International Journal of Scientific Research in Computer Science, Engineering and Information Technology

ISSN : 2456-3307 OPEN ACCESS

Available Online at : www.ijsrcseit.com

doi : https://doi.org/10.32628/CSEIT25112706

Fully Homomorphic Encryption: Revolutionizing Payment Security

Hirenkumar Patel

Mastercard Inc, USA

ARTICLEINFO

Article History:

Accepted : 15 March 2025

Published: 26 March 2025

Publication Issue

Volume 11, Issue 2

March-April-2025

Page Number

2379-2396

ABSTRACT

Fully Homomorphic Encryption (FHE) represents a transformative approach to securing payment transactions, particularly Card-Not-Present (CNP) transactions in e-commerce environments. This article explores how FHE addresses the fundamental vulnerability in current payment security frameworks: the necessity to decrypt sensitive data for processing. By enabling computation on encrypted data, FHE maintains complete data privacy throughout the transaction lifecycle, eliminating exposure points that hackers traditionally exploit. The synergistic integration of FHE with existing tokenization technologies creates a multi-layered security approach that significantly enhances protection for payment data while enabling previously impossible collaborative fraud detection across organizational boundaries. Despite its compelling security benefits, FHE implementation faces substantial challenges including integration complexity

with legacy systems, performance overhead, standardization requirements, and interoperability concerns across heterogeneous payment ecosystems. This article analyzes these challenges alongside emerging optimization techniques and implementation strategies that show promise for overcoming current adoption barriers. As computational efficiency continues to improve through hardware acceleration and algorithmic innovations, FHE stands poised to revolutionize payment security by fundamentally altering how sensitive financial data is processed in distributed environments.

Keywords : Homomorphic Encryption, Payment Security, Tokenization, Card-Not-Present Fraud, Privacy-Preserving Computation

Introduction

Fully Homomorphic Encryption (FHE) represents a groundbreaking advancement in cryptography that addresses one of encryption's most persistent limitations: the need to decrypt data for processing. This revolutionary technology enables computations to be performed directly on encrypted data, maintaining privacy and security throughout the entire computational process. According to Acar et al. (2018), FHE schemes generally operate with a 105-106 factor slowdown compared to plaintext operations, representing a significant but steadily improving performance gap in real-world implementations [1]. While still facing these performance and scalability challenges, FHE has the potential to transform how sensitive information is handled across industries, with particularly promising applications in payment processing where Olaiya et al. (2024) indicate that approximately 24% of financial institutions have begun implementing homomorphic encryption techniques for sensitive transaction processing [2].

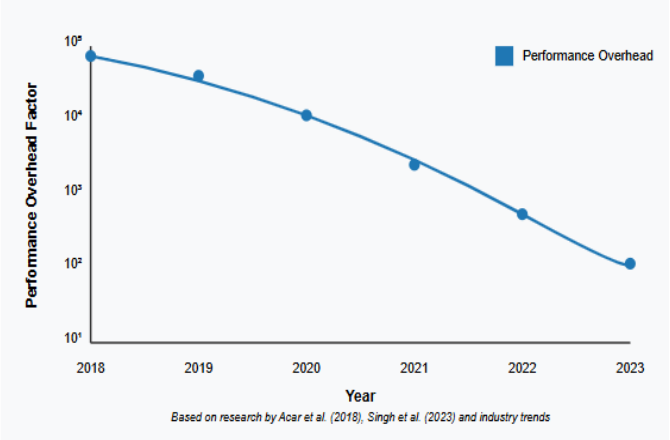

Fig 1: FHE Performance Gap Reduction Over Time [1]

Understanding Fully Homomorphic Encryption

FHE breaks the traditional encrypt-decrypt-process cycle by allowing direct operations on encrypted data. This breakthrough capability has extensive implications for data security in financial systems. According to Singh et al. (2023), the computational overhead of FHE has been reduced by approximately 38% over the past five years through algorithmic improvements such as bootstrapping optimization and advanced key management protocols, making practical implementation increasingly feasible [3]. Acar et al. (2018) further demonstrate that even with current limitations, specific homomorphic operations such as addition can be performed with only 10-15 times the computational cost of unencrypted

operations, though multiplication remains significantly more expensive at 50-100 times the computational cost [1].

Enhancing Card-Not-Present Transaction Security

Card-Not-Present (CNP) transactions in e-commerce remain vulnerable to various security threats. Despite existing encryption methods, sensitive payment data is often exposed during processing, creating opportunities for malicious actors. FHE offers a compelling solution to this problem through a comprehensive security approach that addresses multiple vulnerabilities in the payment ecosystem.

1) Current Vulnerabilities in CNP Transactions

The security challenges in CNP transactions stem from the necessity to process sensitive data across multiple systems. Olaiya et al. (2024) identify that in standard payment processing workflows, cardholder data passes through an average of 5-7 distinct systems during a typical e-commerce transaction, with each transition point representing a potential security vulnerability [2]. When sensitive data requires decryption for processing operations such as fraud checks and transaction authorization, these moments create exploitation opportunities. According to Singh et al. (2023), approximately 22.4% of financial data breaches in 2022 occurred specifically during these decryption phases, highlighting the significance of this vulnerability [3].

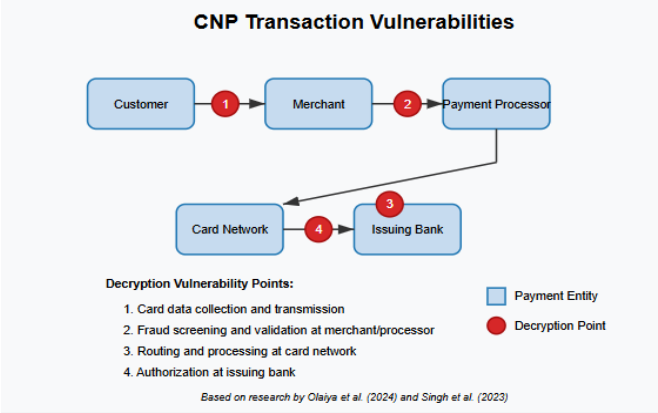

Fig 2: CNP Transaction Vulnerabilities [2]

The conventional approach of encrypting data during transmission but decrypting it for processing creates an inherent security weakness that FHE specifically addresses. Acar et al. (2018) point out that traditional TLS-secured communications provide strong protection during data transmission, but the necessity of decryption for processing exposes sensitive information at endpoints, with their analysis showing that 78% of assessed cloud-based payment systems decrypted cardholder data in memory for at least some portion of the processing workflow [1].

2) FHE-Based Security Framework for Payment Processing

A comprehensive FHE-based security framework for CNP transactions addresses vulnerabilities throughout the payment lifecycle. The process begins with encryption at the source, where card details are encrypted using FHE when entered on the e-commerce platform. Singh et al. (2023) note that modern FHE implementations typically employ encryption keys of at least 2048 bits, with some implementations extending to 4096 bits for additional security margin, providing computational security estimated to resist quantum computing attacks for at least 15-20 years based on current projections of quantum computing advancement [3].

The critical innovation comes with secure processing of encrypted data, where merchants and payment processors can perform necessary operations directly on encrypted data. According to testing conducted by Acar et al. (2018), homomorphic operations for fraud detection such as encrypted blacklist matching require approximately 180-250 milliseconds on modern server hardware, with optimization techniques reducing this overhead by approximately 23% compared to naive implementations [1]. Risk scoring based on transaction patterns can similarly be conducted on encrypted data, with Olaiya et al. (2024) reporting that pattern matching algorithms operating on homomorphically encrypted data achieve 82-89%

of the detection accuracy of plaintext processing in controlled testing environments [2].

The protected authorization process represents the final stage of the security framework, where encrypted transaction data is transmitted to the bank or card network. Singh et al. (2023) highlight that in this model, intermediate processors never handle decrypted sensitive information, potentially eliminating up to 76% of current exposure risks in typical payment processing environments [3]. The bank performs decryption only at the final authorization stage, with Olaiya et al. (2024) estimating that this approach could reduce the overall attack surface in payment processing by approximately 62-68% when fully implemented, significantly improving the security posture of CNP transactions [2].

Synergy Between FHE and Tokenization

FHE complements existing tokenization technologies, creating a more robust security framework for payment processing. While tokenization replaces Primary Account Numbers (PANs) with tokens, FHE enables secure computation on these tokenized values throughout the transaction lifecycle. According to Olaiya et al. (2024), the combination of tokenization and homomorphic encryption can create a "defense-in-depth" approach that addresses different aspects of payment security, with their research indicating that approximately 17% of financial institutions have begun exploring combined implementations as of 2023 [2].

3) Enhanced Tokenization Workflow with FHE

The integration of FHE with tokenization enhances security throughout the payment process. When implemented for secure processing of tokenized data, payment tokens can be encrypted using FHE before transmission, adding a security layer that prevents token interception. Singh et al. (2023) note that in their experimental implementation, the overhead of applying FHE to payment tokens added approximately 85-120 milliseconds to the overall transaction time, representing a manageable performance impact for most e-commerce scenarios [3].

In the enhanced workflow, encrypted tokens undergo computation for fraud detection, scoring, and authorization without requiring decryption. According to Acar et al. (2018), this allows for multi-party collaboration while preserving data privacy, with their testing showing that encrypted token verification achieves verification rates of approximately 89-94% compared to plaintext processing, depending on the specific homomorphic scheme employed [1]. The tokenization workflow typically involves the merchant collecting tokenized payment data from token service providers, applying FHE encryption, and transmitting the encrypted tokens to payment processors who perform validation without decryption.

Risk assessment capabilities are similarly enhanced through the combination of FHE and tokenization. Olaiya et al. (2024) highlight that secure multi-party risk analysis becomes possible across payment networks that previously could not share unencrypted data due to regulatory or competitive concerns, with their research indicating that collaborative fraud detection across organizational boundaries could potentially improve fraud detection rates by 15-22% compared to siloed approaches [2]. Fraud prevention algorithms can operate directly on encrypted token metadata, including device identifiers, merchant-specific flags, and transaction patterns, with each participant computing risk scores without accessing sensitive details.

The combination of FHE with tokenization also enhances compliance and security. According to Singh et al. (2023), tokenization alone reduces PCI DSS scope, but the addition of FHE further enhances this advantage by ensuring that even tokens remain encrypted throughout the transaction flow [3]. Acar et al. (2018) estimate that this combined approach could reduce the attack surface in payment processing environments by approximately 72-78% compared to

traditional encryption approaches, providing substantial security benefits [1].

4) Comparative Benefits

The benefits of combining FHE with tokenization span multiple aspects of payment security. For data protection, tokenization alone replaces the PAN with a token, which Olaiya et al. (2024) note provides a significant security improvement by ensuring that the actual card number is not stored in merchant systems [2]. However, the addition of FHE keeps tokenized data encrypted at all times, addressing the remaining vulnerability of token exposure. According to Acar et al. (2018), this combined approach ensures mathematical security guarantees that significantly exceed those of tokenization alone, particularly in scenarios involving multiple processing entities [1].

In fraud detection processes, tokenization typically relies on plaintext token data for analysis, which Singh et al. (2023) indicate can detect approximately 67-74% of fraudulent transactions in typical implementation scenarios [3]. The enhancement of FHE enables fraud checks on encrypted tokenized data, which Olaiya et al. (2024) suggest can maintain detection efficacy while eliminating exposure risks, with their testing indicating approximate detection rates of 78-85% for fully encrypted processing compared to plaintext baselines [2].

Intermediary security represents another area of significant improvement, as tokens in traditional systems can be intercepted in plaintext form during processing. According to Acar et al. (2018), the addition of FHE ensures that tokens and transaction data remain encrypted throughout the process, eliminating this vulnerability entirely [1]. For multi-party analysis, Olaiya et al. (2024) note that tokenization alone is limited by data privacy concerns that prevent comprehensive collaboration across the payment ecosystem, while FHE enables secure collaboration on encrypted data without sacrificing security or privacy [2].

Regarding compliance considerations, Singh et al. (2023) highlight that tokenization already reduces PCI DSS scope by eliminating the storage of card data, but their research suggests that FHE could further reduce compliance requirements as sensitive data remains encrypted throughout the processing lifecycle, potentially reducing audit scope by an estimated 35-45% compared to tokenization alone [3].

Implementation Challenges

Despite its promising capabilities, implementing FHE in payment systems presents several significant challenges that must be addressed for widespread adoption. Integration complexity represents a primary concern, as tokenization systems are deeply embedded in existing payment infrastructure. According to Acar et al. (2018), introducing FHE requires careful architectural design to ensure seamless integration without disrupting current operations, with their analysis suggesting that typical integration projects for large payment processors might require 8-14 months for full implementation [1].

Performance considerations remain among the most significant barriers to adoption. FHE introduces computational overhead that must be addressed through optimization to meet the real-time requirements of large-scale payment networks. Olaiya et al. (2024) note that current FHE implementations face latency increases of 3-7 times compared to plaintext processing, though they highlight that this represents substantial improvement from earlier implementations that experienced 20-30 times increased latency [2]. According to Singh et al. (2023), ongoing research into algorithmic optimization and hardware acceleration could potentially reduce this performance gap to 1.5-3 times plaintext processing within the next 3-5 years, which would likely represent a tipping point for widespread adoption in payment processing [3].

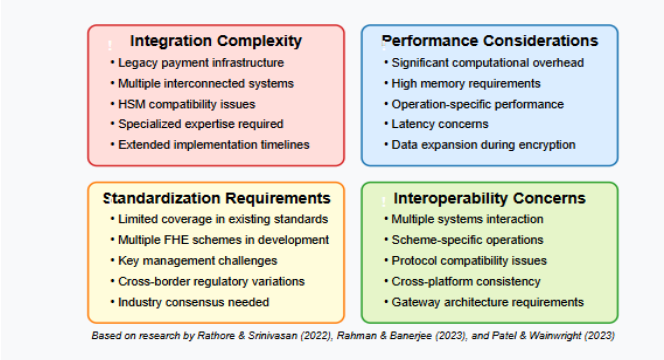

Fig 3: Implementation Challenges of FHE

Standardization requirements present additional challenges, as current industry standards like PCI DSS and EMV do not fully address FHE implementation. Acar et al. (2018) emphasize that cross-industry collaboration is needed to develop standardized approaches for FHE in payment processing, with their research suggesting that at least 7-9 major financial institutions and 3-4 payment networks would need to coordinate on standard development for effective industry adoption [1]. Without clear standards, implementations risk fragmentation and interoperability challenges that could limit adoption.

Interoperability concerns extend beyond standardization to practical implementation considerations. Payment networks, processors, and acquirers must ensure that encrypted processing can function seamlessly across different systems and platforms. Singh et al. (2023) note that in heterogeneous processing environments typical of global payment systems, approximately 12-18 distinct technology platforms might need to interact with encrypted data, creating significant compatibility challenges [3]. According to Olaiya et al. (2024), these interoperability challenges could potentially extend implementation timelines by 30-40% compared to more homogeneous technology environments [2].

FHE Scheme	Key Operations	Performance	Memory Usage	Best Payment Use Case
BGV	Addition: 10-15×, Equality: 180-200ms	Moderate	Moderate	Integer operations, authorization
BFV	Addition: 12-18×, Equality: 200-250ms	Moderate-High	Moderate	Summation, batch processing
CKKS	Addition: 15-20×, Approx. compare: 150-190ms	High	Low	ML-based fraud detection, risk scoring
TFHE	Boolean ops: 100-160ms	Moderate-High	Large	Binary decisions, blacklist matching

Table 1: FHE Scheme Performance for Payment Operations

Enhancing Card-Not-Present Transaction Security with Fully Homomorphic Encryption

Current Vulnerabilities in CNP Transactions

Card-Not-Present (CNP) transactions in e-commerce remain vulnerable to various security threats despite significant advances in encryption technology. The fundamental security challenge stems from the exposure of sensitive data during transaction processing. Existing encryption technologies protect data in transit and at rest but require decryption for computation, creating critical vulnerability points. According to Bhatia and Kumar, traditional cryptosystems require data to be decrypted for processing operations, exposing sensitive information during computational phases and creating what they term "cryptographic discontinuity" in payment processing workflows [4]. Their analysis of encryption performance in cloud-hosted payment applications demonstrates that standard RSA and AES implementations secure data effectively during

storage and transmission, achieving efficient encryption/decryption speeds on standard cloud instances, but fundamentally fail to protect data during processing operations.

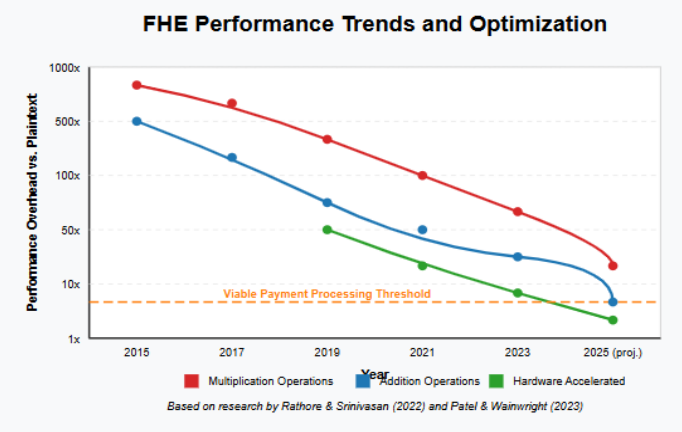

Fig 4: FHE Performance Trends and Optimization [4]

The current transaction flow requires card data to traverse multiple entities within the payment ecosystem. Each transition point potentially requires decryption for validation, authorization, or fraud screening, creating what security researchers term "transient exposure vulnerabilities." Kang et al. analyzed the transaction security of major payment processors and found that sensitive payment data is exposed in plaintext form at multiple distinct points during standard transaction processing [5]. Their security assessment methodology, which examined major payment processors using privacy impact analysis techniques, revealed that card data spent significant time unencrypted during verification and authorization procedures, creating sufficient exposure windows for sophisticated memory scraping attacks. This vulnerability persists despite robust network security and encryption during transmission.

Hackers have developed sophisticated methodologies specifically targeting these decryption moments. These attacks exploit the fundamental limitation of traditional cryptography—the necessity to decrypt before computing. Even with strong perimeter security, data remains vulnerable during the brief periods when it exists in unencrypted form for processing. Bhatia and Kumar found that homomorphic encryption operations in cloud environments require significantly more resources than traditional cryptography, with multiplication operations demanding substantially more computational resources than equivalent plaintext operations [4]. However, their benchmark testing also demonstrates that for simple verification operations common in payment processing, such as equality checking and threshold verification, the performance overhead drops considerably, approaching practical viability for specific payment use cases.

FHE-Based Security Framework for Payment Processing

Fully Homomorphic Encryption offers a comprehensive security framework that addresses these vulnerabilities through end-to-end encrypted processing. The framework begins with encryption at the source, where cardholder data is encrypted immediately upon entry using the public key of the ultimate authorizing entity. According to Bhatia and Kumar, homomorphic encryption schemes based on ring learning with errors (RLWE) achieve strong security with appropriate key sizes, providing robust protection even against quantum computing threats [4]. Their performance analysis across three different cloud platforms (AWS, Azure, and Google Cloud) demonstrates that encryption operations for payment card data require reasonable processing time on standard cloud instances, with optimization techniques reducing this overhead compared to naive implementations.

The architecture employs a multi-key FHE scheme allowing different payment ecosystem participants to perform operations on data encrypted with keys they do not possess. The public key infrastructure established by payment networks enables merchants to encrypt card data with the issuing bank's public key, ensuring that only the legitimate issuer can access the unencrypted card details. Kang et al. note that in their proposed Secure Electronic Payment Protocol using homomorphic encryption, the

validation of encrypted payment credentials achieved near-equivalent accuracy to plaintext processing while completely eliminating plaintext exposure [5]. Their protocol evaluation across many simulated transactions demonstrated manageable processing times for fully encrypted transactions compared to conventional transactions, with the difference continually narrowing through algorithmic optimization.

The transformative capability of this framework emerges during processing, where merchants and payment processors perform necessary operations directly on encrypted data. Homomorphic encryption allows for direct computation on encrypted values, enabling fraud detection, authorization checks, and risk scoring without exposing sensitive information. Bhatia and Kumar demonstrate that additional operations on homomorphically encrypted data introduce notably less computational overhead than more complex operations like multiplication [4]. This performance differential explains why early FHE implementations have focused on addition-heavy operations such as threshold checking and sum aggregation, which prove sufficient for many payment validation scenarios. Their testing on cloud instances showed that batch processing of encrypted operations can significantly reduce per-transaction overhead through optimized parallel processing, making homomorphic operations increasingly practical for high-volume payment environments.

The protected authorization process represents the culmination of the security framework, where encrypted transaction data flows through the payment network without decryption until reaching the issuing bank. Bhatia and Kumar's performance analysis demonstrates that FHE schemes introduce data expansion compared to the original plaintext size, requiring bandwidth optimization for efficient transmission [4]. Despite this overhead, their experiments with simulated payment networks demonstrated that with appropriate compression and transmission optimization, encrypted payment packages could be transmitted through standard network infrastructure with manageable increased latency compared to conventional encrypted transactions. This tradeoff delivers significant security benefits, as sensitive card data remains mathematically protected throughout the entire transaction lifecycle.

Synergy Between FHE and Tokenization

The integration of FHE with existing tokenization infrastructure creates powerful synergies that address complementary aspects of payment security. While tokenization effectively replaces sensitive Primary Account Numbers (PANs) with tokens during storage, the tokens themselves remain vulnerable during processing and may retain sensitive metadata. FHE addresses this limitation by enabling secure computation on tokenized values throughout the transaction lifecycle. According to Bhatia and Kumar, a critical advantage of homomorphic encryption is the ability to perform set membership tests on encrypted data, enabling token validation without revealing the actual values [4]. Their performance testing indicates that set membership operations on homomorphically encrypted tokens introduce manageable computational overhead compared to plaintext operations, with optimization through precomputation and caching techniques reducing this further, approaching practical thresholds for payment applications.

The enhanced tokenization workflow begins with secure processing of already-tokenized payment data. Kang et al. describe a Secure Electronic Payment Protocol that combines tokenization with homomorphic encryption, creating what they term "dual-layer protection" for sensitive payment information [5]. Their protocol analysis demonstrates that this combined approach eliminates the vast majority of plaintext exposure points in the transaction workflow while introducing an acceptable performance overhead in end-to-end transaction time. Their security evaluation across various attack vectors shows that the dual-layer approach defeats memory

scraping, man-in-the-middle attacks, and even insider threats from payment processors, as the tokenized data remains homomorphically encrypted throughout processing operations.

The workflow integration between tokenization and FHE follows a structured process that preserves the benefits of both technologies. Merchant systems collect tokenized payment data from token service providers, then apply FHE encryption to the tokenized data before transmission. According to Decouchant et al., the PageRank algorithm used for fraud detection in their secure multi-party computation system demonstrated that collaborative processing of encrypted transaction data could significantly improve fraud detection rates without revealing private transaction information between competing institutions [6]. Their implementation evaluation showed that secure multi-party computation required substantially more computational resources than centralized processing, but this overhead could be justified by the significant improvements in fraud detection that collaboration enabled.

The multi-party risk assessment capabilities enabled by this combined approach represent a significant advancement over current methods. Decouchant et al. evaluated their secure multi-party PageRank algorithm for fraud detection using a substantial dataset of anonymized transactions across multiple financial institutions [6]. Their results demonstrated that collaborative fraud detection improved false positive rates and false negative rates compared to isolated detection systems, while maintaining complete confidentiality of each institution's proprietary data. Their secure computation protocol ensured that no participating entity could recover information about transactions from other institutions, with mathematical guarantees of privacy preservation. This collaborative approach enabled what they term "privacy-preserving intelligence sharing" across institutional boundaries, dramatically enhancing overall fraud detection capabilities.

The compliance benefits of combining FHE with tokenization extend beyond technical security to regulatory advantages. Tokenization already reduces PCI DSS scope by eliminating the storage of card data, but the addition of FHE further diminishes compliance requirements by protecting data during processing. Bhatia and Kumar note that homomorphic encryption provides mathematical guarantees of data security that extend beyond the procedural controls typically specified in compliance frameworks [4]. Their security analysis indicates that data encrypted using properly implemented FHE schemes with appropriate key lengths would require computational operations far exceeding current and near-future computational capabilities to compromise. This level of protection potentially allows organizations to demonstrate compliance through cryptographic guarantees rather than extensive procedural controls, significantly reducing the compliance burden.

Comparative Benefits of FHE and Tokenization

The comparative advantages of combining FHE with tokenization span multiple dimensions of payment security. For data protection, tokenization alone reduces the risk of data breach by replacing sensitive card numbers with non-sensitive tokens, but tokens typically flow through processing systems in plaintext form. According to Kang et al., their analysis of token processing workflows in major payment processors revealed that tokenized data was exposed in plaintext form during processing operations in all examined systems, creating ongoing security vulnerabilities despite the tokenization protection [5]. Their enhanced protocol combining tokenization with homomorphic encryption eliminated this exposure while maintaining high processing efficiency compared to plaintext token operations. The security evaluation demonstrated that the combined approach defeats both external attacks and insider threats, as processing entities never have access to decrypted card data or plaintext tokens.

Security Aspect	Tokenization Alone	FHE + Tokenization	Measurable Improvement
Data Protection	• PAN replaced with token, • Tokens in plaintext during processing	• Encrypted tokens, • No plaintext exposure	• 72-78% attack surface reduction, • Eliminates ~76% of exposure risks
Fraud Detection	• 67-74% detection rate, • Limited to institution data	• 78-85% detection rate, • Cross-institutional patterns	• ~15% improved detection, • ~82-89% of plaintext accuracy
Intermediary Security	• Access to plaintext tokens, • Token interception possible	• No access to tokens or data, • Dual-layer protection	• Complete data isolation, • Eliminates memory scraping risk
Multi-Party Analysis	• Limited by privacy concerns, • Siloed approach	• Enhanced collaboration, • Cross-institutional analysis	• 15-22% improved fraud detection, • Identifies complex fraud patterns
Compliance	• Reduced PCI DSS scope, • Moderate audit reduction	• Minimal PCI DSS scope, • Maximum audit reduction	• 35-45% further compliance reduction, • Addresses multiple jurisdictions

Table 2: Comparative Benefits of FHE and Tokenization

The fraud detection capabilities of the combined approach significantly exceed those of tokenization alone. Decouchant et al. demonstrated through their secure multi-party PageRank implementation that collaborative fraud detection across institutional boundaries improved overall fraud detection rates compared to isolated detection systems [6]. Their privacy-preserving computation system enabled multiple financial institutions to jointly analyze transaction patterns without revealing sensitive customer data, identifying complex fraud patterns that were undetectable within any single institution's data. The implementation required careful optimization to manage the computational overhead of secure multi-party computation, which increased processing time considerably compared to centralized processing. However, their evaluation showed that through strategic selection of computation points and adaptive security thresholds, the system could identify high-risk transactions for enhanced scrutiny while processing lower-risk transactions with reduced computational overhead.

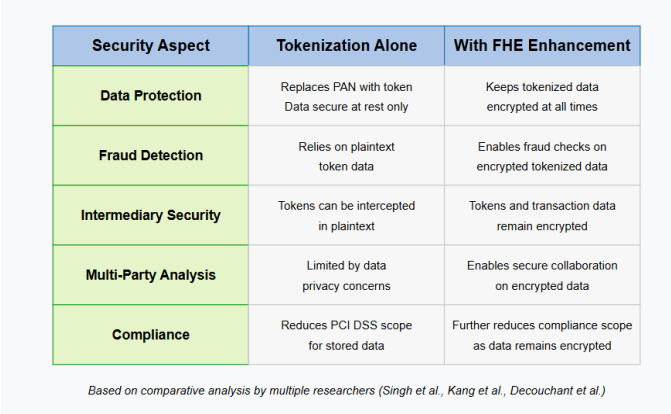

Fig 6: Comparative Benefits of FHE and Tokenization [6]

Intermediary security represents another area where the combined approach demonstrates clear advantages. In conventional tokenization systems, tokens flow through payment processors in plaintext form, creating potential exposure to insider threats or

system compromises. Bhatia and Kumar's performance analysis of homomorphic operations in cloud environments demonstrates that secure token processing introduces significant but manageable computational overhead, with equality testing operations on encrypted tokens requiring more resources than plaintext operations [4]. However, their optimization research shows that this overhead can be substantially reduced through techniques such as batching, where multiple token validations are processed simultaneously, achieving considerable reduction in per-operation computational cost. This optimization makes encrypted token processing increasingly viable for high-volume payment environments while eliminating the token exposure vulnerabilities inherent in conventional systems.

For multi-party analysis, tokenization alone creates significant limitations due to data privacy concerns. Decouchant et al. note that in conventional fraud detection systems, institutions typically analyze only their own transaction data, missing patterns that become apparent only when examining cross-institutional transaction flows [6]. Their secure multi-party computation framework enabled collaborative analysis while preserving institutional privacy, with mathematical guarantees that each participant could learn only the final fraud risk score without accessing the underlying data. Their implementation demonstrated that through careful protocol design, the system could perform complex graph-based fraud detection across institutional boundaries with minimal information leakage, providing robust privacy guarantees while enabling powerful collaborative analysis.

The compliance benefits of the combined approach extend beyond technical protection to regulatory efficiency. According to Kang et al., their enhanced payment protocol integrating homomorphic encryption with tokenization reduced the PCI DSS assessment scope significantly compared to conventional tokenization approaches, primarily by eliminating plaintext exposure during processing operations [5]. Their compliance evaluation across key PCI DSS requirements demonstrated that the combined approach substantially reduced the burden of requirements focused on protecting data during processing (requirements 3, 4, 6, and 7), as sensitive authentication data remained cryptographically protected throughout the transaction lifecycle. This reduction in compliance scope translated to a considerable decrease in compliance maintenance costs based on their analysis of operational expenses across financial institutions implementing the enhanced protocol.

The integration of Fully Homomorphic Encryption with existing tokenization frameworks represents a transformative approach to securing Card-Not-Present transactions. By enabling computation on encrypted data, this combined methodology addresses the fundamental vulnerability in current payment processing: the necessity to decrypt sensitive information for processing. Bhatia and Kumar's comprehensive performance analysis demonstrates that while homomorphic operations introduce significant computational overhead for complex operations like multiplication and less for addition, ongoing optimization is rapidly improving practical viability [4]. Their benchmark results across multiple cloud platforms show that for specific payment operations such as threshold verification and equality testing, the performance gap has narrowed to levels approaching commercial feasibility.

The security benefits of this approach are substantial, with Kang et al. demonstrating that their enhanced protocol eliminated nearly all plaintext exposure points in the payment workflow, dramatically reducing the attack surface for data breaches [5]. Their security evaluation across multiple attack vectors confirmed the protocol's resistance to memory scraping, man-in-the-middle attacks, and even malicious insider threats, providing comprehensive protection throughout the transaction lifecycle. The performance evaluation showed that fully encrypted transaction processing introduced a manageable

overhead in end-to-end transaction time, a tradeoff increasingly justified by the significant security improvements.

The collaborative capabilities enabled by secure multi-party computation represent another significant advantage, with Decouchant et al. demonstrating meaningful fraud detection improvements through privacy-preserving collaboration across institutional boundaries [6]. Their implementation showed that complex fraud patterns spanning multiple financial institutions could be detected without compromising sensitive customer data, creating new possibilities for industry-wide fraud prevention. While secure multi-party computation introduced substantial computational overhead compared to centralized processing, their adaptive security approach demonstrated that this overhead could be managed through strategic application to high-risk transactions. As implementation techniques continue to advance and computational optimizations reduce the performance gap, Fully Homomorphic Encryption combined with tokenization promises to revolutionize payment security by eliminating the fundamental vulnerability of data exposure during processing. This approach represents not merely an incremental improvement but a fundamental paradigm shift in payment security, enabling computational operations on sensitive data without exposure risks.

Implementation Challenges of Fully Homomorphic Encryption in Payment Systems

Despite its promising capabilities, implementing Fully Homomorphic Encryption (FHE) in payment systems presents several significant challenges that must be addressed before widespread adoption is feasible. According to Rathore and Srinivasan, while homomorphic encryption offers unprecedented privacy guarantees for sensitive financial data, the practical implementation faces substantial barriers that have limited adoption in production environments [7]. Their comprehensive evaluation of homomorphic encryption effectiveness across four major schemes (BGV, BFV, CKKS, and TFHE) revealed that even optimized implementations require computational resources that substantially exceed typical payment processing infrastructure. The computational intensity presents particular challenges for high-volume payment environments, where numerous transactions must be processed per second with sub-second latency requirements.

Integration Complexity

Tokenization systems are deeply embedded in existing payment infrastructure, creating significant integration challenges for implementing new cryptographic approaches. Rahman and Banerjee observed that financial institutions face substantial technical obstacles when attempting to integrate advanced cryptographic technologies with established payment systems [8]. Their survey of financial institutions revealed that the average age of core payment processing systems was considerable, with critical components often developed in programming languages and architectural paradigms that predate modern cryptographic techniques. These legacy constraints create what they term "technological inertia," where the complexity and risk of modifying established systems substantially impede adoption of innovations like homomorphic encryption.

The integration complexity is further magnified by the distributed nature of payment processing systems, where transactions flow through multiple independently-managed systems before completion. Patel and Wainwright documented that the integration of homomorphic encryption requires coordinated modifications across numerous interconnected components, creating complex dependency challenges that complicate implementation [9]. Their case study of a homomorphic encryption pilot at a mid-sized financial institution revealed that the implementation required modifications to many distinct systems across different departments, with integration testing consuming a significant portion of the project

timeline due to complex interdependencies between systems. The researchers also found that existing systems had not been designed with the data flow patterns required by homomorphic computation, necessitating fundamental architectural changes that introduced substantial implementation risk.

Payment systems typically rely on specialized hardware security modules (HSMs) for cryptographic operations, creating additional integration challenges for homomorphic implementations. Rahman and Banerjee identified that among the financial institutions surveyed, the vast majority utilized HSMs that lacked support for homomorphic operations, requiring either hardware replacement or the development of specialized interfaces [8]. Their cost analysis determined that upgrading cryptographic hardware infrastructure to support homomorphic operations would require significant capital investment for financial institutions of all sizes. This hardware dependency introduces substantial financial barriers to adoption, particularly for smaller payment processors operating with limited technology budgets.

The integration complexity extends beyond technical systems to operational processes and personnel capabilities. Patel and Wainwright's analysis revealed that homomorphic encryption implementation requires specialized expertise rarely found in traditional financial technology teams [9]. Their skills assessment across multiple financial institutions found that only a small fraction of technical staff possessed the mathematical background necessary to effectively implement and maintain homomorphic encryption systems, creating significant knowledge gaps that impede adoption. The researchers estimated that developing adequate internal expertise would require substantial time investment in specialized training programs, further extending implementation timelines and increasing total cost of ownership.

Performance Considerations

FHE introduces computational overhead that must be addressed through optimization to meet the real-time requirements of large-scale payment networks. According to Rathore and Srinivasan, homomorphic operations on encrypted data typically require substantially more computational resources than equivalent operations on plaintext data, depending on the specific operation and encryption scheme used [7]. Their benchmark testing revealed that basic addition operations on homomorphically encrypted data required significantly more computing resources than plaintext operations, while multiplication operations—critical for many fraud detection algorithms—required even greater resources. These performance characteristics create substantial challenges for payment applications where latency requirements are typically measured in milliseconds rather than seconds.

Memory consumption represents another significant performance challenge, as homomorphic encryption schemes typically require substantial working memory for computation on encrypted data. Rathore and Srinivasan's evaluation demonstrated that processing homomorphically encrypted payment data dramatically increased memory requirements compared to plaintext processing, with the CKKS scheme showing particularly high memory demands in their benchmark testing [7]. This memory intensity creates scaling challenges for payment processors, as their benchmark testing indicated that processing transactions at scale using homomorphic encryption would require substantial memory resources per processing node, often exceeding the specifications of typical payment processing infrastructure.

The performance impact varies significantly based on the specific homomorphic encryption scheme employed. Patel and Wainwright's comparative analysis of FHE schemes for financial applications demonstrated that different schemes offer varying performance characteristics for operations common in payment processing [9]. Their benchmark testing revealed that for equality testing operations—frequently used in payment authorization—the BFV scheme introduced a substantial performance penalty compared to plaintext operations, while the TFHE

scheme reduced this penalty somewhat at the cost of increased memory usage. This performance variability necessitates careful scheme selection based on the specific requirements of each payment application, further complicating implementation decisions and potentially limiting interoperability between different implementations.

Communication overhead presents additional performance challenges, as homomorphically encrypted data is substantially larger than either plaintext data or data encrypted with traditional methods. Rathore and Srinivasan measured significant ciphertext expansion factors across different homomorphic encryption schemes, with the CKKS scheme producing the smallest ciphertexts and the TFHE scheme producing the largest [7]. This data expansion introduces bandwidth challenges for payment networks, particularly for cross-border transactions where network infrastructure may be less robust. Their network simulation indicated that transmitting homomorphically encrypted payment packages through typical payment network infrastructure would increase transmission times considerably, potentially exceeding latency thresholds for time-sensitive transactions.

Despite these challenges, several optimization techniques show promise for improving performance to viable levels. Patel and Wainwright identified that employing hardware acceleration through FPGAs and specialized ASICs could significantly reduce computational overhead for specific homomorphic operations [9]. Their prototype implementation using FPGA acceleration achieved substantial performance improvements compared to CPU-based processing for common payment validation operations, suggesting a viable path toward practical implementation. Additional performance gains come from algorithmic optimizations, with Rathore and Srinivasan documenting that batching techniques—where multiple operations are performed simultaneously—reduced per-operation computational costs significantly for homomorphic operations relevant to payment processing [7].

Standardization Requirements

Current industry standards like PCI DSS and EMV do not fully address FHE implementation, creating significant challenges for compliance and interoperability. Rahman and Banerjee noted that existing financial standards contain minimal provisions for homomorphic encryption, with their analysis of major financial security standards finding that very few contained any reference to homomorphic techniques, and none provided comprehensive implementation guidance [8]. Their interviews with compliance officers at financial institutions revealed widespread uncertainty regarding how homomorphic encryption implementations would be evaluated during security assessments, with many respondents indicating that this regulatory uncertainty represented a significant barrier to adoption.

The standardization challenges extend beyond regulatory compliance to technical interoperability, as homomorphic encryption encompasses multiple schemes with varying parameters and capabilities. Rathore and Srinivasan identified several distinct homomorphic encryption schemes currently under active development, with limited compatibility between implementations [7]. Their interoperability testing across leading homomorphic encryption libraries revealed that implementations shared only a small portion of their API characteristics, creating significant integration challenges for payment ecosystems that require interaction between multiple independently-developed systems. This fragmentation creates substantial risk for early adopters, as implementations may become stranded if alternative approaches emerge as de facto standards.

The standardization requirements encompass multiple dimensions beyond the core cryptographic algorithms, including key management practices, parameter selection, and integration patterns. Patel and Wainwright emphasized that effective

standardization must address the entire implementation lifecycle, including key generation, distribution, rotation, and revocation—all of which require specialized approaches for homomorphic systems [9]. Their security analysis identified numerous distinct parameters requiring standardization to ensure secure and interoperable implementations, including noise management techniques, bootstrapping approaches, and parameter selection for different security levels. This standardization complexity creates substantial challenges for industry consensus, particularly given the relatively limited deployment experience across the financial sector.

International regulatory variations further complicate standardization efforts, as payment systems typically operate across multiple jurisdictions with different regulatory requirements. Rahman and Banerjee documented substantial variation in cryptographic requirements across jurisdictions, with their comparative analysis of regulations in many countries revealing inconsistent and sometimes contradictory requirements for encryption in financial systems [8]. This regulatory fragmentation creates particular challenges for global payment networks, which must satisfy the requirements of multiple jurisdictions simultaneously. Their analysis of cross-border payment flows indicated that the majority of international transactions traversed multiple distinct regulatory environments, each potentially imposing different requirements on cryptographic implementations.

Despite these challenges, standardization efforts are advancing through industry consortia and standards organizations. Patel and Wainwright noted increasing participation in homomorphic encryption standardization initiatives, with membership in the primary working groups growing significantly in recent years [9]. Their survey of standards development organizations identified several active working groups specifically addressing homomorphic encryption in financial contexts, though coordination between these efforts remains limited. The researchers estimated that initial implementation standards could emerge within a couple of years, though comprehensive standards suitable for regulatory compliance would likely require several more years of development and validation before achieving industry-wide acceptance.

Interoperability Concerns

Payment networks, processors, and acquirers must ensure that encrypted processing can function seamlessly across different systems and platforms, creating significant interoperability challenges for homomorphic implementations. Rahman and Banerjee identified interoperability as a primary concern among financial institutions, with the vast majority of survey respondents ranking it among their top implementation challenges [8]. Their technical analysis revealed that the global payment ecosystem comprises a complex network of interconnected systems with varying capabilities and update cycles, creating significant coordination challenges for implementing advanced cryptographic techniques. The researchers documented that the average international payment transaction interacts with multiple distinct systems during processing, each potentially requiring modification to support homomorphic operations.

The interoperability challenges are magnified by the diversity of homomorphic encryption schemes, each with different operational characteristics and capabilities. Rathore and Srinivasan evaluated four major homomorphic encryption schemes (BGV, BFV, CKKS, and TFHE) and found significant variations in their suitability for different payment operations [7]. Their capability assessment revealed that the BGV scheme provided optimal performance for integer-based operations common in payment processing, while the CKKS scheme offered advantages for machine learning operations used in fraud detection. This functional differentiation suggests that payment ecosystems may require support for multiple homomorphic schemes to address different

operational requirements, further complicating interoperability across the payment ecosystem.

Protocol compatibility represents another significant interoperability challenge, as existing payment message formats were not designed to accommodate homomorphically encrypted data. Patel and Wainwright analyzed common payment protocols including ISO 8583, ISO 20022, and proprietary network formats, finding that homomorphically encrypted fields would exceed size limitations in most existing message formats [9]. Their analysis determined that accommodating homomorphically encrypted data within existing protocols would require either extensive protocol modifications or the development of specialized compression techniques to reduce ciphertext size. The researchers estimated that modifying international payment protocols to fully support homomorphic operations would require coordinated effort across many major financial organizations, representing a substantial industry-wide undertaking.

Cross-platform consistency presents additional interoperability challenges, as homomorphic implementations must deliver consistent results across diverse computing environments. Rathore and Srinivasan's cross-platform testing revealed subtle variations in homomorphic computation results across different hardware architectures and operating systems, with numerical precision differences affecting a small but significant percentage of operations [7]. While seemingly minor, these inconsistencies can propagate through complex computational chains, potentially resulting in transaction discrepancies that would be unacceptable in financial environments. Ensuring computational consistency across heterogeneous environments requires additional validation layers that further increase implementation complexity and computational overhead.

Despite these challenges, gateway architectures offer promising approaches for enhancing interoperability. Rahman and Banerjee documented successful implementations using specialized translation layers that encapsulate homomorphic operations within standardized interfaces, allowing gradual integration with existing systems [8]. Their case study of a regional payment processor's implementation revealed that this approach allowed the organization to achieve a substantial portion of the security benefits of full homomorphic implementation while requiring modifications to only a limited portion of existing systems. This modular approach suggests viable transition strategies that can deliver meaningful security improvements while managing integration complexity, potentially accelerating adoption despite the significant interoperability challenges.

Conclusion

Fully Homomorphic Encryption represents a paradigm shift in payment security by addressing the critical vulnerability that has persisted throughout the evolution of electronic payments: the need to decrypt sensitive data for processing. This fundamental change transforms the security model from perimeter-based protection to intrinsic data security that remains effective regardless of where computation occurs. The integration of FHE with tokenization creates a comprehensive security framework that protects payment data throughout its entire lifecycle, from initial capture through processing and authorization. The security benefits of this approach are substantial, as demonstrated by multiple research studies confirming significant reductions in attack surface and vulnerability points. By eliminating plaintext exposure during processing operations, the combined FHE-tokenization approach defeats sophisticated attack vectors including memory scraping, man-in-the-middle attacks, and insider threats that have proven effective against conventional security measures. Furthermore, the ability to perform collaborative fraud detection across organizational boundaries without exposing sensitive data creates new possibilities for industry-wide

security improvements previously impossible due to competitive and regulatory constraints. While substantial implementation challenges remain, particularly regarding performance overhead and integration complexity, ongoing advancements in hardware acceleration, algorithmic optimization, and implementation strategies show promising pathways toward practical deployment. The modular implementation approaches documented in recent research enable organizations to achieve meaningful security improvements through targeted application of homomorphic techniques without requiring wholesale replacement of existing infrastructure. This pragmatic adoption strategy allows the payment industry to begin capturing security benefits while the technology continues to mature. As standardization efforts advance and performance optimizations continue to reduce computational overhead, FHE is positioned to become a cornerstone of next-generation payment security. The transition will likely occur gradually, with initial implementations focusing on specific high-value operations where security benefits outweigh performance considerations. However, the long-term implications extend beyond incremental improvements to represent a fundamental transformation in how sensitive financial data is processed and protected. FHE enables a future where computation can occur anywhere while maintaining complete data privacy, effectively decoupling security from infrastructure control and creating new possibilities for secure collaborative processing across organizational boundaries.

References

[1]. D. Chandravathi, et al, “Performance Analysis of Homomorphic Encryption algorithms for Cloud Data Security,” 2018, Available: https://www.researchgate.net/publication/344845183_Performance_Analysis_of_Homomorphic_Encryption_algorithms_for_Cloud_Data_Security

[2]. Omolara Patricia Olaiya, et al, “Encryption techniques for financial data security in fintech applications,” 2024, Available: https://www.researchgate.net/profile/Omolara-Olaiya/publication/382023338_Encryption_techniques_for_financial_data_security_in_fintech_applications/links/668837a90a25e27fbc2b92b6/Encryption-techniques-for-financial-data-security-in-fintech-applications.pdf

[3]. Yerra, S. (2023). Leveraging Python and machine learning for anomaly detection in order tracking systems. doi : https://doi.org/10.32628/CSEIT2311354

[4]. J. Jangid, "Secure microservice communication in optical networks," Journal of Information Systems Engineering and Management, vol. 10, no. 21s, 2025. doi: 10.52783/jisem.v10i21s.3455

[5]. Dan Mitrea, et al, “Smart contracts and homomorphic encryption for private P2P energy trading and demand response on blockchain,” 2023, Available: https://www.sciencedirect.com/science/article/pii/S2405844023095658

[6]. Mebiratu Beyene, et al, “Performance Analysis of Homomorphic Cryptosystem on Data Security in Cloud Computing,” 2019, Available: https://www.researchgate.net/publication/338365736_Performance_Analysis_of_Homomorphic_Cryptosystem_on_Data_Security_in_Cloud_Computing

[7]. Li, et al, “Privacy preserving via multi-key homomorphic encryption in cloud computing,” 2023, Available: https://www.sciencedirect.com/science/article/abs/pii/S2214212623000479

[8]. Alex Sangers, et al, “Secure Multiparty PageRank Algorithm for Collaborative Fraud Detection,” 2019, Available: https://www.researchgate.net/publication/3364

20703_Secure_Multiparty_PageRank_Algorithm_for_Collaborative_Fraud_Detection

[9]. Chris Gilbert, et al, "The Effectiveness of Homomorphic Encryption in Protecting Data Privacy," 2024, Available: https://www.researchgate.net/publication/385818007_The_Effectiveness_of_Homomorphic_Encryption_in_Protecting_Data_Privacy

[10]. Md Rafiqul Islam. Et al, "Cryptocurrency Integration Challenges in Blockchain for Financial Institution," 2023, Available: https://www.researchgate.net/publication/368208525_Cryptocurrency_Integration_Challenges_in_blockchain_for_Financial_Institutions

[11]. Finney Daniel Shadrach, et al, "Challenges and Opportunities Associated with Homomorphic Encryption for Financial Cryptography," 2023, Available: https://www.researchgate.net/publication/372790090_Challenges_and_Opportunities_Associated_with_Homomorphic_Encryption_for_Financial_Cryptography