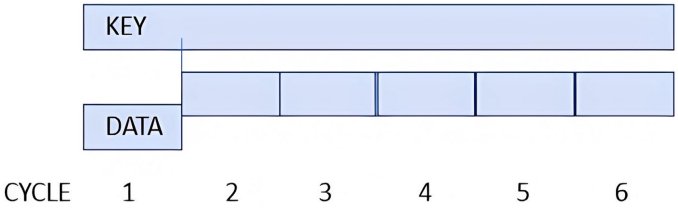

Figure 7. A basic cycle overview where the key wraps around the data 6 times.

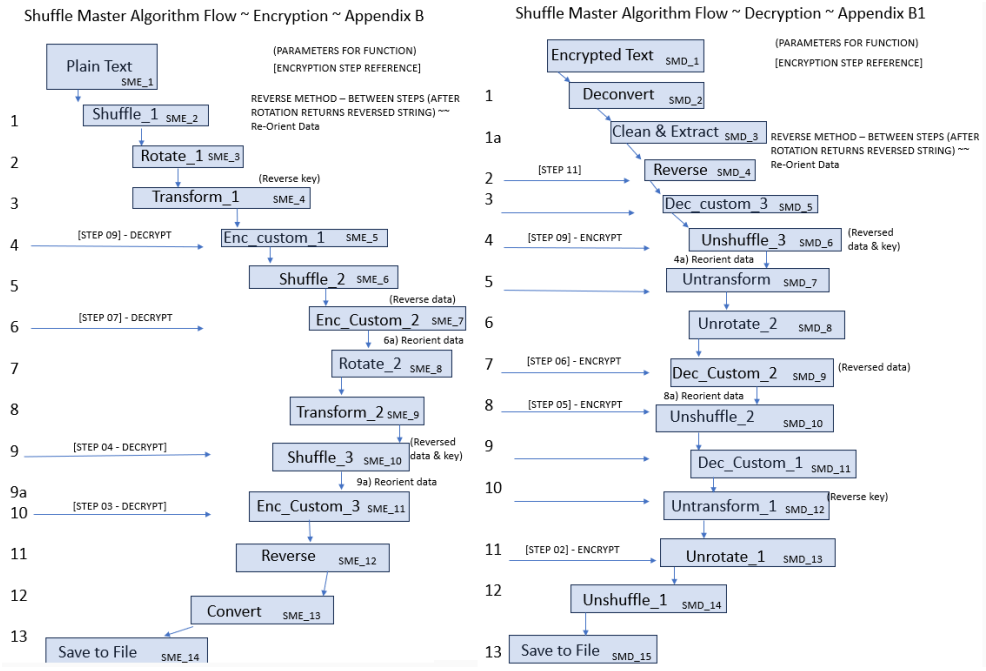

Figure 8. Shuffle Master data flow chart.

PROVISIONAL PATENT APPLICATION Docket No. P14868US00

TITLE: CUSTOM CYCLIC ENCRYPTION SYSTEM AND ASSOCIATED METHOD OF USE

TECHNICAL FIELD

[0001] The present disclosure generally relates to a system for high-level encryption that extends beyond existing constraints. This is a custom cyclic encryption system that offers strength and robustness.

BACKGROUND

[0002] The background description provided herein gives context for the present disclosure. The work of the presently named inventors, as well as aspects of the description that may not otherwise qualify as prior art at the time of filing, are neither expressly nor impliedly admitted as prior art.

[0003] Currently, experts are concerned about the existing encryption algorithms that can be cracked in record time with quantum computing. The AES-128 allegedly takes thousands, if not millions of years to crack with current technology. However, it is speculated that this time can be cut to days or months when quantum computing is more available. There is a tremendous need for encryption algorithms that are strong enough to withstand quantum computer decryption attempts.

[0004] The current encryption mechanisms have been in place for around twenty to forty years, with a few exceptions. There is a considerable need for a whole new set of standards. About ten years ago, computers really started growing in capacity and speed, so new encryption standards were developed. However, these were based on a weak initial understanding of people's willingness to go after protected information. While there are many existing encryption algorithms and standards in place, these algorithms and standards have controlled the technology that is about to undergo explosive and exponential capacity and speed increases. As quantum computing becomes available for both governments and hackers, our current antiquated encryption methodology will simply not hold up.

[0005] Thus, there exists a need in the art for an encrypted computer system that can withstand decryption from a quantum computer.

Figure 9. Perfect computing solutions, Inc. Patent Application introduction.

PROVISIONAL PATENT APPLICATION Docket No. P14868US00

ABSTRACT

[0232] This disclosure is a cyclic encryption system and the associated method of use that utilizes shuffling, transforming, rotating, or permutations to manipulate data with a key that exceeds the normal 4096 bits and prevents bypassing this methodology with any combination of the methods mentioned above. There are capabilities to handle keys with over two million characters and higher. A file is encrypted using each character in the key. Once the end of the file has been reached, the process starts over at the beginning with the next character in the key and starts at the beginning of the target file again, encrypting the data again and again until the key has been completed. The limit of the size of the key is dependent on the size of the computing device and when crashing occurs. The data is utilized in all types of computing devices in virtually all industries.

Figure 10. Perfect Computing Solutions, Inc. Patent Application abstract.

Feature	AES-256	RSA-2048	Lattice PQC	ECE (This Work)
Max Encryptable Block	32 chars	~512 chars	Varies	NONE
Quantum Resistance	✗	✗	✓	✓✓
Key/Data Entanglement	✗	✗	Partial	✓✓
Cyclical Multi-Phase Design	✗	✗	✗	✓ (12 Phases)
Lattice-Based Output	✗	✗	✓	✓✓
Nonlinear Bit Propagation	✗	✗	✗	✓
Multi-Lattice Embedding	✗	✗	Partial	✓ (Configurable)
Cryptanalysis Resistance	Medium	Medium	Strong	Extreme
Multi-Lattice Encoding	✗	✗	✓	✓✓
Self-Referential Encryption	✗	✗	✗	✓

Figure 11. Capability assessment of various styles of encryption methods.

4.4. Author and Affiliation

Jacob Koch of Perfect Computing Solutions has been involved with creating software for different types of industry including construction management, culinary preparation and recipe storage, investment clubs among others. He has been in multiple network administrator roles as well as IT security and helpdesk. In addition to these roles, he has also been in the construction layout industry. In recent years, we have been dedicated to protecting and launching this exciting new architecture in the world.

4.5. Additional Notes about ECEA

ECEA has undergone live challenge phases designed to evaluate cryptanalytic re-

sistance:

Stage 1: 488-Bit Key, 2795-Character Encrypted Output

- Duration: 30 days
- Participants: ~2000 cryptographers (including academic and enterprise teams*)
- Outcome: 0 successful decryptions

Stage 2: 64-Bit Key, 1108-Character Encrypted Output

- Duration: 30 days
- Participants: ~1000 active participants
- Progress: 0 successful decryptions

These results demonstrate ECEA's resilience across both high-entropy and reduced-key environments, driven by non-linear propagation and multi-vector encoding.

4.6. Final Thoughts

Controlled chaos—engineered with intent.

When key and data are the same length, entropy dies.

By adding recursion, complexity, and logic, the entropy is free to spiral off into oblivion.

ECEA rejects patterns, burns predictability, and redefines the whole industry.

Grover fails. Shor collapses. AI starves with no predictability.

It's not an upgrade. It's a paradigm shift.

This disruptive technology will shake up the whole industry!

Acknowledgements

We would like to thank everyone who participated in the CTF Challenge and contributed to its success.

A special note of appreciation goes to those who doubted the vision and said it would never work—your skepticism served as unexpected fuel, and for that, you hold a unique place in this journey.

We are especially grateful to the investors who believed in this project from the beginning, cheering us on and providing the funding and encouragement necessary to bring it to life. This achievement would not have been possible without your unwavering support and confidence. While many have touched on quantum resilience, no one has walked the path as we have. We are pioneers blazing a path to the future.

Provisional Patent Application

Provisional Patent Application (#63/716,466) was filed and approved on 11-05-2024.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] Public Information. Flowchart of LB-RSA Key Generation.

https://www.researchgate.net/figure/Flowchart-of-LB-RSA-Key-Generation_fig1_341504566

[2] Sravya, G., Kumar, P.S. and Padmavathy, R. (2024) Survey of Post-Quantum Lattice-Based Ciphertext-Policy Attribute-Based Encryption Schemes for Cloud Storage: Taxonomy, Open Issues, and Future Directions. *IEEE Transactions on Services Computing*, **17**, 4540-4557. https://doi.org/10.1109/tsc.2024.3479930

[3] Xagawa, K. (2018) Practical Cryptanalysis of a Public-Key Encryption Scheme Based on Non-Linear Indeterminate Equations at SAC 2017. In: Lange, T. and Steinwandt, R., Eds., *Post-Quantum Cryptography. PQCrypto* 2018, Springer International Publishing, 142-161. https://doi.org/10.1007/978-3-319-79063-3_7

[4] Xu, Z., Pemberton, O., Oswald, D. and Zheng, Z. (2023) Reveal the Invisible Secret: Chosen-Ciphertext Side-Channel Attacks on NTRU. In: Buhan, I. and Schneider, T., Eds., *Smart Card Research and Advanced Applications*, Springer, 227-247. https://doi.org/10.1007/978-3-031-25319-5_12

[5] Wendt, D.W. (2024) Combatting Generative AI Threats. In: Wendt, D.W., Ed., *The Cybersecurity Trinity*, Apress, 151-172. https://doi.org/10.1007/979-8-8688-0947-7_5

[6] Ahn, J., Hussain, R., Kang, K. and Son, J. (2025) Exploring Encryption Algorithms and Network Protocols: A Comprehensive Survey of Threats and Vulnerabilities. *IEEE Communications Surveys & Tutorials*, 1.

[7] E'mari, S.A., Sanjalawe, Y. and Allehyani, B.A. (2025) Quantum Computing Implications in Generative AI Cybersecurity. In: Almomani, A. and Alauthman, M., Eds., *Examining Cybersecurity Risks Produced by Generative AI*, IGI Global, 609-642. https://doi.org/10.4018/979-8-3373-0832-6.ch025

[8] Vadisetty, R. and Polamarasetti, A. (2024) Quantum Computing for Cryptographic Security with Artificial Intelligence. 2024 12*th International Conference on Control, Mechatronics and Automation* (*ICCMA*), London, 11-13 November 2024, 252-260. https://doi.org/10.1109/iccma63715.2024.10843897

[9] Panteli, A. (2025) White Paper: Quantum Readiness-Strategic Imperatives for Enterprise Organisations. *European Journal of Business and Management Research*, **10**, 144-151. https://doi.org/10.24018/ejbmr.2025.10.3.2705