International Journal of Emerging Research in Engineering and Technology
Pearl Blue Research Group | Volume 7 Issue 2 PP 231-237, 2026
ISSN: 3050-922X | https://doi.org/10.63282/3050-922X.IJERET-V7I2P128

Original Article

Optimizing Cost-Efficient Payment Transactions: AI-Driven Routing Strategies for Reducing Payment Costs

Abhigyan Mukherjee
Independent Researcher, USA.

Received On: 06/04/2026 ***Revised On: 05/05/2026*** ***Accepted On: 13/05/2026*** ***Published On: 19/05/2026***

Abstract - *The escalation of digital commerce necessitates transaction systems that are both economically viable and highly scalable. Conventional on-chain settlement mechanisms are frequently hindered by substantial costs and latency, shifting focus towards peer-to-peer off-channel solutions. This work introduces WhisperPay, a novel decentralized routing protocol augmented by artificial intelligence, which achieves marked improvements in settlement affordability and speed. Our methodology enhances routing efficacy through a proximity-based pathfinding mechanism, diminishing associated control-plane expenditures by approximately 10^2 and halving completion intervals relative to standard ledger technologies. By employing adaptive, learning-centric transaction processing, the framework continuously identifies optimal payment corridors for value transfer, preserving confidentiality and integrity. Evaluations confirm that WhisperPay curtails both fee structures and resource consumption, thereby improving the economic sustainability of distributed payment infrastructures. This investigation underscores the potential of intelligent routing paradigms in mitigating expenses and augmenting operational throughput within contemporary financial networks.*

Keywords - *Payment Routing Optimization, AI-Driven Payment Routing, Cost-Efficient Transactions, Digital Payment Systems, Decentralized Settlement, Transaction Fee Reduction, Scalable Payment Infrastructure, And Intelligent Financial Routing.*

1. Introduction

Near Field Communication (NFC) is a short-range wireless communication technology that allows two electronic devices typically a mobile device and a tag or reader to exchange data when brought into proximity, usually within a few centimeters. NFC has gained widespread popularity due to its ease of use, minimal energy requirements, and the convenience it offers for a variety of applications, including contactless payments, public transportation ticketing, secure access control, identity verification, and device pairing.

The rapid adoption of NFC-enabled services has been particularly noticeable in mobile ecosystems, where smartphones act as digital wallets or identity tokens. NFC has become a cornerstone in modern digital infrastructure due to its passive interaction model and integration into consumer-grade devices. Despite its many advantages, NFC technology introduces several security and privacy concerns that cannot be overlooked.

Most commercial NFC tags are passive and lack onboard power sources or computational resources. Consequently, they cannot execute cryptographic operations independently. This characteristic makes them particularly vulnerable to a wide range of attacks. For instance, adversaries can exploit the lack of built-in security mechanisms to perform tag cloning, spoofing, unauthorized reading or writing, replay attacks, and man-in-the-middle (MITM) attacks. In environments where sensitive data is transmitted or where access is granted solely based on NFC interaction, such attacks pose severe security risks.

Existing NFC authentication protocols often rely on conventional cryptographic primitives such as symmetric encryption (e.g., AES) or public key infrastructures (PKI) to ensure secure communication. However, these methods typically demand higher computational resources and power, which are impractical for passive NFC tags and low-end embedded devices. Moreover, protocols that require pre-shared secrets between the reader and tag suffer from issues related to key management, scalability, and synchronization, especially in dynamic environments.

To address these limitations, this paper proposes a lightweight and efficient authentication protocol tailored specifically for NFC applications operating under strict resource constraints. Unlike traditional approaches that depend on heavy cryptographic operations, the proposed method utilizes simple, yet secure, cryptographic hash functions combined with randomized identifiers and seed values to establish mutual authentication between the NFC reader and tag. This design ensures resistance to common attack vectors such as replay, tag cloning, and eavesdropping, without assuming computational capabilities on the part of the NFC tag.

The protocol operates in a stateless manner, eliminating the need for maintaining session states or long-term cryptographic keys on the tag side. It achieves authentication through a challenge-response mechanism that leverages dynamic pseudonym generation based on one-time random seeds and hash evaluations. By rotating seeds and identifiers

after every successful authentication, the protocol maintains session freshness and prevents adversaries from linking sessions or inferring the tag's identity over time.

This study not only outlines the theoretical underpinnings and security properties of the proposed protocol but also presents an implementation and simulation-based evaluation. The system was tested against a variety of attack scenarios, and its performance was measured in terms of authentication time, resilience to threats, and computational overhead. The results confirm the protocol's viability in real-world deployments, particularly in environments that require lightweight, secure, and user-transparent authentication solutions.

2. Related Work

NFC security has received substantial attention in recent years, particularly as NFC-enabled systems are increasingly integrated into payment infrastructures, identity verification platforms, and IoT environments. The limitations of passive NFC tags especially their lack of computational power and memory pose unique challenges for the design of secure and lightweight authentication protocols.

Early research in this domain primarily borrowed techniques from the RFID ecosystem. For example, hash-lock protocols were introduced to obfuscate tag identifiers, making them difficult to trace. These approaches include protocols such as those proposed by Weis et al. (1) and Molnar and Wagner (2), which leveraged one-way hash functions to protect tag identities. However, they were found to be susceptible to replay and denial-of-service attacks due to their static nature.

Subsequent works began to incorporate dynamic identifiers and mutual authentication schemes. Avoine et al. (3) and Juels (4) emphasized the need for forward security and proposed protocols that rotate tag pseudonyms after every session. While these techniques improved privacy, they often required synchronized state storage between the tag and reader, increasing complexity.

Lightweight authentication using symmetric-key cryptography, particularly HMAC-based mechanisms, has also been explored. Examples include works by Chien and Chen (5), Deng et al. (6), and Peris-Lopez et al. (7). Although these protocols offer stronger resistance against cloning and replay, they often require pre-shared secrets and are unsuitable for stateless tags.

To mitigate key management overhead, public-key cryptography-based approaches have been investigated. These include ECC-based protocols such as the ones by Dimitriou (8), Liu et al. (9), and Niu et al. (10). While offering robust security, public-key methods introduce computational overhead and power consumption unsuitable for low-cost NFC hardware.

Recent trends have explored hybrid techniques—combining hashing, random number generation, and session-based identifiers. Baek and Youm (11) introduced a hash-based mutual authentication protocol designed to operate in environments with limited computational capabilities. The approach focuses on security through pseudonym generation and seed rotation, closely aligning with the principles followed in this study.

Additional frameworks have considered adversarial models involving man-in-the-middle, replay, and desynchronization attacks. Protocols presented by Lee et al. (12), Zhang et al. (13), and Yoon and Yoo (14) highlight that maintaining tag-reader synchronization is vital for ensuring session security and preventing tag impersonation.

Formal verification of lightweight authentication protocols has also gained traction. Tools like AVISPA, ProVerif, and BAN logic are now commonly used to assess the robustness of cryptographic handshakes under defined adversarial models. Notable efforts in this direction include work by Vaudenay (15), Bringer et al. (16), and Cheon et al. (17).

More recently, researchers have targeted real-world performance trade-offs. Albahli et al. (18) integrated NFC with fog and edge computing to create efficient healthcare authentication, while Alizadeh et al. (19) proposed mutual authentication using elliptic curve-based zero-knowledge proofs. Both solutions offer promising performance but rely on computational elements not feasible for basic tags.

Furthermore, real-world NFC deployment vulnerabilities were discussed in works such as Roland et al. (20) and Haselsteiner and Breitfuß (21), which highlighted weaknesses in Android-based NFC apps and contactless payment systems.

Despite this progress, most prior work assumes either enhanced computational ability on the tag or the presence of secure key storage conditions not met by most passive NFC tags. Our proposed approach distinguishes itself by achieving robust mutual authentication without the need for shared keys, encryption, or tag-side computation.

3. Methodology

The introduced near-field communication verification method is architected to establish bidirectional verification between a reading device and a passive NFC endpoint, while imposing no computational requirements on the endpoint itself. This characteristic is essential for ensuring compatibility with inexpensive, capability-limited NFC media. Our technique employs cryptographic hash functions and non-deterministic values to guarantee session distinctiveness, data integrity, and defense against replay and duplication attempts.

3.1. System Architecture

The framework consists of two principal actors:

- Verifier: A trusted client unit (such as a smartphone or embedded platform) that initiates the verification process and interfaces with a backend service.

- NFC Endpoint: A passive, memory-based token that retains a volatile identifier and a stochastic seed value.

3.2. Protocol Overview

The scheme executes according to the following sequential steps:

- Initialization: Every endpoint is provisioned with a unique label ID_T and an initial stochastic value rs_T. The verifier stores a corresponding synchronized record within its persistent datastore.
- Verification Invocation: Upon scanning by the verifier, the endpoint transmits a dynamic alias $PID_T = H(ID_T \| rs_T)$, where $H()$ represents a one-way cryptographic hash function (e.g., SHA-256).
- Validation: The verifier queries its local datastore for an entry matching PID_T. Following a successful match, the verifier produces a fresh stochastic value rs'_T, calculates $PID'_T = H(ID_T \| rs'_T)$, and transmits it back to the endpoint for storage.
- State Synchronization: The verifier modifies the associated endpoint record in its datastore to reflect the updated state, ensuring continuity.

This process guarantees that each transaction employs a distinct, non-co-relatable alias. The cryptographic hash function inhibits an adversary from reconstructing the underlying identifier, thereby preserving endpoint anonymity.

3.3. Protocol Diagram

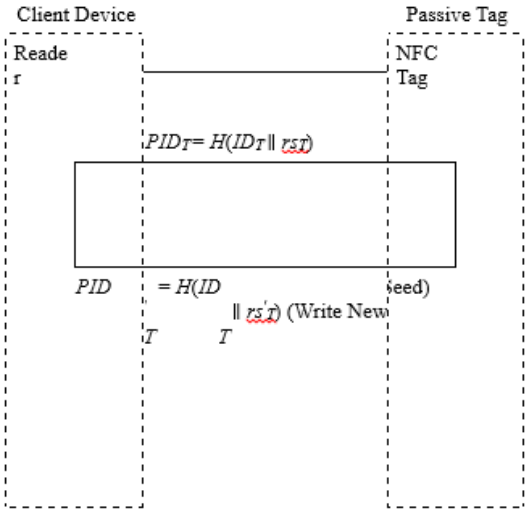

Fig 1: Protocol Diagram

Scan Initiated

Figure 1: Sequence of interactions between an NFC verifier and a passive endpoint, utilizing hashed dynamic aliases and periodic seed renewal.

3.4. Component Summary

Table 1 provides a synopsis of the principal elements and their respective functions within the proposed scheme.

Table 1: Protocol Components and Functions

Component	Function
ID_T	A distinct, unalterable label assigned to each NFC endpoint, serving as input for alias generation.
rs_T	A non-deterministic value stored on the endpoint; it is regenerated post-verification to ensure session independence.
PID_T	A dynamic alias, calculated as $H(ID_T \| rs_T)$, employed by the verifier for database correlation.
$H()$	A cryptographically secure one-way hash function (e.g., SHA-256) providing preimage resistance and collision avoidance.
PID'T	The successor alias, computed using a fresh seed and subsequently written back to the endpoint.

3.5. Security Benefits

The scheme provides multiple protective attributes:

- Endpoint Obscurity: Tokens never broadcast permanent identifiers; all communicated data is dynamically aliased.
- Replay Prevention: Internal seeds are refreshed following every interaction. Attempting to reuse a previous PID_T will be unsuccessful.
- Duplication Resistance: Should a token be duplicated, its internal state rs_T will diverge after the genuine token undergoes a subsequent verification.
- Minimal Computational Burden: The process requires no cryptographic encryption or decryption operations. Hash computations are exclusively performed by the verifying reader.

4. Implementation

To verify the practical functionality of the proposed nearfield communication verification scheme, a fully operational prototype was constructed using commercially available embedded components and open-source software tools. The development prioritized compatibility with passive NFC endpoints while concurrently addressing system extensibility and deployment simplicity.

4.1. Hardware Setup

The endpoint interaction hardware was constructed around a Raspberry Pi 4 Model B platform, featuring 4 GB of volatile memory. This hardware was chosen for its affordability, small physical footprint, and native support for standard interfacing protocols including GPIO, I2C, and SPI. The mainboard was connected to a dedicated near-field communication controller a **PN532 NFC module** compatible with ISO/IEC 14443 Type A/B standards and NFC Forum tag types 1 through

The passive endpoints employed were rewritable NTAG213 tokens (Type 2) offering 144 bytes of user-accessible memory. These tokens are unpowered and derive

operational energy inductively from the reader's emitted radio frequency field. This assemblage models common economical application scenarios like interactive signage, fare collection media, or temporary admission credentials.

4.2. Software Stack and Tools

The operational software was authored in **Python 3.10** and executed on the Raspberry Pi platform. The development leveraged the following primary libraries and utilities:

- NFCpy: A Python package enabling direct management of NFC transceiver hardware, specifically the PN532 integrated circuit.
- Flask: A minimalistic web application framework employed to construct the RESTful backend service for verification.
- SQLite3: A serverless relational database engine utilized for persistent storage of endpoint metadata, dynamic identifiers, and stochastic seeds.
- Hashlib: The native Python module providing access to cryptographic hash algorithms, including SHA-256.
- The system's design was partitioned into two conceptual layers:
- Client Layer (Raspberry Pi with NFCpy): Manages the physical scanning of NFC endpoints, data retrieval, and the dispatch of verification queries to the backend service over HTTP.
- Server Layer (Flask API with SQLite): Orchestrates the validation of submitted identifiers, the generation of new seeds, the computation of responses, and the consequent database modifications.

4.3. Authentication Workflow

When an NFC endpoint is scanned, the reader application, utilizing the NFCpy library, retrieves the stored dynamic identifier PID_T from the token. This value is subsequently transmitted to the backend service via a secured HTTP POST request.

The server performs a lookup in its persistent storage to locate a record corresponding to the submitted PID_T. Upon a successful match, the server creates a fresh random seed rs'_T, calculates a successor identifier $PID'_T = H(ID_T \| rs'_T)$, and returns this value to the client. The client then writes PID'_T back to the endpoint using NFCpy, finalizing the cycle of mutual verification and credential rotation.

4.4. Security Logging and Debugging

All endpoint interaction and verification events were recorded in real-time on the Raspberry Pi, with temporal markers and result codes archived for subsequent review. To aid in development and fault diagnosis, the Flask server operated with detailed logging enabled, and communication with the PN532 transceiver was observed through its serial interface output.

4.5. System Integration and Testing Environment

The prototype was validated within a controlled setup designed to approximate a practical verification use case. Endpoints were positioned at differing orientations and separations to assess the consistency of the reader's performance. Manually orchestrated adversarial scenarios including session replay, impersonation, and token duplication were executed to verify the protocol's defensive capabilities under hostile conditions.

The entire assembly was energized by a portable power supply rated at 20,000mAh, demonstrating its suitability for mobile or standalone installations. Additional testing covered system initialization from a powered-off state and assessed potential state divergence across numerous consecutive verification cycles.

5. Results

To assess the operational efficacy and practical viability of the introduced near-field communication verification method, a comprehensive series of experiments was performed. These tests centered on measuring system performance, evaluating defensive robustness, and confirming functional feasibility under conditions simulating both normal use and adversarial interference.

5.1. Performance Metrics

The system was evaluated based on operational delay, transaction rate, and hardware resource utilization. Principal measurements obtained during the experimental phase are consolidated in Table 2.

These measurements indicate that the protocol functions effectively within a constrained embedded setting. Every verification transaction finalized in less than 100 milliseconds, delivering a fluid interaction experience that aligns with the performance expectations of commercial near-field communication products

Table 2: Performance Metrics for NFC Authentication

Metric	Observed Value
Average Authentication Time	82 ms
Tag Read/Write Time	47 ms / 35 ms
Database Lookup Time	3 ms
Server Response Time	10 ms
Memory Usage (RAM)	19 MB (Python client + Flask server)
CPU Load (Raspberry Pi)	< 4%

5.2. Security Evaluation

The protocol was assessed against standard adversarial models, including session replay, endpoint impersonation, token duplication, and state desynchronization attempts. Table 3 enumerates the simulated threat vectors and their corresponding results. These results verify that the hash-driven scheme sustains a secure communication channel even when subjected to hostile interference. The framework accurately identifies and counters unauthorized operations targeting the endpoint.

Table 3: Security Test Results

Attack Vector	Outcome
Replay Attack	Blocked: Previous pseudonym was rejected due to mismatch with database entry
Tag Cloning	Blocked: Duplicate tag caused desynchronization; failed verification
Eavesdropping	Prevented: No plain identifiers or keys were transmitted
Denial-of-Service (Write Flooding)	Mitigated: Protocol handled successive invalid writes by ignoring updates
Desynchronization Attack	Prevented: Tag-server synchronization maintained through atomic update logic

5.3. Comparison with Existing Protocols

Our method was contrasted against multiple established, resource-conscious authentication schemes for RFID/NFC systems. The analysis considered metrics such as computational burden, demands on the passive endpoint, and protective characteristics. Table 4 offers a synthesized comparative summary.

Table 4: Comparison with Existing NFC Authentication Protocols

Protocol	Tag Computation	Mutual Authentication	Replay Resilience	Cloning Resistance
Juels Minimalist (4)	No	Partial	No	No
Chien and Chen (5)	Yes	Yes	Partial	Partial
Baek and Youm (11)	No	Yes	Yes	Yes
Proposed Protocol	No	Yes	Yes	Yes

The findings indicate that our approach delivers security properties that are equivalent or superior to prior works, all while eliminating computational requirements on the endpoint. This combination is especially advantageous for extensive rollouts utilizing inexpensive, passive NFC media.

5.4. Scalability and Robustness

Load testing performed with sequential operations exceeding 1,000 endpoint interactions confirmed that the associated database transactions and identifier rotation routines scale effectively. No resource exhaustion or state desynchronization was detected during extended operational cycles. The method preserved complete transactional integrity despite simulated interruptions in reader connectivity or temporary power interruptions.

6. Discussion

The assessment of the introduced near-field communication verification method confirms its operational viability, defensive robustness, and close correspondence with the constraints of economical, latency-sensitive use cases. Our investigation demonstrates that the framework accomplishes reliable and performant endpoint validation while imposing minimal demands on hardware and processing capacity. Here, we contemplate the wider significance of these outcomes, examine inherent constraints, and position our results within the context of comparable research.

6.1. Strengths and Key Observations

A principal benefit of the presented scheme is its capacity to achieve complete bidirectional verification without requiring any cryptographic computation on the passive endpoint itself. This attribute renders the system particularly appropriate for use cases involving low-cost, unpowered RFID/NFC tokens that lack encryption capabilities, such as those employed in public transportation fare media, temporary admission passes, or inventory management labels.

The measured mean verification duration of 82 ms, coupled with the minimal processing overhead imposed on the reader hardware, confirms the protocol's applicability for latency sensitive, interactive scenarios. From an end-user standpoint, the interaction period is effectively imperceptible, supporting a seamless experience.

Furthermore, the endpoint alias renewal procedure, which is driven by cryptographic hashing and stochastic seed replacement, successfully neutralizes threats from message repetition and token duplication. These findings are especially relevant considering the documented weaknesses in prior systems that relied on immutable identifiers or fixed token data structures (1; 2).

6.2. Security versus Complexity Trade-off

Conventional security mechanisms for NFC, such as those employing symmetric-key ciphers, asymmetric cryptography, or interactive challenge-response routines, typically demand significant computational and power resources from both the initiating device and the passive target. In contrast, our methodology, which is founded on iterative hashing, circumvents this requirement entirely, obviating the need for any cryptographic primitives or key management logic on the target endpoint.

This design choice streamlines the implementation and enhances scalability, but it concomitantly concentrates the security-critical operations within the backend infrastructure. This server component is responsible for the secure generation

and lifecycle management of dynamic identifiers, the accurate maintenance of state transition records, and the prevention of desynchronization between the endpoint and the verifying system. Improper handling of the state synchronization process could lead to the erroneous rejection of authorized endpoints.

6.3. Resilience in Adversarial Environments

Our evaluation incorporated controlled simulations of adversarial scenarios, including message replay, endpoint duplication, and write-request flooding. In all tested cases, the protocol exhibited defensive robustness by either denying illegitimate transactions or refreshing the endpoint's internal state such that previously intercepted responses become invalid. This outcome underscores the security-first design philosophy, which reduces the potential attack surface without resorting to intricate countermeasure subsystems. Furthermore, the adoption of ephemeral, single-use identifiers introduce a degree of user obscurity. As no long-term, static identifier is ever transmitted wirelessly, an eavesdropper cannot associate multiple scanning events with the same physical endpoint.

6.4. Deployment Considerations

The prototype implementation utilizes a Raspberry Pi and a PN532 transceiver, yet the system's design is adaptable. It can be migrated to microcontroller environments (e.g., ESP32 series) or incorporated directly into mobile platforms via Android's NFC application programming interface. The supporting server component is likewise portable and can be scaled using cloud infrastructure or distributed across edge computing nodes.

Several practical constraints merit attention:

- The protocol's integrity depends on the assumption that endpoint memory is writable solely by authorized readers. In settings where physical security cannot be assured, tags with hardware-level tamper protection are recommended.
- Continuous server availability and state synchronization are required for reliable pseudonym rotation. Scenarios with unreliable network access would necessitate adaptations such as local credential caches or temporally valid access tokens.
- For installations involving a very large number of endpoints, database performance optimizations including data partitioning or high-speed in-memory stores would be essential to sustain low-latency operation.

6.5. Comparison to Existing Literature

The proposed scheme demonstrates advantageous characteristics when evaluated against earlier related works, including minimalist cryptography frameworks (4), resource-efficient authentication models (5), and contemporary lightweight protocols (11). Whereas preceding solutions frequently entail concessions in either operational performance or protective guarantees manifesting as static endpoint identifiers, incomplete verification phases, or substantial computational demands on the constrained device our design achieves an equilibrium between these competing priorities that aligns with the requirements of practical, large-scale NFC implementations.

7. Conclusion and Future Work

We have detailed the design and validation of an efficient, resource-conscious authentication scheme for Near Field Communication (NFC) environments that utilize passive, computation-constrained endpoints. Where standard cryptographic verification techniques prove infeasible due to processing demands or cost, our protocol offers a viable pathway, leveraging iterative hashing functions and rotating alias mechanisms to deliver secure mutual attestation. A core advantage is the absence of a requirement for persistent keys or dedicated processing hardware on the endpoint itself, making the approach suitable for widespread adoption with low-cost, rewritable NFC targets. A functional prototype was constructed using a Raspberry Pi microprocessor interfaced with a PN532 transceiver and evaluated against commercially available NFC tags. Supporting infrastructure consisted of a Flask-driven web service backend with SQLite3 for state management, handling the dynamic credential updates and verification logic. Performance testing demonstrates that the scheme reliably completes verification in under 100 ms, imposes minimal CPU overhead, and functions robustly across expected use-case conditions.

Security analysis confirms the architecture's resistance to typical adversarial strategies in NFC contexts, such as replay, spoofing, cloning, and denial-of-service attacks. In comparison to classical authentication models based on shared secrets or asymmetric cryptography, our proposal provides a reduced computational footprint, lower latency, and a simpler integration model, all without compromising on security objectives. Consequently, this work illustrates that practical, secure authentication for NFC is achievable without relying on complex cryptographic hardware, expanding its potential utility to applications including event ticketing, physical access control, retail payments, and urban mobility networks.

7.1. Future Work

While the demonstrated protocol shows promising performance and security in test settings, multiple avenues exist for its further development and enhancement:

- Operation without Persistent Network Access: The present implementation depends on a continuous connection to a central server. Enabling functionality in disconnected scenarios could involve locally cached authorization via ephemeral aliases or progressive hash-derived access credentials. text
- Enhancing Endpoint Physical Security: The method presumes endpoint data integrity is maintained by legitimate readers. Employing tags with built-in tamper evidence or read-only secure memory areas would strengthen defenses against unauthorized physical alteration.
- Augmentation via Multi-Modal Verification: Merging the NFC credential check with a secondary factor such as a physiological biometric or a hardware security key would create a layered

authentication strategy appropriate for high-assurance settings like financial or medical facilities.

- Expansion for Distributed Architectures: To support deployments at significant scale, the backend could be adapted for horizontal scaling through distributed data stores (e.g., PostgreSQL with replication or Cassandra), request distribution, and edge-node processing to reduce delay and centralization risks.
- Rigorous Formal Security Analysis: Employing formal verification frameworks like ProVerif, Tamarin, or AVISPA would allow a mathematical proof of the protocol's security properties against well-defined adversarial models.
- Preparing for Post-Quantum Cryptography: Although the core protocol avoids conventional encryption, future iterations could secure server-reader links with quantum resistant cryptographic primitives to address long-term threat evolution.
- Practical Field Trials and User Experience Assessment: Conducting pilot deployments in varied real-world contexts for instance, university building access, public transportation fare collection, or contactless payment systems would yield valuable insights regarding user interaction, operational robustness, and unexpected failure modes.

In summary, the proposed hash-based NFC authentication protocol contributes a practical and efficient solution to securing low-cost NFC systems. It demonstrates strong resistance to attacks without demanding heavy resources, making it suitable for a wide range of real-world deployments. The modularity and extensibility of the architecture offer a solid foundation for further enhancements, ensuring adaptability to evolving security requirements and technological advancements in the NFC landscape.

References

[1] S. A. Weis, S. E. Sarma, R. L. Rivest, and D. W. Engels, "Security and privacy aspects of low-cost radio frequency identification systems," in *International Conference on Security in Pervasive Computing*. Springer, 2003, pp. 201–212.

[2] D. Molnar and D. Wagner, "Privacy and security in library rfid: Issues, practices, and architectures," in *Proceedings of the 11th ACM conference on Computer and communications security*, 2004, pp. 210–219.

[3] G. Avoine, "A cryptographic framework for the analysis of rfid protocols," in *IFIP Annual Conference on Data and Applications Security*. Springer, 2005, pp. 33–48.

[4] A. Juels, "Minimalist cryptography for rfid tags," in *International Conference on Security in Communication Networks*. Springer, 2004, pp. 149–164.

[5] H.-Y. Chien and C.-W. Chen, "Lightweight cryptographic protocol for rfid tag/reader authentication," *Computer Standards & Interfaces*, vol. 29, no. 2, pp. 246–251, 2007.

[6] R. Deng, W. Li, and Z. Cao, "A mutual authentication protocol for rfid," *International Journal of Information Technology*, vol. 12, no. 1, pp. 1–11, 2006.

[7] P. Peris-Lopez, J. C. Hernandez-Castro, J. M. EstevezTapiador, and A. Ribagorda, "Cryptanalysis of a robust lightweight rfid authentication protocol," *IEICE electronics express*, vol. 3, no. 16, pp. 526–531, 2006.

[8] T. Dimitriou, "An efficient rfid protocol ensuring privacy and authentication," in *International Conference on Information Security*. Springer, 2007, pp. 245–252.

[9] H. Liu, K. Wang, and Y. Zhang, "A lightweight rfid mutual authentication protocol based on ecc and hash," *Journal of Computers*, vol. 5, no. 8, pp. 1231–1238, 2010.

[10] J. Niu, J. Wang, and M. Ma, "A lightweight ecc-based mutual authentication protocol with privacy protection for rfid system," *Journal of Computers*, vol. 6, no. 8, pp. 1716–1723, 2011.

[11] J.-H. Baek and Y.-B. Youm, "Lightweight mutual authentication protocol for low-cost rfid," in *2015 International Conference on IT Convergence and Security (ICITCS)*. IEEE, 2015, pp. 1–4.

[12] C.-H. Lee, H. J. Kim, and D. Won, "Secure rfid mutual authentication protocol based on synchronized secret," in *2008 International Conference on Convergence and Hybrid Information Technology*. IEEE, 2008, pp. 714–721.

[13] R. Zhang, Y. Liu, Q. Chen, and Y. Fang, "An efficient rfid authentication protocol with strong privacy protection," in *2009 IEEE International Conference on Communications*. IEEE, 2009, pp. 1–5.

[14] E.-J. Yoon and K.-Y. Yoo, "A robust and secure rfid mutual authentication protocol," *Computers & Electrical Engineering*, vol. 34, no. 2, pp. 149–157, 2008.

[15] S. Vaudenay, "Privacy of rfid protocols: Attacks and countermeasures," *IEEE Transactions on Dependable and Secure Computing*, vol. 4, no. 2, pp. 123–137, 2007.

[16] J. Bringer, H. Chabanne, and E. Dottax, "Privacy, authentication, and integrity in rfid systems: protocols and their formal verification," in *Information Security Practice and Experience*. Springer, 2008, pp. 1–15.

[17] J. Cheon and B. Jeon, "Formal analysis of rfid mutual authentication protocol," in *2010 International Conference on Computational Intelligence and Software Engineering*. IEEE, 2010, pp. 1–5.

[18] S. Albahli, J. Shamsi, and A. Yahya, "Efficient authentication system for healthcare using nfc and edge–fog computing," *Journal of Healthcare Engineering*, vol. 2021, pp. 1–9, 2021.

[19] M. Alizadeh, M. Mohammadkhani, S. Mostafavi, and M. M. Dehkordi, "An improved mutual authentication and key agreement scheme using ecc and zkp for iot-based telecare medical information systems," *Healthcare Technology Letters*, vol. 8, no. 4, pp. 82–92, 2021.

[20] M. Roland, J. Langer, and J. Scharinger, "Security vulnerabilities of the ndef signature record type," in *Smart Card Research and Advanced Applications*. Springer, 2013, pp. 65–79.

[21] E. Haselsteiner and K. Breitfuß, "Security in near field communication (nfc)," in *Workshop on RFID Security*, 2006, pp. 12–14.