JCST Journal of Data Acquisition and Processing
ISSN: 1004-9037
https://sjcjycl.cn/
DOI: 10.5281/zenodo.754742

RESILIENT FINANCIAL TECHNOLOGY FRAMEWORK WITH TRANSACTION SECURITY AND FAIRNESS

Nada Mohammed i Alnuhait[1] and Shaik Shakeel Ahamad[2]
[1&2]Department of Information Technology, College of Computer and Information Sciences
Majmaah University, Al-Majmaah, 11952, Saudi Arabia

Abstract— This paper proposes a secure and smart architecture for a Resilient Financial Technology Framework with Transaction security (RFTTS). Proposed framework ensures end to end payment processing and Transaction security. Proposed framework overcomes reverse engineering attacks as it implements Defense-in-Depth strategy. Proposed framework ensures security of data at rest and during the transit. Proposed framework collects digital evidence from the memory of the smartphone, Bank servers and from the networks using digital forensics tools. The proposed framework meets with the PCI-DSS (Payment Cards Industry Data Security Standard) regulation.

Finally, we have successfully implemented our protocol using kotlin language in Android Studio, with two Mobile Payment Applications (MPA) and POS Payment Application (PPA), Elliptic Curve Digital Signature Algorithm (ECDSA) is used and Advanced Encryption Standard (AES) with GCM (Galois/Counter Mode) mode is used for encryption and decryption of Customer Payment Data at MPA and PPA.

Keywords— Resilient Financial Technology Framework with Transaction security (RFTTS); SVO logic; Point-to-Point encryption (P2PE); Scyther tool; PCI-DSS regulation; Transaction Security; Reverse Engineering attacks; Digital Forensics tools; PCI-DSS (Payment Cards Industry Data Security Standard);

Introduction

Fin-Tech, short for financial technology, encompasses the use of technology in financial operations, such as conducting money transactions for various financial activities including both corporate and consumer interactions. It simplifies, streamlines, enhances accessibility, and generally reduces the cost of financial transactions for both users and corporations. Point-of-sale (PoS) based payments have increased significantly as a result of the increasing usage of smartphones. As a result, hackers that target sensitive information like credit card numbers, order details, and personal data have become more interested. All the entities inside the payment network are susceptible. Vulnerability affects every entity in the payment ecosystem, including banks, retailers, payment gateways, and insurance companies. Financial institutions have to abide by the PCI DSS (Payment Card Industry Data Security) guidelines.

The Payment Card Industry Data Security Standard (PCI DSS) holds significant importance within the banking sector due to its ability to efficiently reduce the likelihood of data breaches, protect customer information from fraudulent activities and theft, foster customer confidence, reduce operational costs, raise employee awareness regarding security protocols, fortify security controls, enhance the bank's reputation, and shield against financial penalties. Point-of-sale (PoS) systems frequently confront comparable vulnerabilities and hazards.

When confronted with operating systems and devices, such as Linux and Windows. Point of Sale (PoS) systems are susceptible to a variety of common attacks, such as keylogging Trojans, login attempt replication, and brute force methods. Critical for ensuring the security of business transactions, encryption from point to point is presently absent from PoS-based mobile payment frameworks. By selecting an all-encompassing electronic commerce solution that concurrently operates as an acquirer, gateway, and processor, merchants can guarantee enhanced efficiency and precision in the processing of their transactions. This, consequently, increases rates of acceptance and contributes to the overall progress of the organization.

Motivation

The main motivations for this research work are as follows:

The cybersecurity market is anticipated to attain a valuation of $300 billion worldwide by 2027, driven primarily by developments in cloud computing, network security and privacy, and the telecommunications sector [2]. The Abu Dhabi Commercial Bank and the National Bank of Fujairah experienced website disruptions due to the most recent DDoS attacks [3].

The significant surge in PoS attacks and regulatory concerns have resulted in a substantial expansion of the PoS security market. It is projected that this market will reach USD 6.1 billion by 2027, up from USD 4.0 billion in 2022. However, the current state of PoS security solutions is impeding the growth of this market [1].

Payment solutions provide convenient and flexible payment options, enabling transactions to be conducted at any location and at any given moment. Mobile Payment Applications (MPAs) and Point of Sale (PoS) Payment Applications (PPAs) play a crucial role in the effective execution of online commerce solutions. The authentication process of Point of Sale (PoS) based payments is conducted on a public channel, making it susceptible to many forms of attacks. To effectively counter any kind of attack, it is essential to develop a payment architecture that is both safe and resilient, including security measures from the initial design stage.

Limitations in the existing literature:

Contributions made: Following are the contributions made by this research work

- We propose a secure and smart architecture for a Resilient Financial Technology Framework with Transaction security.
- Proposed framework ensures end to end payment processing.
- Proposed framework ensures Transaction security.
- Proposed framework overcomes reverse engineering attacks.
- Proposed framework implements Defence-in-Depth strategy.
- Proposed framework ensures security of data at rest and during the transit.
- Proposed framework collects digital evidence from the memory of the smartphone, Bank servers and from the networks using digital forensics tools.
- The proposed framework meets with the PCI-DSS regulation.
- The proposed protocol is formally verified using SVO logic and scyther tool.

This article is organized as follows: Section II presents the related work. Section III presents proposed RFTTS framework. Section IV presents BAN logic based formal verification, Section V presents Threat Modeling, Section VI presents an experimental result, and section

VII compares RFTTS with the related works. Section VIII provides discussion of RFTTS framework, and Section IX concludes the paper.

RELATED WORK

[4] proposes a secure and efficient multi-factor authentication algorithm for mobile money applications where PIN, OTP, and biometric fingerprints authenticate users, but the mobile money applications used in this work are vulnerable to reverse engineering attacks. In their research work, Vincent et al. [5] introduced an enhanced security protocol, identity-based elliptic-curve cryptography (ECC), for a mobile payment system. By employing the American standard code for information interchange (ASCII) values, the system converts all input text into elliptic curve points via a payment gateway during verification. Encrypted payment information is stored on the gateway; however, decryption requires the decryption key provided by the merchant. The assessment outcome for the suggested methodology has validated the system's ability to operate efficiently with limited resources. For low-resource mobile devices, the computational capacity is effective in terms of battery life and quicker encryption and decryption keys. IMEI with ECC is resistant to identity theft and provides integrity, confidentiality, privacy, and non-repudiation. Nevertheless, the payment gateway encrypts smartphone’s data which is not acceptable. [6] proposes an Efficient and Secure NFC Authentication for Mobile Payment Ensuring Fair Exchange Protocol but this work does not resist reverse engineering attacks. Authors of [7] proposes an improved NFC device authentication protocol using asymmetric encryption algorithm, symmetric encryption algorithm, hash function, timestamp but the proposed work fails to explain how the proposed protocol ensures security and withstands the listed attacks. Authors of [8] presents a framework for mutual authentication between a user device and a point of sale (POS) machine using magnetic secure transmission (MST) to prevent the wormhole attack in Samsung pay, but the proposed work fails to provide security and safety of the keys and moreover there is no clarity how the applications withstand reverse engineering attacks. Authors of [9] presents a new NFC mobile payment protocol using improved GSM based authentication, reduces the number of required key-pairs for authentication. But this work fails to provide security and safety of the keys and moreover there is no clarity how the applications withstand reverse engineering attacks. Authors of [10] presents a Fully Authentication Services Scheme for NFC Mobile Payment Systems, but this work fails to ensure end to end security, fails to provide security and safety of the keys and moreover there is no clarity how the applications withstand reverse engineering attacks.

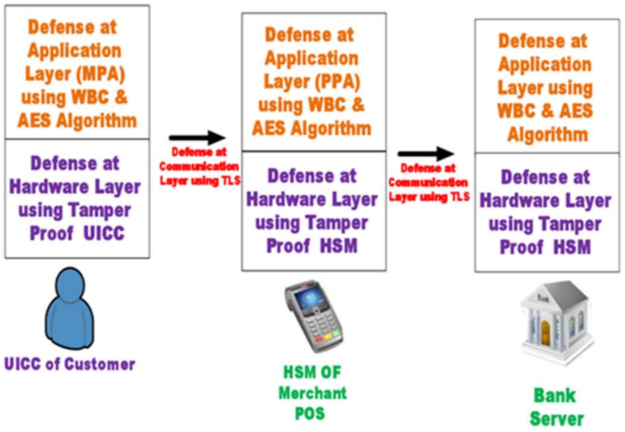
Problem Statement

PROPOSED RFTTS FRAMEWORK

Architecture of the proposed framework

Customer (C), Merchant (M) and Bank (B) are the players in Secure PoS (RFTTS) Framework. Merchant (M) contains POS machine and NFC-enabled POI Device. POS Machine contains Secure Element, POS Application, Application Memory, Payment Client Application and Data Storage. Customer’s NFC-enabled Smart Phone contains SE. The bank has Trusted execution environment (TEE) which is trusted and Applications are isolated and the Keys cannot be compromised. Following are the four locations in which RFTTS framework keeps the data secure

- Data in Memory: When the payment application processes an authorization or settlement, it performs various manipulations with the payment card data in the
- memory of the hosting computer (usually the RAM of the POS machine).
- Data at Rest: MPA and PPA keeps transaction data secure either temporarily or permanaently on the hard drive of Bannk and Merchant.
- Data in Transit: Whenever the transaction data is in transit, the data should not be compromised.
- Integrity of the Application:The integrity of both MPA and PPA should not be compromised i.e. these applications needs to with stand reverese engineering attacks from intruders. .


Defense at Application Layer (MPA) using WBC & AES Algorithm
Defense at Hardware Layer using Tamper Proof UICC
Defense at Communication Layer using TLS
Defense at Application Layer (PPA) using WBC & AES Algorithm
Defense at Hardware Layer using Tamper Proof HSM
Defense at Communication Layer using TLS
Defense at Application Layer using WBC & AES Algorithm
Defense at Hardware Layer using Tamper Proof HSM
UICC of Customer
HSM OF Merchant POS
Bank Server

RFTTS framework uses point-to-point encryption (P2PE), as this method ensures encryption on the device and then allows the encrypted data for transmission to be processed by the

NOTATION	FULL FORM	NOTATION	FULL FORM	NOTATION	FULL FORM
PCIDSS	Payment Card Industry Data Security Standard	**TEE**	Trusted Execution Environment	**OI**	Order Information
PoS	Point of Sale	**TSM**	Trusted Service Manager	**H(OI)**	Hashed Order Information
SE	Secure Element	**CA**	Certifying Authority	$\mathbf{SK_{CB}}$	Shared Symmetric key between 'C' & 'B'
UICC	Universal Integrated Circuit Card	**TL**	Trust Levels	$\mathbf{T_C}$	Time Stamp of Customer
MPA	Mobile Payment Application	**M**	Merchant	$\mathbf{N_C}$	Nonce of
PPA	PoS Payment Application	**C**	Customer	$\mathbf{SK_{MB}}$	Shared Symmetric key between 'M' & 'B'
ECDSA	Elliptic Curve Digital Signature Algorithm	**RFTTS**	Smart and Secure Point of Sale	$\mathbf{N_M}$	Nonce of Merchant
AES	Advanced Encryption Standard	**P2PE**	*Point-to-Point Encryption*	$\mathbf{T_M}$	Time Stamp of Merchant

NFC	Near-Field Communication	**NFC**	Near Field Communication	**LOC_C**	Location of Customer
PI	Payment Information	**$(PI)SK_{CB}$**	Payment Information encrypted using Symmetric key shared between 'C' & 'B'	**LOC_M**	Location of Merchant
AMT	AMOUNT	TransID	Transaction Identity		

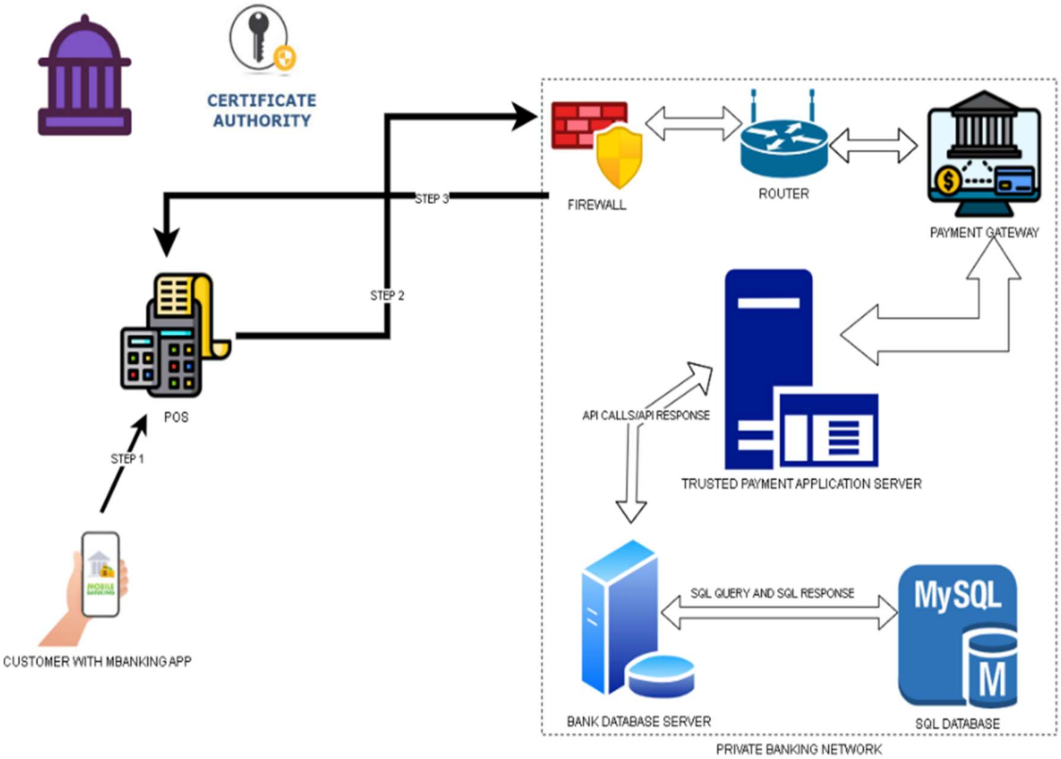


Table 1:NOTATIONS

third-parties for processing. RFTTS framework hardens the MPA and PPA applications by obfuscating, by digitally

Non-Repudiation of Transactions: The proposed protocol uses symmetric encryption to encrypt message. It can satisfy the non-repudiation property by keeping evidence from each message that each party performed in each transaction. To show that our protocol satisfies the non-repudiation of transactions. In the message M1, it can be seen that N and P share the session key SKN-Pj, but P cannot generate this message h(IDN, T1, Request, SKN-Sj) by itself because the session key SKN-Sj is shared between N and S. Only N knows both SKN-Sj and SKN-PSj, hence N cannot refuse that it did not originate this message as the possession of SKN-Pj demonstrates clearly that only N can generate this message h(IDN, T1, request, h(IDN, T1, request, SKN-Sj), {hN-TP}SKN-TPj, SKN-Pj).

Fairness:

signing, updating and patching these applications (MPA and PPA). In addition to these safety measures to the MPA and PPA applications, RFTTS framework adds dynamic library to these applications, this method is called application wrapping.

Proposed Protocol

Step 1: Customer (C) selects items from the super market and reaches the Merchant (M). ‘M’ contains ‘PoS’, which interacts with the ‘C’. ‘C’ uses his mobile payment application and sends the following message to the ‘PoS’

Step 1: $\mathrm{C} \rightarrow \mathrm{M}: \{\mathrm{MS1}\}$

MS1: $\{\mathrm{T_C, N_C, ID_C, ID_M, H(OI)_C, OI_C, (PI)SK_{CI}, LOC_C}\}$

Step 2: Merchant (M) sends ‘MS2’ to the Payment Gateway (PG) after successfully verifying the received ‘MS1’ from the Customer (C).

Step 2: $\mathrm{M} \rightarrow \mathrm{PG}: \{\mathrm{MS2}\}\mathrm{SK_{MPG}}$

MS2: $\{\mathrm{T_C, N_C, T_M, N_M, ID_C, ID_M, H(OI)_C, H(OI)_M, TranID_M, (PI)SK_{CI}, LOC_C, LOC_M}\}$

Step 3: PG checks $\mathrm{H(OI)_C}$ and $\mathrm{H(OI)_M}$ if they are equal; verifies the timestamps and nonce of both the customer and merchant; verifies the location of both the customer and merchant;
If all the verifications are successful, PG forwards MS3 to the Issuer (I) through Private Banking Network(PBN); PBN is a secure banking network which is secure so the messages exchanged among them are not encrypted.

Step 3: $\mathbf{PG} \rightarrow \mathrm{I}: \{\mathrm{MS3}\}$

MS3: $\{\mathrm{ID_C, ID_M, H(OI)_C, H(OI)_M, TranID_M, (PI)SK_{CI}}\}$

Step 4: Issuer (I) checks $\mathrm{H(OI)_C}$ and $\mathrm{H(OI)_M}$ if they are equal; Decrypts the $\mathrm{(PI)SK_{CI}}$; if the customer has sufficient funds in his/her account Issuer (I) approves the transaction else it disproves the transaction. Issuer (I) sends MS4 to the Payment Gateway (PG).

Step 4: $\mathbf{I} \rightarrow \mathrm{PG}: \{\mathrm{MS4}\}$

MS4: {APPROVES or DISAPPROVES, TransID}

Step 5: Payment Gateway (PG) forwards the received message from the Issuer (I) to the Merchant (M) and the Customer (C).

Step 5: $\mathbf{PG} \rightarrow \mathrm{M\ and\ C}: \{\mathrm{MS5}\}$

MS5: {APPROVES or DISAPPROVES, TransID}

Algorithm 1: Credential Generation, Safety and Security of the Keys

```
Step 1: Customer and Merchant (POS) generates his/her public and private keys in their
respective Secure Elements (SE) using Mobile Public Key Infrastructure (MPKI).
IF   {
Customer and Merchant (POS) implements MPKI}
        Ensures End to End Security of the messages
              Else {
        Messages can be compromised
              Exit}

Step 2: IF   {
MB Application (MBA) in the SE of Customer and Merchant (POS) implements WBC
mechanism}
        Ensures safety of the keys in the MBA
              Else {
         Keys in the MBA can be compromised
              Exit}
```

‘SVO LOGIC’ BASED FORMAL VERIFICATION

Authentication Goals for Payment Gateway ‘PG’

THREAT MODELING

In RFTTS framework threat modeling is classified in three steps

Assets and access points identification and the trust levels: An asset is a valuable thing owned by a player of RFTTS framework, and the adversaries wants to manipulate it. Access points are the interfaces through which the adversaries try to can interact with the system in order to gain access to assets. Intruders use access points to enter into the system. There are different levels of trust defined by boundaries.

List of Assets in our proposed RFTTS framework: Mobile Payment Application (MPA), Smart Phone, Point of Sale (PoS), PoS Payment Application (PPA), TEE (Trusted Execution Environment) in the merchant side.

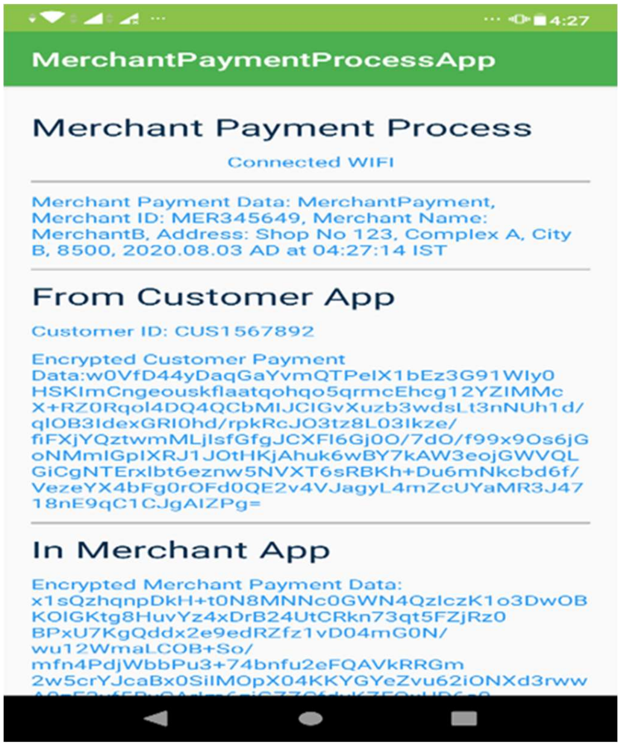
List of Access Points (AP) in our proposed RFTTS framework: Mobile Payment Application (MPA), Smart Phone, Point of Sale (PoS), PoS Payment Application (PPA).

Trust Levels (TL) in RFTTS framework: There are 3 trust boundaries in RFTTS framework

Customer and Device boundary: Customer and Smart phone boundary is between Customer and the MPA in the SE (Secure Element) of the smartphone.

NFC (Near Field Communication) boundary: NFC boundary is between Customer's smartphone and the M's PoS, Customer encrypts the messages using the shared symmetric key between 'C' and 'B' ensuring application security.

CorpNet Trust boundary: CorpNet Trust boundary is between the Merchant (M) and the M's database and Bank (B) and its database, messages are protected using TLS protocol.



Merchant Payment Process

Connected WIFI

Merchant Payment Data: MerchantPayment, Merchant ID: MER345649, Merchant Name: MerchantB, Address: Shop No 123, Complex A, City B, 8500, 2020.08.03 AD at 04:27:14 IST

From Customer App

Customer ID: CUS1567892

Encrypted Customer Payment Data:w0VfD44yDaqGaYvmQTPelX1bEz3G91WIy0HSKImCngeouskflaatqohqo5qrmcEhcg12YZIMMcX+RZ0Rqol4DQ4QCbMIJCIGvXuzb3wdsLt3nNUh1d/qIOB3IdexGRI0hd/rpkRcJO3tz8L03Ikze/fiFXjYQztwmMLjIsfGfgJCXFI6Gj0O/7dO/f99x9Os6jGoNMmIGpIXRJ1JOtHKjAhuk6wBY7kAW3eojGWVQLGiCgNTErxlbt6eznw5NVXT6sRBKh+Du6mNkcbd6f/VezeYX4bFg0rOFd0QE2v4VJagyL4mZcUYaMR3J4718nE9qC1CJgAIZPg=

In Merchant App

Encrypted Merchant Payment Data: x1sQzhqnpDkH+t0N8MNNc0GWN4QzlczK1o3DwOBKOIGKtg8HuvYz4xDrB24UtCRkn73qt5FZjRz0BPxU7KgQddx2e9edRZfz1vD04mG0N/wu12WmaLCOB+So/mfn4PdjWbbPu3+74bnfu2eFQAVkRRGm2w5crYJcaBx0SiIMOpX04KKYGYeZvu62iONXd3rww

Threat Modeling of RFTTS farmework

Recognize and Rank all the possible threats: Threats are recognized by examining the assets and access points in the RFTTS framework which compromise the security properties such as authentication, confidentiality, non-repudiation, availability and integrity.

Discover solutions and make mitigation plan: After recognizing the assets and threats there should be solutions to overcome these threats.

Solutions for Spoofing: Spoofing is not possible in RFTTS framework as all the entities store their credentials in the SE and TEE.

Solutions for Tampering: Tampering is not possible in RFTTS framework as all the entities exchange only encrypted messages among themselves.

Solutions for Repudiation: RFTTS employs Auditing Manager (AM), which works in coordination with CA.

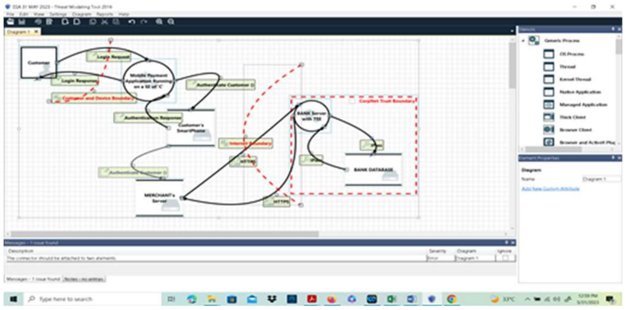
Solutions for Information Disclosure: Information disclosure is not possible in RFTTS framework as all the entities exchange only encrypted messages among themselves which ensures confidentiality.

Solutions for Denial of service: RFTTS framework uses “Fortguard Anti-DDoS” tool in order to overcome Denial of Service attacks.

Solutions for Elevation of privilege: End to end security which involves application and communication security will be able to overcome attacks in order to elevate the privileges.

EXPERIMENTAL SETUP AND RESULTS

RFTTS is implemented in Android Studio using Kotlin language.



The connector should be attached to two elements.

Experimental Results of RFTTS framework

COMPARISON WITH RELATED WORK

This section presents the comparative analysis of our proposed RFTTS framework. Table compares RFTTS framework with the related works discussed in the section 2. RFTTS framework has the best features than the features discussed in related works.

Research Works Features	[4]	[5]	[6]	[7]	Our Proposed
Confidentiality	No	No	No	Yes	Yes
Authentication				Yes	Yes
Integrity	No	No	No	Yes	Yes
PCIDSS standard	No	No	No	No	Yes

Ensures Application Security	**No**	**No**	**No**	**No**	**Yes**
Ensures Communication Security	**No**	**No**	**No**	**Yes**	**Yes**
Withstands Heartbleed Vulnerability	**No**	**No**	**No**	**Yes**	**Yes**
Withstands Replay Attacks	**No**	**No**	**No**	**No**	**Yes**
Withstands Man-In-The-Middle Attacks	**No**	**No**	**No**	**No**	**Yes**
Withstands Impersonation Attacks	**No**	**No**	**No**	**No**	**Yes**
Withstands reverse engineering attacks	**No**	**No**	**No**	**No**	**Yes**

Table 2:Comparision with related works

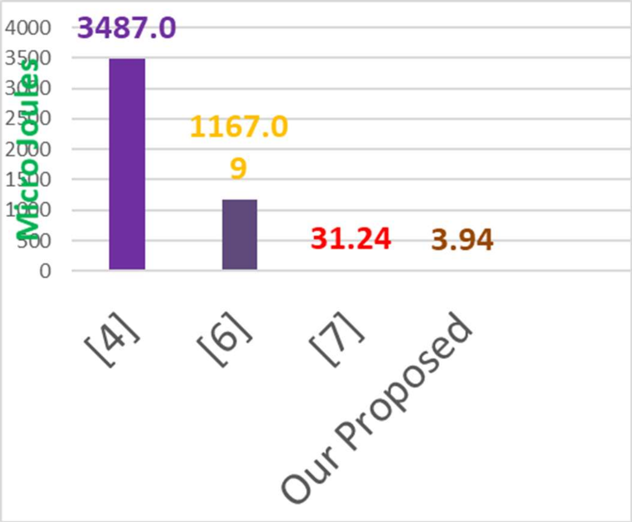
PERFORMACE ANALYSIS

Protocol	**Overall computation cost in seconds**
[4]	8TS+6TSig+8TH

	(1.05169)
[6]	7TS+2TSig+2TH (0.91493)
[7]	12TS+22TH (1.5724)
Our Proposed	2 TS+2TH (0.2614)

We compared the performance analysis of our proposed RFTTS framework with the related work in terms of “overall energy cost in Micro Joules” and “overall computational cost in Seconds”. According to [11], the time complexities calculated in seconds are TH = 0.0004 seconds and TS = 0.1303 seconds. As per [12] One ECPM (Elliptic Curve Point Multiplication) is 0.001015 seconds. RFTTS framework has better performance compared with the related works. As per [5] the energy required to generate AES encryption/decryption (ES) is 1.21 Micro Joules/byte and for generating hash code (EH) using SHA-1 algorithm is 0.76 Micro Joules. As per [12] the energy required for One ECPM (Elliptic Curve Point Multiplication) is equal to 578.55 Micro Joules from [12].

Table 3: Overall energy cost of RFTTS protocol



Bar Chart for overall energy cost of RFTTS protocol

Protocol	Overall Energy cost in Micro Joules
[4]	8ES+6ESig+8EH 8(1.21) +6(578.55) +8(0.76) =3487.06
[6]	7ES+2ESig+2EH 7(1.21) +2(578.55) +2(0.76) =1167.09

[7]	12ES+22EH 12(1.21) +22(0.76) =31.24
Our Proposed	2 ES+2EH 2(1.21) +2(0.76) =3.94

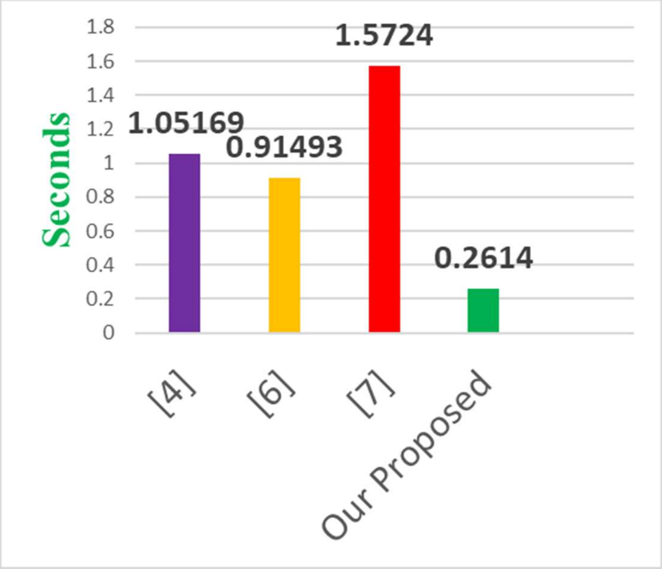


Table 4: Overall Computational Cost of RFTTS protocol

Protocol	Computation cost of the C in seconds	Computation cost of M in seconds	Computation cost B in seconds	Overall computation cost in seconds
Jen-Ho Yang an et al. [1]	3TS+2TH (0.3917)	2TS+1TH (0.261)	2TS (0.2606)	0.9133
Pourghomi, P et al. [2]	7TS (0.9121)	3TS (0.3903)	8TS (1.0424)	2.3448
Isaac, J et al. [3]	4 TS +4 TH (0.5228)	5 TS+2TH (0.6523)	3 TS (0.3909)	1.5660
Jen-Ho Yang a et al. [4]	5ECPM+2TS (0.265675)	5ECPM+2TS (0.265675)	2ECPM+4TS (0.52323)	1.05458
Jen-Ho Yang et al. [5]	4ECPM+2 TS (0.26466)	3ECPM+1TS (0.133345)	1ECPM+2TS (0.261615)	0.65962
Our Proposed	1 TS (0.1303)	1TS (0.1303)	2TS (0.2606)	**0.5212 Seconds**

Protocol	Energy cost for Client (C) Micro Joules	Energy cost for merchant (M) Micro Joules	Energy cost for Bank (B) Micro Joules	Overall Energy cost in Micro Joules
Jen-Ho Yang a et al. [1]	3TS+2TH (5.15)	2TS+1TH (3.18)	2TS (2.42)	10.75

Pourghomi, P et al. [2]	7TS (8.47)	3TS (3.63)	8TS (9.68)	21.78
Isaac, J et al. [3]	4 TS +4 TH (7.88)	5 TS+2TH (7.57)	3 TS (3.63)	19.08
Jen-Ho Yang a et al. [4]	5ECPM+2 TS (2895.17)	5ECPM+2 TS (2895.17)	2ECPM+4 TS (1161.94)	6952.28
Jen-Ho Yang et al. [5]	4ECPM+2 TS (2316.62)	3ECPM+1TS (1736.86)	1ECPM+2TS (580.97)	4634.45
Our Proposed	1SyE (1*1.21=1.21 µJ)	1SyE (1*1.21=1.21 µJ)	2SyE (2*1.21=2.42 µJ)	**4.84 µJ**

DISCUSSION

Payments industry is the main target of intruders especially PoS based payments. Intruders exploit four attack surfaces, they are User Credentials, Application Integrity, Device Integrity (Smart phone, PoS and Bank Server) and communication security. Following are the recommendations for PoS based payments solutions

PoS based payment solutions should ensure end to end security.

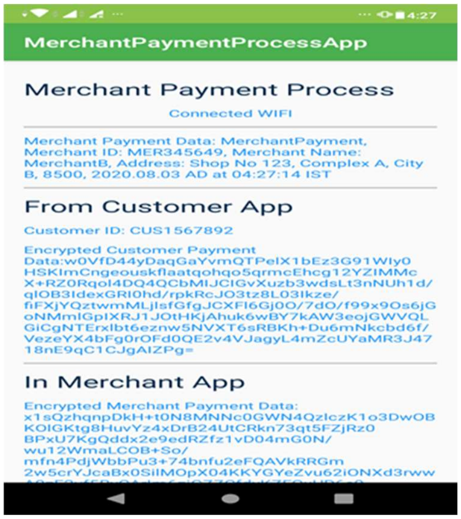
PoS based payment solutions should be in compliance to PCIDSS standard.

PoS based payment solutions should overcome reverse engineering attacks

PoS based payment solutions should overcome heart-bleed vulnerabilities.

PoS based payment solutions should overcome DoS and DDoS attacks.

PoS based payment solutions should adopt SE, TPM and TEE in order to withstand most of the attacks.



Merchant Payment Process

Connected WIFI

Merchant Payment Data: MerchantPayment, Merchant ID: MER345649, Merchant Name: MerchantB, Address: Shop No 123, Complex A, City B, 8500, 2020.08.03 AD at 04:27:14 IST

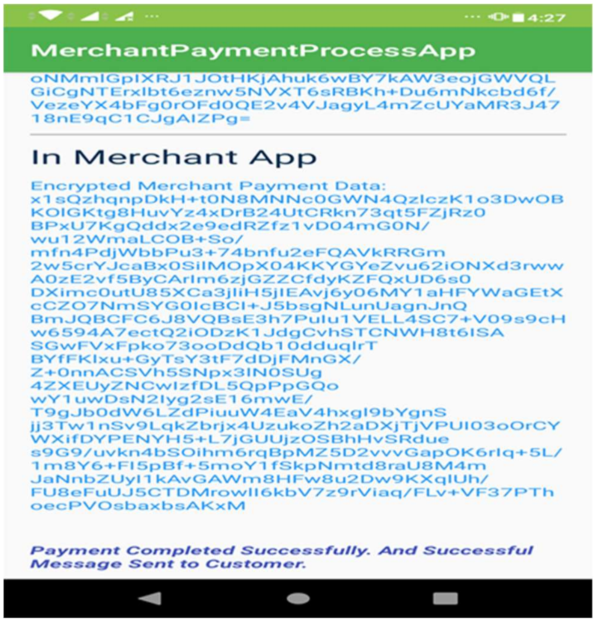
From Customer App

Customer ID: CUS1567892

Encrypted Customer Payment Data:w0VfD44yDaqGaYvmQTPelX1bEz3G91Wly0HSKImCngeouskflaatqohqo5qrmcEhcg12YZIMMcX+RZ0Rqol4DQ4QCbMIJCIGvXuzb3wdsLt3nNUh1d/qIOB3IdexGRI0hd/rpkRcJO3tz8L03Ikze/fiFXjYQztwmMLjlsfGfgJCXFI6Gj0O/7dO/f99x9Os6jGoNMmlGpIXRJ1JOtHKjAhuk6wBY7kAW3eojGWVQLGiCgNTErxlbt6eznw5NVXT6sRBKh+Du6mNkcbd6f/VezeYX4bFg0rOFd0QE2v4VJagyL4mZcUYaMR3J4718nE9qC1CJgAIZPg=

In Merchant App

Encrypted Merchant Payment Data:
x1sQzhqnpDkH+t0N8MNNc0GWN4QzlczK1o3DwOBKOIGKtg8HuvYz4xDrB24UtCRkn73qt5FZjRz0BPxU7KgQddx2e9edRZfz1vD04mG0N/wu12WmaLCOB+So/mfn4PdjWbbPu3+74bnfu2eFQAVkRRGm2w5crYJcaBx0SiIMOpX04KKYGYeZvu62iONXd3rww

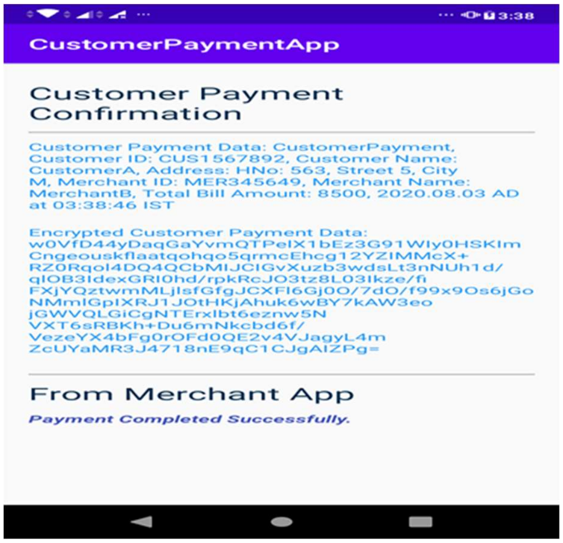


```
oNMmlGpIXRJ1JOtHKjAhuk6wBY7kAW3eojGWVQL
GiCgNTErxlbt6eznw5NVXT6sRBKh+Du6mNkcbd6f/
VezeYX4bFg0rOFd0QE2v4VJagyL4mZcUYaMR3J47
18nE9qC1CJgAIZPg=
```

In Merchant App

```
Encrypted Merchant Payment Data:
x1sQzhqnpDkH+t0N8MNNc0GWN4QzlczK1o3DwOB
KOlGKtg8HuvYz4xDrB24UtCRkn73qt5FZjRz0
BPxU7KgQddx2e9edRZfz1vD04mG0N/
wu12WmaLCOB+So/
mfn4PdjWbbPu3+74bnfu2eFQAVkRRGm
2w5crYJcaBx0SilMOpX04KKYGYeZvu62iONXd3rww
A0zE2vf5ByCArlm6zjGZZCfdyKZFQxUD6s0
DXimc0utU85XCa3jliH5jIEAvj6y06MY1aHFYWaGEtX
cCZO7NmSYG0IcBCl+J5bsgNLunUagnJnQ
BmJQBCFC6J8VQBsE3h7Pulu1VELL4SC7+V09s9cH
w6594A7ectQ2iODzK1JdgCvhSTCNWH8t6ISA
SGwFVxFpko73ooDdQb10dduqlrT
BYfFKlxu+GyTsY3tF7dDjFMnGX/
Z+0nnACSVh5SNpx3IN0SUg
4ZXEUyZNCwIzfDL5QpPpGQo
wY1uwDsN2lyg2sE16mwE/
T9gJb0dW6LZdPiuuW4EaV4hxgl9bYgnS
jj3Tw1nSv9LqkZbrjx4UzukoZh2aDXjTjVPUI03oOrCY
WXifDYPENYH5+L7jGUUjzOSBhHvSRdue
s9G9/uvkn4bSOihm6rqBpMZ5D2vvvGapOK6rIq+5L/
1m8Y6+FI5pBf+5moY1fSkpNmtd8raU8M4m
JaNnbZUyI1kAvGAWm8HFw8u2Dw9KXqlUh/
FU8eFuUJ5CTDMrowlI6kbV7z9rViaq/FLv+VF37PTh
oecPVOsbaxbsAKxM
```

Payment Completed Successfully. And Successful Message Sent to Customer.



Customer Payment Confirmation

```
Customer Payment Data: CustomerPayment,
Customer ID: CUS1567892, Customer Name:
CustomerA, Address: HNo: 563, Street 5, City
M, Merchant ID: MER345649, Merchant Name:
MerchantB, Total Bill Amount: 8500, 2020.08.03 AD
at 03:38:46 IST

Encrypted Customer Payment Data:
w0VfD44yDaqGaYvmQTPelX1bEz3G91WIy0HSKIm
Cngeouskflaatqohqo5qrmcEhcg12YZIMMcX+
RZ0Rqol4DQ4QCbMIJCIGvXuzb3wdsLt3nNUh1d/
qIOB3IdexGRI0hd/rpkRcJO3tz8L03Ikze/fi
FXjYQztwmMLjIsfGfgJCXFl6Gj0O/7dO/f99x9Os6jGo
NMmlGpIXRJ1JOtHKjAhuk6wBY7kAW3eo
jGWVQLGiCgNTErxlbt6eznw5N
VXT6sRBKh+Du6mNkcbd6f/
VezeYX4bFg0rOFd0QE2v4VJagyL4m
ZcUYaMR3J4718nE9qC1CJgAIZPg=
```

From Merchant App

Payment Completed Successfully.

Conclusion

Payments are the primary target of intruders, as they mainly exploit the vulnerabilities in mobile payment applications and PoS based applications. Existing PoS (Point of Sale) based payment frameworks are vulnerable, vulnerable to reverse engineering attacks, do not perform point-to-point encryption and do not ensure communication security. This research work proposes a Smart and Secure PoS (RFTTS) Framework which overcomes these attacks. Our proposed RFTTS framework ensures point-to-point encryption, Application hardening and Application wrapping. RFTTS framework overcomes repackaging attacks. RFTTS framework has very less communication and computation cost. RFTTS framework also addresses Heartbleed vulnerability. RFTTS protocol is successfully verified using Burrows–Abadi–Needham (BAN) logic, so it ensures all the security properties. RFTTS is threat modeled and implemented successfully.

References

https://www.globenewswire.com/news-release/2022/09/21/2519914/0/en/The-global-POS-Security-market-size-is-expected-to-grow-from-an-estimated-value-of-USD-4-0-billion-in-2022-to-USD-6-1-billion-by-2027-at-a-Compound-Annual-Growth-Rate-CAGR-of-8-6.html

https://www.businesswire.com/news/home/20230123005388/en/Cybersecurity-Market---Global-Forecast-to-2027-Opportunities-Emerging-in-Increasing-Use-of-AI-ML-And-Blockchain-Technologies-for-Cyber-Defense---ResearchAndMarkets.com

https://thecyberexpress-com.cdn.ampproject.org/c/s/thecyberexpress.com/cyber-attack-on-uae-banking-sector-adcb-nbf/amp/

Ali G, Dida MA, Elikana Sam A. A Secure and Efficient Multi-Factor Authentication Algorithm for Mobile Money Applications. Future Internet. 2021; 13(12):299. https://doi.org/10.3390/fi13120299

Vincent, O.R.; Okediran, T.M.; Abayomi-Alli, A.A.; Adeniran, O.J. An Identity-Based Elliptic Curve Cryptography for Mobile Payment Security. SN Comput. Sci. 2020, 1, 1–12

Thammarat C. Efficient and Secure NFC Authentication for Mobile Payment Ensuring Fair Exchange Protocol. Symmetry. 2020; 12(10):1649. https://doi.org/10.3390/sym12101649

Lu H-J, Liu D (2021) An improved NFC device authentication protocol. PLoS ONE 16(8): e0256367.

https://doi.org/10.1371/journal.pone.0256367

Brij B. Gupta and Shaifali Narayan, “A Key-Based Mutual Authentication Framework for Mobile Contactless Payment System Using Authentication Server”. Journal of Organizational and End User Computing, Volume 33(2), March-April 2021

Forough Sadat Mirkarimzade Tafti, Shahriar Mohammadi, Mehdi Babagoli, “A new NFC mobile payment protocol using improved GSM based authentication,” Journal of Information Security and Applications, vol. 62, pp. 1–10, Nov. 2021

M. Alshammari and S. Nashwan, "Fully authentication services scheme for nfc mobile payment systems," Intelligent Automation & Soft Computing, vol. 32, no.1, pp. 401–428, 2022.

Yeh KH. A secure transaction scheme with certificateless cryptographic primitives for IoT-based mobile payments. IEEE Syst J. 2018;12:2027–38.

Al-Haj, A.; Al-Tameemi, M.A. Providing security for NFC-based payment systems using a management authentication server. In Proceedings of the 2018 4th International Conference on Information Management (ICIM), Oxford, UK, 25–27 May 2018; pp. 184–187.

Sethia, D.; Gupta, D.; Saran, H. NFC secure element-based mutual authentication and attestation for IoT access. IEEE Trans. Consum. Electron. 2018, 64, 470–479.

Qiao Z, Yang Q, Zhou Y, Zhang M. Improved secure transaction scheme with certificateless cryptographic primitives for IoT-based mobile payments. IEEE Syst J. 2022;16:1842–50.

Mega, B. Framework for Improved Security on Usage of Mobile Money Application Based on Iris Biometric Authentication Method in Tanzania. Master’s Thesis, The University of Dodoma, Dodoma, Tanzania, 2020.

Osman, F.; Nakanishi, H. High Correctness Mobile Money Authentication System. Int. J. Psychosoc. Rehabil. 2020, 24, 3544–3556. [CrossRef]

Coneland, R.; Crespi, N.Wallet-on-wheels—Using a vehicle's identity for secure mobile money. In Proceedings of the 2013 17th International Conference on Intelligence in Next Generation Networks (ICIN), Venice, Italy, 15–16 October 2013; pp. 102–109.

Hassan, M.A.; Shukur, Z. A secure multi factor user authentication framework for electronic payment system. In Proceedings of the 2021 3rd International Cyber Resilience Conference (CRC), Langkawi Island, Malaysia, 29–31 January 2021; pp. 1–6.

Vincent, O.R.; Okediran, T.M.; Abayomi-Alli, A.A.; Adeniran, O.J. An Identity-Based Elliptic Curve Cryptography for Mobile Payment Security. SN Comput. Sci. 2020, 1, 1–12. [CrossRef]

Wang F, Yang N, Shakeel PM, Saravanan V. Machine learning for mobile network payment security evaluation
system. Trans Emerging Tel Tech. 2021. https:// doi. org/ 10. 1002/ ett. 4226.