ISSN 1975-8359(Print) / ISSN 2287-4364(Online)
The Transactions of the Korean Institute of Electrical Engineers Vol. 66, No. 12, pp. 1921~1936, 2017
http://doi.org/10.5370/KIEE.2017.66.12.1921

LTE 네트워크에서의 IoT 장치를 위한 향상된 보안 거래

Enhanced Secure Transaction Protocol for IoT Devices via LTE Network

Ebrahim AL-Alkeem* · Chan Yeob Yeun† · Yousef Al Hammadi** · Hyun Ku Yeun · and Young-Ji Byon*

Abstract – Internet of Thing (IoT) and NFC (Near Field Communication) have got a good adaptable structure that it can be easily combined with any wireless network. Since IoT/NFC can be used to communicate wirelessly with all the transactions that can be done remotely without any physical connections. In this paper, we propose an enhanced secure IoT/NFC protocol based on LTE network that enhances the original security level provided by the LTE. Our approach is new in a sense that it covers LTE in contrast to old networks like GSM and 3G, which substantially treated in the literature. Moreover, both GSM and 3G have several drawbacks when they are combined with the NFC technology, which has potential weakness in confidentiality, integrity, and authentication. Hence our new approach will resolve the security of the new LTE system. We expect that our protocol will result in new secure applications for the smart phone markets.

Key Words : NFC, EAP-AKA, ProVerif, KDF, MME, HSS

1. Introduction

The security of NFC (Near field Communication) / IoT (Internet of Things) [1, 2] is becoming very important as the number of applications that supports are increasing. Nowadays many people are using NFC technology in their daily activities as it can be used for financial transactions, downloading applications, contactless transactions and etc. However, there are a number of security challenges that NFC protocols need to address such as confidentiality, integrity, availability, authentication and authorization. The security of mobile transactions is not an easy task since the characteri stics of this system are closely investigated. For example, there should be a number of security issues of the transactions, starting from the price checking and to the actual execution of the transaction. The security requirements of NFC protocols mainly depend on the general network security framework which has been much discussed in the previous literature [3, 4].

In this paper, we propose a new NFC protocol that satisfies the desired security requirements to handle confidentiality, integrity, availability and authentication. Security analysis with formal verification using a tool based on pi-calculus is also provided for the proposed protocol. The aim of the proposed protocol is to ensure that the majority of the security issues, which are illustrated in Fig. 1, will be tackled. Also, this paper is an enhancement of [1] by adding a verification security proof.

The main aim of this project is to investigate the security and privacy of the NFC protocol using the LTE network. In order to maintain the security requirements and to improve the level of privacy of a transaction through the wireless network. It also addresses the security threats that affecting the both GSM and 3G network and will highlight the advantages of introducing LTE to the NFC transaction protocol.

Furthermore, an evaluation of the results and comparison with other networks is provided in order to prove the efficiency of the new protocol. Note that NFC with LTE obtains its robustness from the two layers provided by the LTE network, which will be discussed in Section 3, Note, however, that GSM and 3G use one layer only [5, 6].

The remaining part of this paper is structured as follows. Section II highlights the related works and main motivations. Also, Section II discusses the previous work. Section III combining NFC with GSM and 3G. Section IV shows the proposed NFC protocol. Section V security analysis of the protocol. Finally, Section VI concludes the paper.

† Corresponding Author : Dept. of ECE, Khalifa University of Science and Technology, Abu Dhabi, UAE.
E-mail: chan.yeun@kustar.ac.ae
* Dept. of ECE, Khalifa University of Science and Technology, Abu Dhabi, UAE.
** Dept. of ICT, UAE University, Al Ain, UAE
***Dept. of Engineering, HCT, Abu Dhabi, UAE

Received : November 10, 2017; Accepted : November 26, 2017

2. Related works

NFC is a niche technology to exchange data between a smart device and a tag or between two smart devices, one of which will act like a reader (phone) and the other will act like a target (tag or another phone). Usually, the target does not require any power since it can be a passive component as the reader will do the scanning process. NFC basically is a subsidiary of RFID (Radio Frequency Identification). However, IoT/NFC has shorter reading range about 20 cm and is used in many applications nowadays, such as performing contactless payment through the mobile phone. Also, it can be used in advertising and downloading applications and etc. [7, 8]. The following Figure 1 shows NFC related attacks.

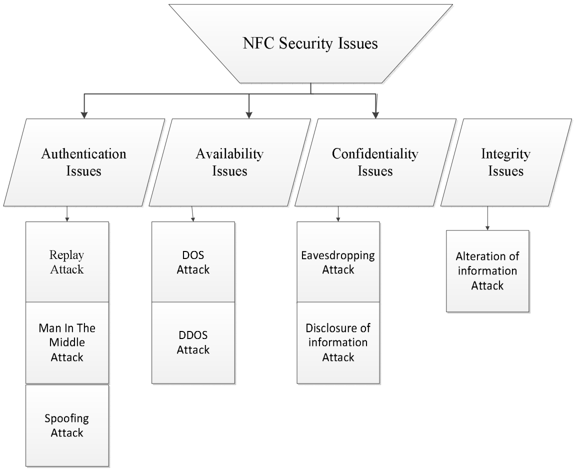


Fig. 1 NFC Security Attacks

Basically, the NFC enabled phone reads an NFC tag and communicates with the backend server. In some cases the NFC phone could be both the tag and the reader or there would be no need to contact the backend server. NFC tag or POS (Point of Sale) can be found in many items such as smart posters, the POS, electronic devices, etc. It usually comes with small chip hidden behind a sticker with NFC logo. The payment processing server can be communicated through the NFC by different communication technologies. Smart transactions provided by the backend server and might be vary according to applications such as a web page for reserving movie ticket, issuing receipts, or highly secure financial transaction service which requires high secure connection. The NFC can be used in many applications such application download, ticket purchasing, tourist guide and money transactions. In this paper, we will go through the payment scenario employing the NFC technology. Figure 2 shows a complete smart retail environment based on item level tagging and NFC applications. The following model can increase the efficiency of the retails operation and will smooth the purchasing process Figure 2.

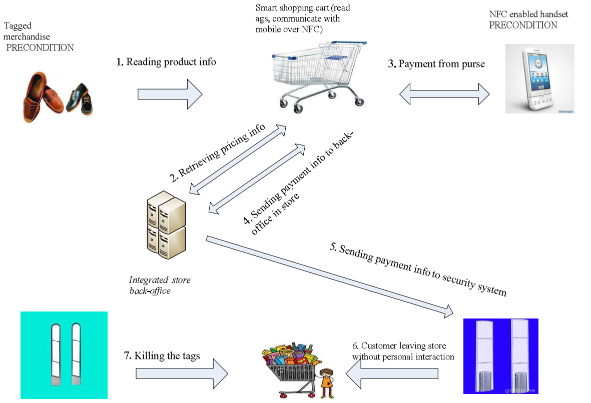

Tagged merchandise PRECONDITION
1. Reading product info
Smart shopping cart (read ags, communicate with mobile over NFC)
3. Payment from purse
NFC enabled handset PRECONDITION
2. Retrieving pricing info
4. Sending payment info to back-office in store
Integrated store back-office
5. Sending payment info to security system
7. Killing the tags
6. Customer leaving store without personal interaction

Fig. 2 NFC-based payment process in retail environment.

This efficient model could minimize both operational time and cost as it reduces the human interaction in the operation as a user will be more self-dependent and can conduct most of the operations by himself. There is no need for an employee to tell the client about the price and to go through payment process. This is a very smart approach which minimizes the time requires and efficient.

The basic NFC system mainly consists of three basic components as the following Figure 3 shows Mainly the NFC phone reads an NFC tag and communicates with the backend server. In some cases an NFC tag can be another NFC phone or there would be no need to contact the backend server.

NFC tag or POS can be found in many items such as smart posters, POS, electronic devices, etc. it's usually comes like small chip hidden behind a sticker with NFC logo.

It contains small data based on their applications such as uniform resource identifiers (URIs), contact information, authentication credentials, valuable information, etc.

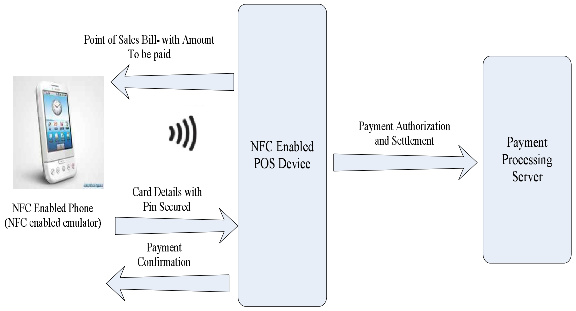

Point of Sales Bill- with Amount To be paid
NFC Enabled Phone (NFC enabled emulator)
Card Details with Pin Secured
Payment Confirmation
NFC Enabled POS Device
Payment Authorization and Settlement
Payment Processing Server

Fig. 3 NFC-based payment process in retail environment.

NFC chips are embedded within mobile handsets enabling them to read NFC tags. Mobile phone industry has shown several NFC mobile phones manufactured in last few years. It is also can possible to include NFC chip within subscriber identity module (SIM) card or even in micro SD cards [22].

Payment processing Server can be communicated through the NFC by different communication technology. Service provided by the backend server might vary according to applications. It can be a web page for reserving movie ticket, issuing receipts, or highly secure financial transaction service which requires high secure connection.

NFC technology is dealing with money transactions which includes user information and credit card details therefore the privacy is critical. Possible security attacks include eavesdropping, interception attacks, data modification, and physical thefts. There are many challenges that faces the NFC protocol which we addressed most of them in this paper.

3. Combining NFC technology with GSM and 3G

In this section, we discuss about some previous works done in the NFC protocol, which includes combining NFC technology with GSM [8], and combining NFC technology with 3G [9]. Furthermore, it provides a comparison between the three wireless networks.

3.1 NFC transaction based on GSM network

As mobile phone becomes very important in our life as it simplifies the daily activity and saves much time for us. The way of using phone has been changed from the past as people use mobile phones for making calls and sending SMS, however, nowadays mobile phones are used in many applications such as m-payment, e-resource and etc. In this the scheme which discus the combination of NFC transaction based on GSM Network [10]. The study shows how the NFC mobile user can operate on the GSM network. It can be noticed that the scheme can be divided in to four major parts such as initial state, Price checking, Triple authentication, and transaction execution. Each part has different responsibility [11].

The whole system is basically based on the GSM network and uses its secret triplets [12, 13]. The security is acceptable in GSM network, however, the major disadvantage is that the length of the Signed Response (SRES) and 64 bit ciphering key (Kc) is too short and the customer has to trust the Mobile Originated (MO) with their long term secrets. Also the merchants must also establish a relationship with the MO which in most cases will increase the risk. The GSM scheme presents the standard cipher key length which up to 64 bits. Therefore it can be extended to 3G system which uses a longer cipher key length (128 bits) in order to increase the level of security.

Reusing the existing GSM network and performing the NFC technology will allow the scheme to enjoy the full GSM security capability as we are adding it over the existing GSM network as it can be used anywhere GSM network is available. Also there are many disadvantages which lead us to move to combining the NFC with the 3G as the scheme will gain both the advantage and disadvantage of the system [14, 15]. A main weakness is the short length of the SRES and K length in the GSM also customer should blindly trust the MO with their long term secrets. We can say that the NFC technology over GSM gain the same disadvantage of the GSM network therefore we suggest moving to the next generation to minimize the drawbacks.

3.2 NFC transaction based on 3G network

In this part NFC technology has been added to the 3G network in order to reduce the drawbacks which were affecting the previous scheme.

3G authentication already has mature authentication mechanism which can lead to an improvement in security aspects from the previous GSM scheme. Both GSM and 3G systems use challenge-response authentication and encryption/decryption for user identification and data confidentiality. The transaction based on 3G network can be divided into three major parts: Price checking, Mutual authentication, and transaction execution [16]. There are couple of requirements that should be available to perform a success M-payments through NFC network. Both client and the POS have to register with the Mobile Network Operator (MNO) and NFC enabled. The client should use the same authentication algorithm which been applied in the 3G network.

The benefits of using 3G network for NFC payment is that the NFC system will use the same security which could be offered by the 3G network. There are some of drawbacks that can be considered in the 3G security which can be solved with the Long Term Evolution (LTE) system that is used in the proposed scheme. While utilizing the Temporary Mobile Subscriber Identity (TMSI) for initial user identification the system could be restricted if been used in a moving vehicle which can be considered as a weak point.

Also the mutual authentication process which can change the phone frequency and that will lead to change on cipher and integrity keys while the client enter the network for the first time. Also supporting only the online transaction as user will not be able to perform the transaction offline in order to minimize the cost. And there are more drawbacks that affect this scheme which similar to 3G drawbacks therefore we recommend moving to 4G as it solve most of them.

3.3 Comparison between NFC protocols based on the three wireless networks

The following Table 1 explains the deference between the three networks in terms of security requirements and the advantages of adding the NFC protocol to these networks.

From the table above, we can summarize that GSM & 3G network have limitation in terms of the security requirements that could affect the NFC applications which mainly based on the mobile transactions. People required a trusted methodology to perform their transaction and protect themselves from any intruder. Therefore, we suggest the following scheme based on LTE network in order to increase the level of the security provided by the NFC protocol. As system will gain the same security level that LTE has.

Table 1 Comparison between wireless networks in term of security

Security requirements	GSM	3G	LTE
Authentication	Single Authentication	Mutual Authentication	Mutual Authentication
Availability	Support GSM network	Only in 3G network coverage	Only in LTE network coverage
Authorization	Based on SRES	Based on IMSI	Two level of authorization (IMSI & SNID)
Confidentiality	Short length of SRES & Kc. Client has to trust the MO	Longer Cipher key length (128 bits)	Two level of security (NAS & AS) which support ciphering & integrity
Security Mechanism	Use same GSM security Mechanism	Use 3G security Mechanism	Use LTE security
Security level	Weak	Stronger	Strongest

4. The Propesed NFC Protocol

The proposed design shows an example of a user start selecting an item and checking the price through the NFC POS. The client will make a decision based on the receiving price and proceed with the payments through the LTE network. We understood that the GSM and 3G have many limitations in terms of security, which can be addressed by using LTE network. Also we noticed that the NFC technology is depending heavily on the network itself as it gains the same security level which network has. Therefore, combining NFC technology with the LTE network will increase the overall level of security. Also, our novel approach has been introduced for the first time. The previous works only covered networks such as GSM [17] and 3G [18].

The proposed scheme provides new approach which has not been discussed in the previous literature, as our solution depends on the LTE infrastructure which provides two security levels that can provide isolation to the system when attacks occur.

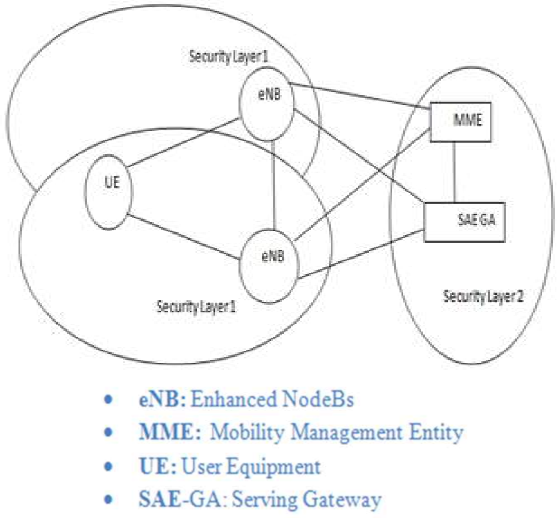


Fig. 4 Security layer of LTE

As Figure 4 illustrates, there are two layers of the security in LTE. The reason behind it is that the eNB is not capable to be a fully trusted zone as attacker can gain access and we could add additional layer in order to increase the level of security. This is the reason why the LTE has two security layers: Security of AS (Access Layer) and NAS (Non-Access Layer). Security layer one is usually between the UE and the eNB which provides encryption and integrity protection for the AS. However, the NAS security is between the MME (Mobile Management Entity) and the UE (User Equipment) which provide encryption and integrity protection for NAS signaling. The NFC payment process through the LTE network can be divided into five major parts: Price Checking, Authentication, NAS Security, AS Security, and transaction execution. The proposed scheme explains the internal process of a customer transaction using the NFC mobile to perform a purchase in a store through the LTE network.

The proposed scheme explains the internal process of a customer transaction using the NFC mobile to perform a purchase in a store through the LTE network. A detailed explanation for the five major parts can be shown in the following Figure 5-8.

4.1 Stage 1 price checking

There are many requirements that need to be fixed in order to perform a success transaction. Our proposed design is to achieve the transaction in a real shop environment. Both the POS (Point of Sale) and the client phone should be NFC enabled. The Mobile phone should support the LTE technology and afford high calculation capability. The store area should be well covered by the LTE network and both should operate under the same LTE cell.

The Mobile phone used should support the LTE technology and afford high calculation capability. The store area should be well covered by the LTE network and both should operate under same LTE cell.

The first four steps include the initial item scanning, displaying price and getting the confirmations between both POS and the customer mobile. The following Figure 5 explains the price checking stage.

Step 1-2: The POS starts with scanning RFID tags from each selected product. The shop will keep the details of the process for future use which include the order number (ON), the total price (TP) and date/time (DT). The DT is used for reference in case of losing information. Also is more convenient to use the time stamp or nonce in the beginning of the transaction which will be required later for synchronization and verification. In this design, we used nonce to avoid reply attack.

Step 3: The price will be checked by the customer and will decide based on his requirement, the attacker can interfere in part as he can send a fake price to the client which can be less or more than the original price in order to gain benefits or make loss for both part the shop or the client as if the customer received less price and he accept it the acknowledgment sent to the POS will be less than the actual price and it will get rejected, however, if the customer received higher price than the one which been sent by the POS and get accepted in this case the client is losing money. By keep changing the price the system will over loaded and will lead to crash the transaction.

Step 4: by assuming a secure key were applied on the order information and after agreeing on the price, the customer will place the NFC phone on the POS and will enter the PIN to approve the payments. In this case the POS will send full details about the product including the price. The client will approve the payment by entering the PIN number to proceed with the next cycle which is the authentication process.

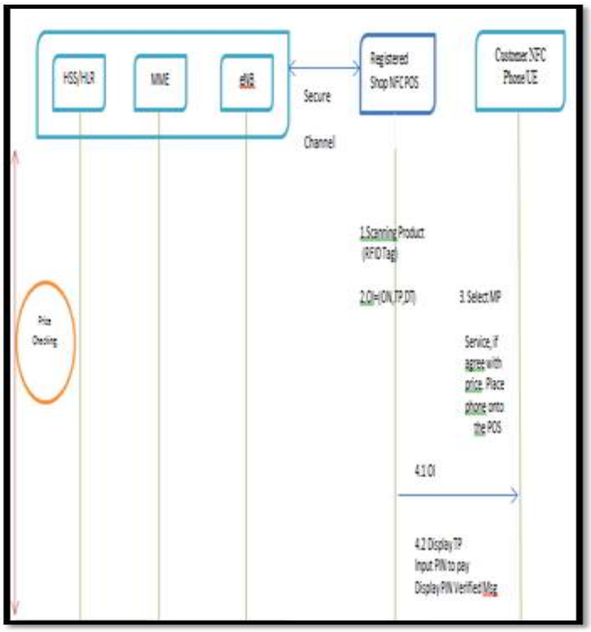


Fig. 5 Price checking stage.

4.2 Stage 2 Identification & Authentication

We assume that the communication between the payment gate way and the shop POS is over a secure channel. The customer phone should be on and authenticated with the LTE network. Both client phone and the server should share the secret key (KSIASME). The LTE uses the AKA algorithm, which is a challenge and response protocol that offers a mutual authentication. This means that both the client and the network exchange the challenge and response which allows proving the identity of the client. Therefore, the possibility of MITM (man in the middle attack) attack is not possible with the mutual authentications. The mutual authentication stage is where the mobile reader will agree on the TP(Total Price) with the POS as it will validate the user to guarantee secure transaction. Figure 6 explains this stage.

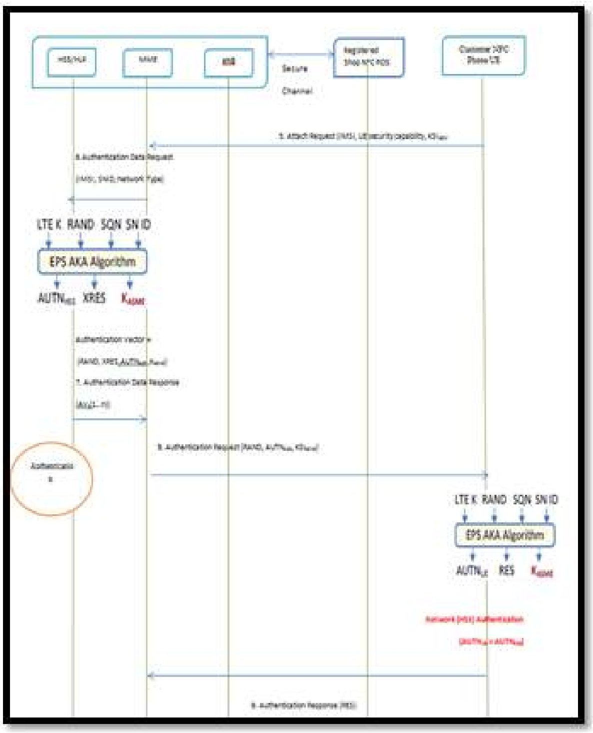


Fig. 6 Identication & Authentication.

The authentication stage is where the mobile reader will agree on the TP with the POS as it will validate the user to guarantee secure transaction through the following steps.

Step 5: The client will send the authentication request to the MME which will act as an interface including the international mobile subscriber identity & the key (IMSI, KSI¬¬ ASME)

Step 6: After that the MME will request for the authentication vectors related to that specific IMSI by sending authentication data request to the HSS. Once the MME knows the user IMSI it can request the EPS AKA algorithm to generate the authentication vector from the HSS /HLR. The following figure explains the EPS AKA algorithm process.

EPS AKA algorithm can be divided in to two main functions the crypto functions and the KDF. The main function of AKA is to utilize the secret key cryptography. Therefore both network and the client passes the pre shared key KSI ASME copy to verify client and for the client to verify the network which result in mutual authentications. The following equations explain the whole process.

$$MAC = f_{1k}(SQN \parallel RAND \parallel AMF) \quad (3.1)$$

$$XRES = f_{2k}(RAND) \quad (3.2)$$

$$CK = f_{3k}(RADN) \quad (3.3)$$

$$IK = f_{4k}(RAND) \quad (3.4)$$

$$AK = f_{5k}(RAND) \quad (3.5)$$

$$K_{ASME} = KDF\ (SQN \Phi AK, SN\ id\ ,\ CK,\ IK) \quad (3.6)$$

$$AUTN = SQN\ \Phi\ AK \parallel AMF \parallel MAC \quad (3.7)$$

$$AV = RAND \parallel XRES \parallel KASME \parallel AUTN \quad (3.8)$$

Step 7: The CK, IK and other parameters such as the serving network identity (SN ID) will not be sent to the MME as the new key generated K ASME which replaces their specifications. The Serving network identity (SNID) is consist of the Mobile Country Code (MCC) and Mobile Network Code (MNC) of the serving network. SNID provide a better key separation between different networks to make sure that any key generated by one network can't be used by other network.

The HSS will share the pre shared key against the IMSI and will calculate the authentication vector (RAND, XRES, AUTNHSS, KASME) which will be sent as response to the MME.

Step 8: The MME will collect the keys, RAND and XRES from the AV and will sends (RAND, AUTNHSS, KSIASME) with the authentication request to the UE. The MME will keep the KASME and the XRES generated by the EPS AKA algorithm. The UE will start authentication by using the EPS AKA algorithm and checking the AUTN received. The AUTN is a parameter been generated in the HSS by its Key. The Mobile terminal will start calculating its own AUTN by its own key and SQN and compare between the received one.

The following equations explain step 8 processes:

$$XAK = f_{5k}(RAND) \quad (3.9)$$

$$SQN = XAK\ \Phi\ SQN\ \Phi\ AK \quad (3.10)$$

$$XMAC = f_{1k}(SQN \parallel RAND \parallel AMF\ ) = ?MAC \quad (3.11)$$

$$XSQN = ?SQ \quad (3.12)$$

Step 9: After calculating the keys and the rest. It will send the RES along with authentication response to the MME after receiving the RES the MME will compare it with the XRES if it matches then the authentication is successful else will send authentication failure to the UE. The following equations explain step 9 process.

$$RES = f2_{k}(RAND) \quad (3.13)$$

$$CK=f3_{k}(RAND) \quad (3.14)$$

$$IK=f4_{k}(RAND) \quad (3.15)$$

$$K_{ASME}= KDF\ (SQN \oplus AK,SN\ id,CK,IK) \quad (3.16)$$

4.3 Stage 3 NAS Security

This part will cover the process of Non-Access Stratum (NAS) Security Mode Command (SMC) procedure which allows the system to perform its security through the ciphering & protecting the integrity. The Key Derivation Algorithm (KDF) is mainly used to generate long term master secret key from a short shared secret key also to arrange for a secure channel session. There are two keys generated in the NAS security stage by using the KDF algorithm which are KNASenc & KNASint. Both UE amd the MME get the keys according the KASME which make sure that the level of confidentiality between UE and MME is high in [7, 8]. The NAS SMC will trigger the client terminal to generate the NAS ciphering key (KNASenc) and NAS integrity key (KNASint). Figure 7 explains this stage [25, 27].

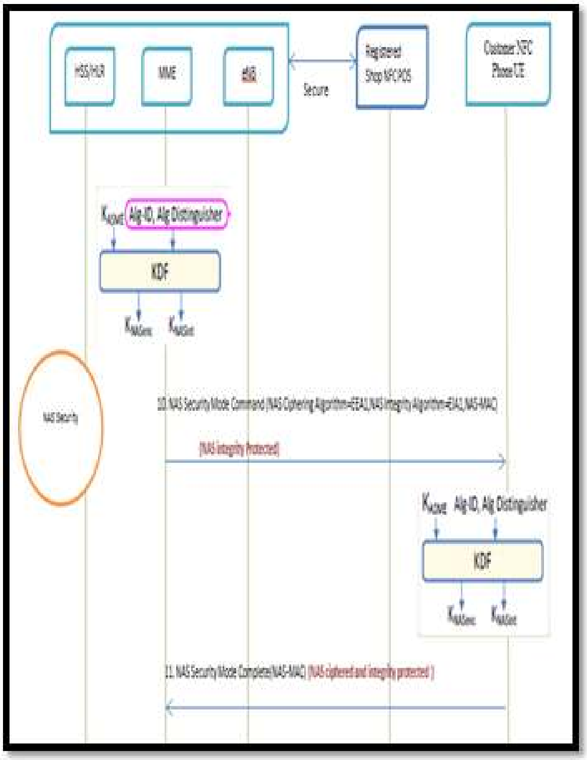

HSS/HLR
MME
eNB
Secure
Registered Shop NFC POS
Customer NFC Phone UE
K_ASME
Alg-ID, Alg Distinguisher
KDF
K_NASenc
K_NASint
NAS Security
10. NAS Security Mode Command (NAS Ciphering Algorithm=EEA1,NAS Integrity Algorithm=EIA1,NAS-MAC)
(NAS integrity Protected)
K_ASME
Alg-ID, Alg Distinguisher
KDF
K_NASenc
K_NASint
11. NAS Security Mode Complete(NAS-MAC) (NAS ciphered and integrity protected)

Fig. 7 NAS Security.

Step 10: The MMS will reset the DL count and will calculate the Knas-int, Knas-int as described on the figure above. The MMS will send the NAS security mode command (NAS integrity algorithm = EIA1, NAS-MAC), (NAS Ciphering algorithm = EEA1, NAS).

Step 11: The UE will calculate the KNASenc and KNASint by using the KDF algorithm and will send the completion message after a success process. MME will receive the NAS security mode complete (NAS- MAC) {NAS ciphered and integrity protected}.

4.4 Stage 4 AS Security

In this stage system, one will generate KeNB which will be used in the eNB(enhanced nodeB) to produce the following keys, respectively.

- K UPenc: This key is used to provide confidentiality protection between UE and eNB, Both UE and eNB get the keys according to KeNBS and identity of encryption algorithms.
- K RRCint: This key is essential for protecting the integrity between the UE and eNB
- K RRCenc: The confidentiality of RCC will be protected by this key which deduced according to the K eNB and the encryption algorithm identifier

Step 12: After receiving the security command from the UE, The MME will compute the KeNB through the KDF algorithm based on the existing KASME. The KeNB will be sent to the eNB to proceed with the next step. The initial context setups request (UE security capability, KeNB) will be sent to the eNB .

Step 13: The eNB will start calculating the keys, KRRCenc, KRRCint, KUpenc by using the KDF algorithm which was explained earlier [7]. Then the eNB will send RRC security mode command with (AS integrity algorithm, AS ciphering algorithm, and MAC-1) to the UE to perform the AS process.

Step 14: The UE in this step will practice the same As process to generate the keys KRRCenc, KRRCint, KUpenc and will acknowledge back the RRC security mode complete to eNB. By this step the system will be ciphered and integrity protected as the following in Figure 8.

Step 15: The eNB will declare the completion of the authentication process and will send a success acknowledgment to the UE requesting for the price information to proceed with the transaction process.

The following Table 2 explains the security association for both NAS & AS.

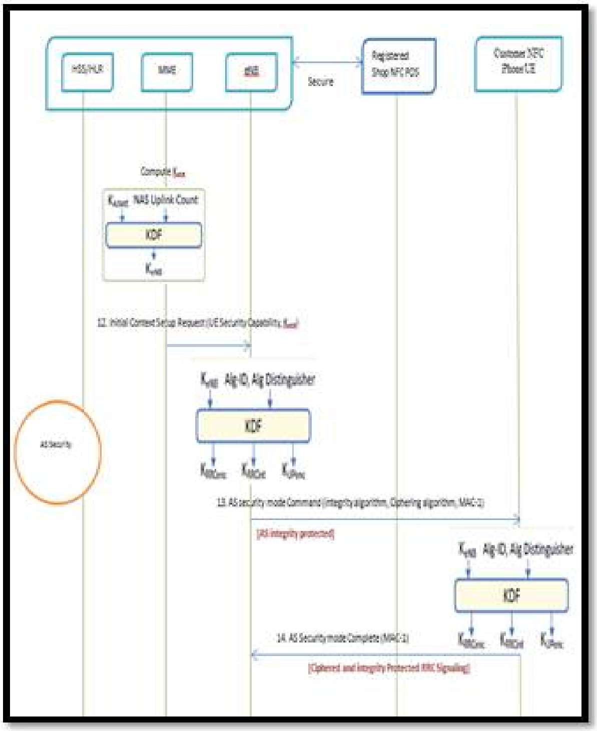

HSS/HLR
MME
eNB
Secure
Registered Shop NFC POS
Customer NFC Phone UE
Compute KeNB
KASME NAS Uplink Count
KDF
KeNB
12. Initial Context Setup Request (UE Security Capability, KeNB)
KeNB Alg-ID, Alg Distinguisher
KDF
KRRCenc KRRCint KUPenc
AS Security
13. AS security mode Command (integrity algorithm, Ciphering algorithm, MAC-1)
[AS integrity protected]
KeNB Alg-ID, Alg Distinguisher
KDF
KRRCenc KRRCint KUPenc
14. AS Security mode Complete (MAC-1)
[Ciphered and integrity Protected RRC Signaling]

Fig. 8 AS Security.

Table 2 Comparison between NAS & AS

Security Association	Non-Access Stratum (NAS)	Access Stratum (AS)
Termination points	UE and MME	UE and eNB
Ciphering	NAS Signalling	RRC Signalling (Signaling radio bearer) User plane (data radio bearer)
Integrity and replay protection	NAS Signalling	RRC Signalling (Signaling radio bearer)
Security protocol layers	NAS	PDCP
Security Command procedures	NAS	RCC

4.5 Stage 5 Transaction execution

In this stage, the client will confirm the price by selecting accept button and will confirm the payment process. The PI will be generated by the customer and the process information is prepared to be sent to the POS which includes (Receipt No, Total price, Total counts). The backend system will generate time stamp for the transaction to minimize the risk of replay attack. After that, the transaction process will be activated through the HSS which reflect the results to both the POS and the Client in [9, 10].

Step 16: The client will confirm the price by selecting accept button and will confirm the payment process

Step 17: The PI will be generated by the customer and the process information will prepared to be sent to the POS which includes (Receipt No, Total price, TSU, TC) the backend system will generate time stamp for the transaction to minimize the risk of replay attack.

Step 18: The UE encrypt the PI, S, and IMSI by the Kc which is result for the user approval on the payment details. The PI gets encrypted to avoid any future change by done by the shop. The S is usually present the authentication process which been discussed in the previous stage and make sure that it is for the same client. The IMSI is used to identify the user and allow the servicer to maintain the billing center can be used to perform the pursues and deduction. The UE will encrypt the message by using two keys as Kc1 is shared between the UE & the POS and Kc is shared between the UE and PG to avoid any modification on the price at the POS. The following Figure

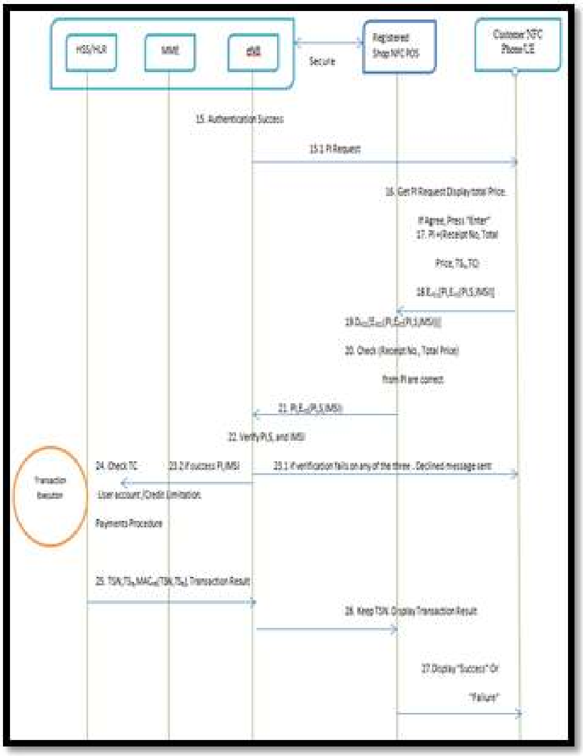

HSS/HLR
MME
eNB
Secure
Registered Shop NFC POS
Customer NFC Phone UE
15. Authentication Success
15.1 PI Request
16. Get PI Request Display total Price.
If Agree, Press "Enter"
17. PI =(Receipt No, Total
Price, TSu, TC)
18 EKc1(PI, EKc(PI, S, IMSI))
19 DKc1(EKc1(PI, EKc(PI, S, IMSI)))
20. Check (Receipt No, Total Price)
from PI are correct.
21. PI, EKc(PI, S, IMSI)
22. Verify PI, S, and IMSI
Transaction Execution
24. Check TC
23.2 if success PI, IMSI
23.1 if verification fails on any of the three , Declined message sent
User account /Credit Limitation.
Payments Procedure
25. TSN, TSu, MACKs(TSN, TSu), Transaction Result
26. Keep TSN. Display Transaction Result
27. Display "Success" Or
"Failure"

Fig. 9 Transaction execution.

9 explains the discussed process.

Step 19-20: The POS decrepit the received message by the Kc1 and compare the value with the price been sent earlier in the price checking stage if both are matched it will proceed with the next stage.

Step 21-22: The shop will send (PI, Ekc (PI, S, IMSI)) to the payment gateway to verify the transaction, the PG will decrepit the message received by the Kc and check the S is its belong to the same user. The IMSI in this stage works as a second confirmation about the user. Also it will compare between both prices the one which is sent by the POS and the encrypted one. To avoid any change in the price at the POS.

Step 23-24: If the verification fails on any one of these three (PI,S,IMSI) parts the PG will send a declined message to the client and will terminate the transaction. Otherwise it will proceed with the payment procedures with the HSS.

Step 25: Once the user pass the credit check then the transaction result will be sent to the eNB.

Step 26-27: The POS will keep a copy of the TSN to be used as a proof in the future. The user will receive a success or failure message on the transaction status.

In this stage system, one will generate

5. Security Analysis of the Proposed Protocol

The simulation of NFC protocol conducted in this project based on the Proverif software version 1.86. Proverif is a fully automated tool that works to prove that the protocol is robustly safe by applying several attacking methods. It analyzes the security of cryptographic protocols. Using Proverif tool the outcome can be one of the following

- ProVerif succeeds to prove robust safety.
- ProVerif finds an attack as a counterexample to root bust safety
- ProVerif can neither prove nor disprove robust safety.

It is a tool that can support the cryptographic primitives, including encryption and decryption (symmetric and asymmetric), digital signatures, and hash function. In addition, other primitives such as rewriting rules and equations can be modeled using terms. This verification provides the validity of the NFC protocol. In addition this chapter will cover the security analysis of the NFC protocol based on the result obtained from the ProVerif simulations. The result enabled us to know the efficiency and the security strength of the NFC protocol.

There are several attacks that could affect the NFC protocol. The main motivation to use “ProVerif (Protocol Verification)” [19, 20, 21] is to analyze and estimate the security capability of a system. ProVerif can test the system from security point of view by focusing on major security requirements and trying all the possible ways to attack it. It is used for verifying security properties for cryptographic protocols using a specification language that is based on an extension of pure Pi-Calculus. Consequently, ProVerif can prove security properties, correspondence and observational equivalence. Note that this proofing capability helps analyze the secrecy and authentication of security protocols that are permitted from the provider.

The ProVerif tests all the probable attempts of the intruder to break the mutual authentication definition in the NFC protocol. The following Table 3 shows the mutual authentication result.

Although the ProVerif is a powerful tool that can verify all the possible attacks regarding the mutual authentication, some drawbacks are not covered yet in this tool.

The mutual authentication is used to authenticate a communication between two parties in which each one is authenticating the other by using nonce, in order to prevent the system from masquerading, spoofing and the man-in-the-middle attacks. By testing the ProVerif on our

Table 3 Result of Mutual Authentication test on NFC protocol

The query	Proverif output
event(EndCMME()) ⇒ event(BeginCMME()).	True
event(EndMMEHSS()) ⇒ event(BeginMMEHSS()).	True
event(EndHSSMME()) ⇒ event(BeginHSSMME()).	True
event(EndHSSMME()) ⇒ (event(EndMMEHSS()) && event(BeginMMEHSS())).	True
event(EndMMEC()) ⇒ event(BeginMMEC()).	True
event(EndMMEC()) ⇒ (event(EndCMME()) && event(BeginCMME())).	True
event(EndCPOS()) ⇒ event(BeginCPOS()).	True
event(EndPOSeNB()) ⇒ event(BeginPOSeNB()).	True
event(EndeNBHSS()) ⇒ event(BegineNBHSS()).	True
event(EndHSSeNB()) ⇒ event(BeginHSSeNB()).	True
event(EndHSSeNB()) ⇒ (event(EndeNBHSS()) &&event (BegineNBHSS())).	True
event(EndeNBPOS()) ⇒ event(BegineNBPOS()).	True
event(EndeNBPOS()) ⇒ (event(EndPOSeNB()) && event(BeginPOSeNB())).	True
event(EndPOSC()) ⇒ event(BeginPOSC()).	True
event(EndPOSC()) ⇒ (event(EndCPOS())&& event(BeginCPOS())).	True

protocol, we will be able to find the security threats that will affect messages that follow. For example, the attacker can send a response to the processor before receiving the request itself.

We could use the value only once to create a mutual authentication between two processors and protect the system. The ProVerif tests all the probable attempts of the intruder to break the mutual authentication definition in the NFC protocol. Also, the ProVerif will try all assumptions to identify any gap in the confidently of the NFC protocol and will work to introduce an intruder to the protocol and try all the possibility to attack the system messages through the wireless channel. Table 4 shows the results of confidentiality test on NFC protocol [23, 24].

The ProVerif tests the key exchange methodology in the NFC protocol and determines the possible attacks on the system via the public channel. The main challenge is to make sure that the key can be exchanged safely across the system. In our proposal, the integrity is based on the message transaction between two processors. We encrypt the transaction to make sure that the message has not been modified by any outside attacker. That means only the legitimate sender and receiver can read the message and the integrity of the original message is assured. Therefore, providing the security of the session keys and mutual authentication of the protocol can detect any modification attacks [26].

Also, the system provides the mutual authorization process by identifying the client through the IMSI (International Mobile Subscriber Identity) of the phone, which makes sure that only authorized user is eligible to proceed with the transaction. Also the SNID (Serving Network Identity) generated by the MME will work to identify the specific client to proceed with the authentication process through the EPS AKA algorithm. The IMSI will be used by the HSS to check the credit limit for particular client in order to proceed with the deduction process.

Table 4 Results of confidentiality test on NFC protocol

Parameter	Initial security status	Proverif output
K_{s1}, K_{s2}, K_{s3}, K_{s4},K_{sI}, K_{c}, K_{c1},	Secure	Secure
IMSI	Unknown	Secure
SNID	Unknown	Secure
N_{c}	Unknown	Secure
N_{c12}	Unknown	Secure
N_{c13}	Unknown	Secure
N_{c14}	Unknown	Secure
N_{c15}	Unknown	Secure
N_{p11}	Unknown	Secure
N_{n1}	Unknown	Secure
Skey1	Unknown	No output
Skey2	Unknown	No output
Pri	Unknown	Secure
TRAN1	Unknown	Secure

6. Conclusion

This paper proposed an enhanced secure IoT/NFC protocol to improve security and privacy of NFC transactions based on the LTE network. The proposed protocol secures the interactions between all the five parties, which are Client, POS, eNB, MME, and HSS (home subscriber server). Our approach is new since it is based on LTE network, which is different from other old networks like GSM and 3G studied extensively in the previous literature. The security and privacy of NFC transaction based on the LTE network in general and securing the interactions between all the five party (Client, POS, eNB, MME, and HSS). The aim of this project involves two main parts. Firstly, to investigate the security and privacy of NFC transaction protocol. Secondly, to design and implement a secure approach based on the LTE network. Also to allow the user to benefit from the wide range of security that been offered by the LTE network. It also focused on the related and recent works done in this field.

This proposal discusses many security attacks that can affect the security and privacy of the mobile transaction. Also it discusses the vulnerability of each wireless network that's been used to perform the contactless transaction through the NFC technology.

We provided design specification for implementing our secure protocol over LTE network. Finally, we presented security analysis of our protocol based on the formal method, ProVerif.

References

[1] Al Alkeem, E., Yeun, C.Y., Baek, J.S.: Secure NFC Authentication Protocol Based on LTE Network. Lecture Notes in Electrical Engineering, Springer, vol. 280, pp. 363-371. (2014)

[2] Lo, N.W, Yeh, K.H., Yeun, C.Y.: New Mutual Agreement Protocol to Secure Mobile RFID-Enabled Devices.

Information Security Technical Report, Elsevier, vol. 13, 3, pp. 151-157, (2008).

[3] Chen, W.D., Hancke, G.P., Mayes, K.E., Lien, Y., Chiu, J.H.: Using 3G Network Components to Enable NFC Mobile Transactions and Authentication. Progress in Informatics and Computing (2010)

[4] Massoth, M., Bingel, T.: "Performance of Different Mobile Payment Service Concepts Compared with a NFC-Based Solution," iciw, 2009 Fourth International Conference on Internet and Web Applications and Services, pp. 205-210. (2009)

[5] ISO/IEC 18092 (ECMA-340): Information technology - Telecommunications and information exchange between systems - Near Field Communication - Interface and Protocol (NFCIP-1).

[6] ISO/IEC 14443: Identification cards－Contactless integrated circuit cards- Proximity cards.

[7] NFC Forum: White paper on 'smart posters. Tech. Rep. (2011)

[8] NFC Forum: White paper on essentials for successful NFC mobile ecosystem. Tech. Rep., (2008)

[9] NFC Forum: White paper on 'the keys to truly interoperable communications. Tech. Rep., (2007)

[10] Han, C.K., Choi, H.K, Baek, J.W.: Evaluation of Authentication Signaling Loads in 3GPP LTE/SAE Networks. IEEE 34th Conference on Local Computer Networks. pp. 37-44. (2009)

[11] 3GPP. TS 36.331 V9.1.0, 3rd Generation Partnership Project: Technical Specificaion Group Radio Access Netword; Evolved Universal Terrestrial Radio Access (E-UTRA), Radio Resource Control (RRC) Protocol Specification

[12] 3GPP. TS 33.102 V9.2.0, 3rd Generation Partnership Project: Technical Specification Group Services and System Aspects; 3G Security ; Security architecture protocol specification

[13] 3GPP. TS 24.301 V9.3.0, 3 rd Generation Partnership Project; Technical Specification Group Core Network and Terminals; Non Access Stratum (NAS) protocol for Evolved Packet System (EPS): Stage 3 protocol Specification.

[14] Communications Fraud Control Association (CFCA): www.cfca.org

[15] Dastani, M., van Riemsdijk, M.B., Dignum, F., Meyer, J.C.: A programming language for cognitive agents: goal directed 3APL. Programming multiagent systems, first international workshop (Pro-MAS' 03), vol.3067 of LNCS, Berlin, 2004, pp. 111－130. (2004)

[16] Bordini, R.H., H¨ubner, J.F.: BDI agent programming in AgentSpeak using jason (tutorial paper). In Computational Logic in Multi-Agent Systems VI (CLIMA), vol. 3900 of Lecture Notes in Artificial Intelligence, Heidelberg, Germany, pp. 143－－164. (2006)

[17] Rinard, M.C., Scales, D.J., Lam, M.S.: Jade: a high-level, machine-independent language for parallel programming. IEEE Computer Society, vol. 26, 6, pp. 28 －38. (1993)

[18] Zhang, Z., Zhang, X., Sandhu, R.; ROBAC: Scalable Role and Organization Based Access Control Models. International Conference on Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom), pp. 1-9. (2006).

[19] Abadi, M., Fournet, C.; Mobile Values, New Names, and Secure Communication. Proc. of the 28th ACM Symposium on Principles of Programming Languages (POPL' 01), pp.104--115. (2001)

[20] Blanchet, B, Chaudhuri, A.: Automated Formal Analysis of a Protocol forSecure File Sharing on Untrusted Storage. IEEE Symposium on Security and Privacy, pp. 417-431. (2008)

[21] Blanchet, B, Smyth, B.: ProVerif 1.85: Automatic Cryptographic Protocol Verifier, User Manual and Tutorial: http://www.proverif.ens.fr/manual.pdf, (2011)

[22] Yeun, C.Y., Farnham, T., "Secure m-commerce with WPKI," In proceedings of 1st International Workshop for Asian PKI, October 2001. (This is a reference for e-payment)

[23] Shehada, D., Yeun, C.Y., Zemerly, M.J., Al Qutayri, M., Al Hammadi, Y., Damiani, E., Hu, J., "BROSMAP: A Novel Broadcast Based Secure Mobile Agent Protocol for Distributed Service Applications," Security and Communication Networks, Vol. 2017, 3606424, 2017. (This is a reference for e-payment

[24] Lo, N.W., Yeh, K.H., Yeun, C.Y., "New mutual agreement protocol to secure mobile RFID-enabled devices," Information Security Technical Report, Vol. 13, No. 3, pp. 151-157, 2008. (This is a reference for RFID/IoT/NFC)

[25] Shemaili, M.A.B., Yeun, C.Y., Zemerly, M.J., Mubarak, K., "A novel hybrid cellular automata based cipher system for internet of things," In Future information technology, Springer, pp. 269-276, 2014. (This is a reference for RFID/IoT/NFC)

[26] Al Alkeem, E., Yeun, C.Y., Zemerly, M.J., "Security and privacy framework for ubiquitous healthcare IoT devices" In proceeding of 10th International Conference for Internet Technology and Secured Transactions (ICITST), pp. 70-75, December 2015. (This is a reference for RFID/IoT/NFC)

[27] Al Alkeem, E., Shehada, D., Yeun, C.Y., Zemerly, M.J., Hu, J., "New secure healthcare system using cloud of things," Cluster Computing, Vol. 2, No. 3, pp. 2211-2229, 2017. (This is a reference for RFID/IoT/NFC)

Appendix

ProVerif code

```
let processPOS=
new pr:bitstring;
event BeginPC(pri);
(*Message 1*)
out(c,pri);
(*Message 7*)
in(c, m5:bitstring)
let (p:price,m6:bitstring)=dec(m5,Kc1) in
let (=pri)=p in
event EndCPOS(p,Kc1,m6);
(*Message 8*)
new np11:nonce;
out(c, (p,np11,m6));
event BeginPOSeNB(p,np11,m6);
(*Message 11            *)
in(c, m10:bitstring);
let (=np11,nc15:nonce,m11:bitstring)=dec(m10,Ks4) in
event EndeNBPOS(np11,nc15,m11,Ks4);
(*Message 12            *)
out( c, enc((nc15,m11),Kc1));
event BeginPOSC(nc15,m11,Kc1);
let processC=
(*Message 1*)
in(c,=pri);
event EndPC(pri);
(*Message 2*)
new nc:nonce;
event BeginCMME(IMSI,KSI,nc);
out (c, enc((IMSI,KSI,nc),Ks1));
(*Message 5*)
in(c,m4:bitstring);
let (=nc,nx1:nonce,Skey2:key)=dec(m4,Ks1) in
let (=Skey2)=AKA(SNID,nx1) in
event EndMMEC(nc,nx1,Skey2);
(*Message 6*)
in(c,m13:bitstring);
let (r1:bitstring,nn1:nonce)=dec(m13,Kc) in
event EndeNBC(r1,nn1,Kc);
(*Message 7            *)
new nc11:nonce;
new pr1:price;
out (c, enc((pr1,enc((pr1,IMSI,nc11),Kc)),Kc1));
event BeginCPOS(nc11,Kc,Kc1);
(*Message   12 *)
in (c, m11:bitstring);
let (=nc11,m12:bitstring)=dec(m11,Kc1) in
event EndPOSC(nc11,m12,Kc1);
let processeNB=
(*Message 6*)
new nn:nonce;
out(c,enc((r,nn),Kc));
event BegineNBC(r,nn,Kc);
(*Message 8*)
in(c, (p1:price,np12:nonce,m7:bitstring));
let (=p1,=IMSI,nc12:nonce)=dec(m7,Kc) in
event EndPOSeNB(p1,np12,nc12);
(*Message 9*)
new nNB:nonce;
out(c, enc((p1,nNB,nc12,IMSI),Ks3));
event BegineNBHSS(nNB,nc12,Ks3);
(*Message 10            *)
in(c,m9:bitstring);
let (=nNB,nc14:nonce,TRAN1:bitstring)=dec(m9,Ks3) in
event EndHSSeNB(nc14,TRAN1);
(*Message   11            *)
out(c, enc((np12,nc14,TRAN1),Ks4));
event BegineNBPOS(nc14,TRAN1,Ks4);
let processMME=
(*Message 2*)
in (c, m2:bitstring);
let (=IMSI,=KSI,nc1:nonce)=dec(m2,Ks1) in
event EndCMME(IMSI,KSI,nc1);
(*Message 3*)
new nMME:nonce;
out(c, enc((IMSI,SNID,nMME),Ks2));
event BeginMMEHSS(IMSI,SNID,Ks2);
(*Message 4*)
in(c,m4:bitstring);
let (nx1:nonce,Skey1:key,=nMME)=dec(m4,Ks2) in
event EndHSSMME(nc1,nx1,Skey1);
(*Message 5*)
out(c,enc((nc1,nx1,Skey1),Ks1));
event BeginMMEC(nc1,nx1,Skey1);
let processHSS=
(*Message 3*)
in(c, m3:bitstring);
let (x1:client,x2:client,nMME1:nonce)=dec(m3,Ks2) in
let (=SNID)=x2 in
```

```
event EndMMEHSS(x1,x2,nMME1);
new n1:nonce;
(* Message 4*)
out(c, enc((n1,AKA(x2,n1),nMME1),Ks2));
event BeginHSSMME(n1,AKA(x2,n1));
(* Message 9*)
in(c,m8:bitstring);
let(p2:price,nNB1:nonce,nc13:nonce,=IMSI)=dec(m8,Ks3) in
event EndeNBHSS(nNB1,nc13,p2);
(* Message 10            *)
out(c, enc((nNB1,nc13,TRAN(p2,IMSI)),Ks3));
event BeginHSSeNB(nNB1,nc13,Ks3);
```

Message follow Demonstrations

POS events consist of five main messages as the following figures. Message 1 shows the process which start from POS and end in Client part

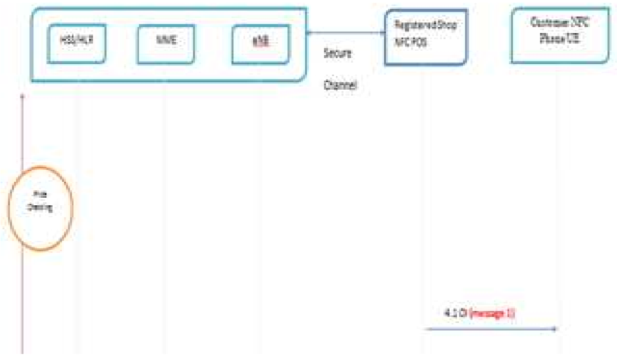
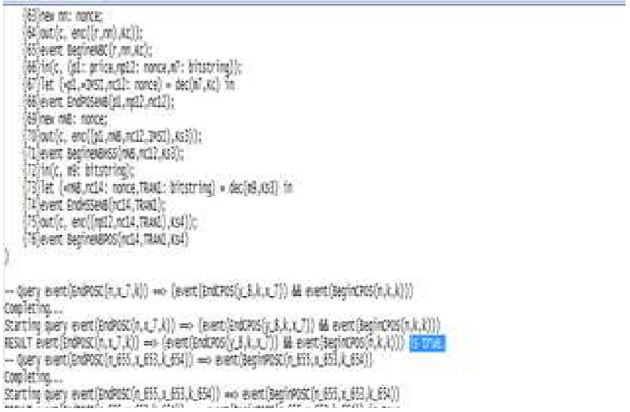


Fig. 10 Price checking stage, Message 1

The proverif simulation shows a true results for the security part of the protocol after testing it as the following screen shoot.



```
new nn: nonce;
out(c, enc((r,nn),Kc));
event BeginNBC(r,nn,Kc);
in(c, (p1: price,np12: nonce,m7: bitstring));
let (=p1,=IMSI,nc12: nonce) = dec(m7,Kc) in
event EndPOSeNB(p1,np12,nc12);
new nNB: nonce;
out(c, enc((p1,nNB,nc12,IMSI),Ks3));
event BegineNBHSS(nNB,nc12,Ks3);
in(c, m8: bitstring);
let (=nNB,nc14: nonce,TRAN1: bitstring) = dec(m8,Ks3) in
event EndHSSeNB(nc14,TRAN1);
out(c, enc((np12,nc14,TRAN1),Ks4));
event BegineNBPOS(nc14,TRAN1,Ks4)
)

-- Query event(EndPOSC(n,x_7,k)) ==> (event(EndCPOS(y_8,k,x_7)) && event(BeginCPOS(n,k,k)))
Completing...
Starting query event(EndPOSC(n,x_7,k)) ==> (event(EndCPOS(y_8,k,x_7)) && event(BeginCPOS(n,k,k)))
RESULT event(EndPOSC(n,x_7,k)) ==> (event(EndCPOS(y_8,k,x_7)) && event(BeginCPOS(n,k,k))) is true.
-- Query event(EndPOSC(n_655,x_653,k_654)) ==> event(BeginPOSC(n_655,x_653,k_654))
Completing...
Starting query event(EndPOSC(n_655,x_653,k_654)) ==> event(BeginPOSC(n_655,x_653,k_654))
```

Fig. 11 Proverif result of Message 1

The following two figures explain the mutual authentication between (Client & MME) which can be explained by messages 2 & 5.

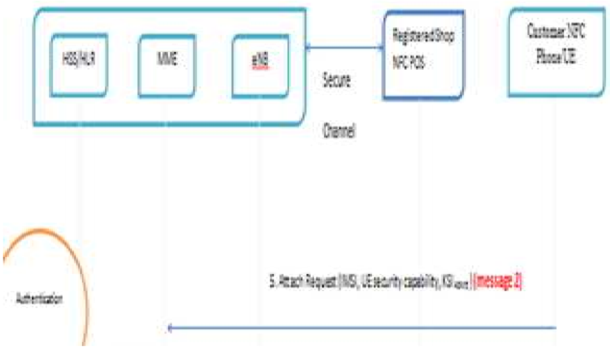


Fig. 12 Authentication stage, Message 2

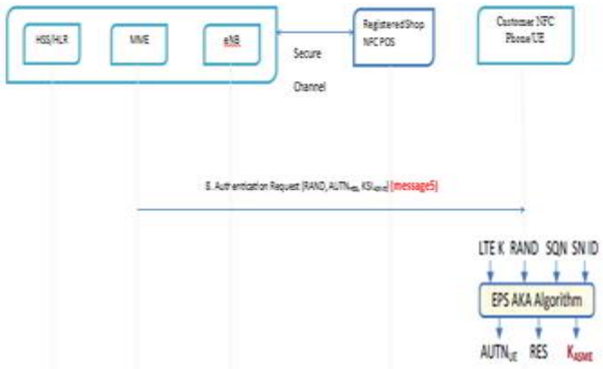
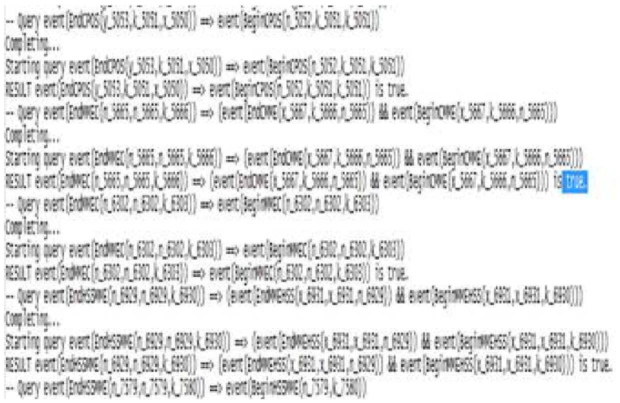


Fig. 13 Authentication stage, Message 5

The following screen shoot shows the result of the Proverif simulation which conclude that the system is mutually authenticated and the attacker fail to attack the system. As true means.



```
-- Query event(EndCPOS(y_5053,k_5051,x_5050)) ==> event(BeginCPOS(n_5052,k_5051,k_5051))
Completing...
Starting query event(EndCPOS(y_5053,k_5051,x_5050)) ==> event(BeginCPOS(n_5052,k_5051,k_5051))
RESULT event(EndCPOS(y_5053,k_5051,x_5050)) ==> event(BeginCPOS(n_5052,k_5051,k_5051)) is true.
-- Query event(EndMMEC(n_5665,n_5665,k_5666)) ==> (event(EndCMME(x_5667,k_5666,n_5665)) && event(BeginCMME(x_5667,k_5666,n_5665)))
Completing...
Starting query event(EndMMEC(n_5665,n_5665,k_5666)) ==> (event(EndCMME(x_5667,k_5666,n_5665)) && event(BeginCMME(x_5667,k_5666,n_5665)))
RESULT event(EndMMEC(n_5665,n_5665,k_5666)) ==> (event(EndCMME(x_5667,k_5666,n_5665)) && event(BeginCMME(x_5667,k_5666,n_5665))) is true.
-- Query event(EndMMEC(n_6302,n_6302,k_6303)) ==> event(BeginMMEC(n_6302,n_6302,k_6303))
Completing...
Starting query event(EndMMEC(n_6302,n_6302,k_6303)) ==> event(BeginMMEC(n_6302,n_6302,k_6303))
RESULT event(EndMMEC(n_6302,n_6302,k_6303)) ==> event(BeginMMEC(n_6302,n_6302,k_6303)) is true.
-- Query event(EndHSSMME(n_6929,n_6929,k_6930)) ==> (event(EndMMEHSS(x_6931,x_6931,n_6929)) && event(BeginMMEHSS(x_6931,x_6931,k_6930)))
Completing...
Starting query event(EndHSSMME(n_6929,n_6929,k_6930)) ==> (event(EndMMEHSS(x_6931,x_6931,n_6929)) && event(BeginMMEHSS(x_6931,x_6931,k_6930)))
RESULT event(EndHSSMME(n_6929,n_6929,k_6930)) ==> (event(EndMMEHSS(x_6931,x_6931,n_6929)) && event(BeginMMEHSS(x_6931,x_6931,k_6930))) is true.
-- Query event(EndHSSMME(n_7579,n_7579,k_7580)) ==> event(BeginHSSMME(n_7579,k_7580))
```

Fig. 14 Proverif output of Messages (2 & 5)

Message 3 & 4 as follow shows the data follow between (MME &HSS).

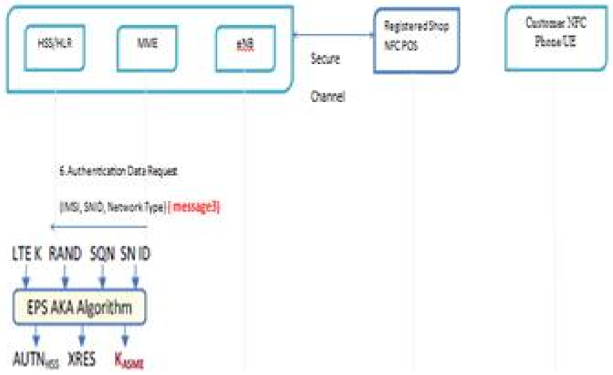


Fig. 15 Authentication stage, Message 3

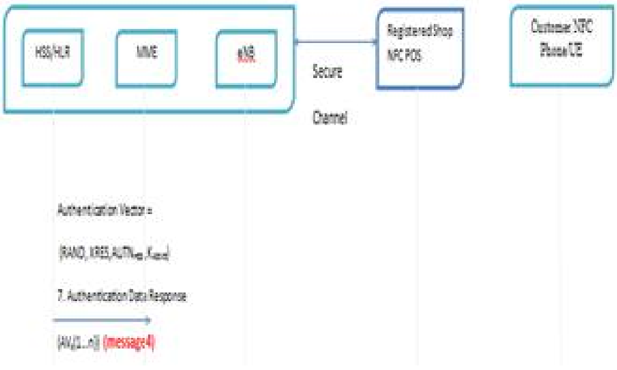
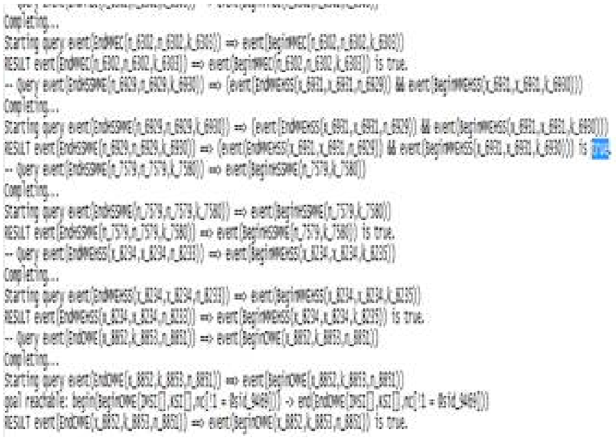


Fig. 16 Authentication stage, Message 4

The simulation results shows a true result to all possible attacks that can harm the security requirements of the protocol.



```
-- Query [illegible]
Completing...
Starting query event(EndMMEC(n_6302,n_6302,k_6303)) ==> event(BeginMMEC(n_6302,n_6302,k_6303))
RESULT event(EndMMEC(n_6302,n_6302,k_6303)) ==> event(BeginMMEC(n_6302,n_6302,k_6303)) is true.
-- Query event(EndHSSMME(n_6929,n_6929,k_6930)) ==> (event(EndMMEHSS(x_6931,x_6931,n_6929)) && event(BeginMMEHSS(x_6931,x_6931,k_6930)))
Completing...
Starting query event(EndHSSMME(n_6929,n_6929,k_6930)) ==> (event(EndMMEHSS(x_6931,x_6931,n_6929)) && event(BeginMMEHSS(x_6931,x_6931,k_6930)))
RESULT event(EndHSSMME(n_6929,n_6929,k_6930)) ==> (event(EndMMEHSS(x_6931,x_6931,n_6929)) && event(BeginMMEHSS(x_6931,x_6931,k_6930))) is true.
-- Query event(EndHSSMME(n_7579,n_7579,k_7580)) ==> event(BeginHSSMME(n_7579,k_7580))
Completing...
Starting query event(EndHSSMME(n_7579,n_7579,k_7580)) ==> event(BeginHSSMME(n_7579,k_7580))
RESULT event(EndHSSMME(n_7579,n_7579,k_7580)) ==> event(BeginHSSMME(n_7579,k_7580)) is true.
-- Query event(EndMMEHSS(x_8234,x_8234,n_8233)) ==> event(BeginMMEHSS(x_8234,x_8234,k_8235))
Completing...
Starting query event(EndMMEHSS(x_8234,x_8234,n_8233)) ==> event(BeginMMEHSS(x_8234,x_8234,k_8235))
RESULT event(EndMMEHSS(x_8234,x_8234,n_8233)) ==> event(BeginMMEHSS(x_8234,x_8234,k_8235)) is true.
-- Query event(EndCMME(x_8852,k_8853,n_8851)) ==> event(BeginCMME(x_8852,k_8853,n_8851))
Completing...
Starting query event(EndCMME(x_8852,k_8853,n_8851)) ==> event(BeginCMME(x_8852,k_8853,n_8851))
goal reachable: begin(BeginCMME(IMSI[],KSI[],nc[!1 = @sid_9469])) -> end(EndCMME(IMSI[],KSI[],nc[!1 = @sid_9469]))
RESULT event(EndCMME(x_8852,k_8853,n_8851)) ==> event(BeginCMME(x_8852,k_8853,n_8851)) is true.
```

Fig. 17 Proverif results of both Messages (3 & 4)

Message 6 shows the price information request after the authentication part is over as the eNB will request the client to send the price.

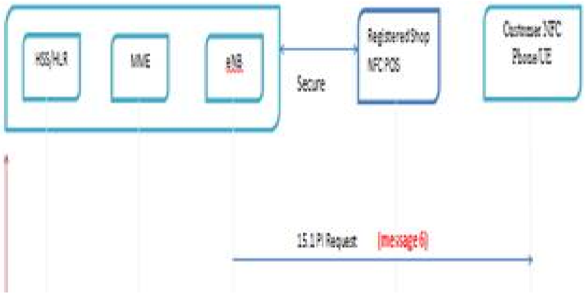


Fig. 18 Transaction execution stage, Message 6

In message 7 the client will send the price encrypted with two keys kc & kc1 in order to protect the price from any external change.

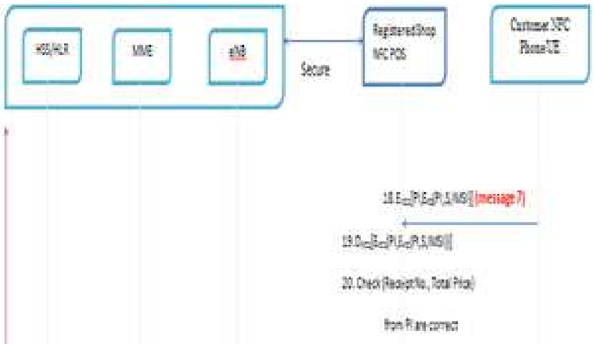

HSS/HLR
MME
eNB
Secure
Registered Shop
NFC POS
Customer NFC
Phone UE
18.E_kc1(PI,E_kc(PI,S,IMSI)) (message 7)
19.D_kc1(E_kc1(PI,E_kc(PI,S,IMSI)))
20.Check (ReceiptNo, Total Price)
from PI are correct

Fig. 19 Transaction execution stage, Message 7

Message 8 & 11 as the following two figures shows. Both POS & eNB are fully secured as per the proverif output.

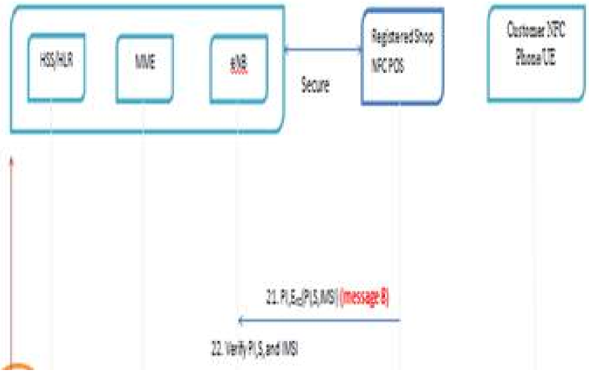


Fig. 20 Transaction execution stage, Message 8

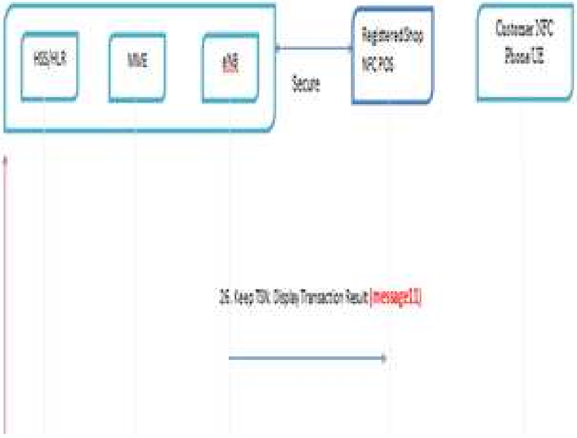
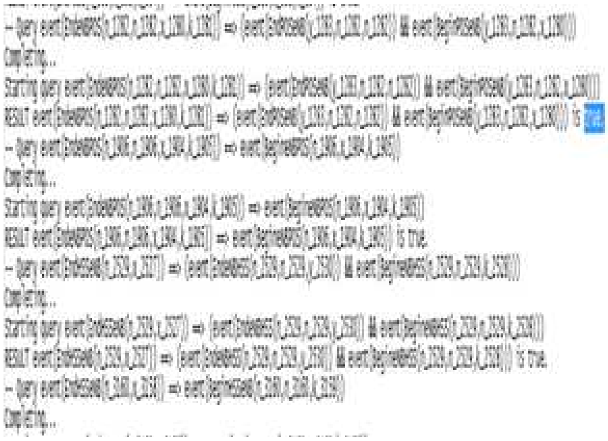


Fig. 21 Transaction execution stage, Message 11

The proverif results shows true for both messages (8 &11) which mean that both messages are secured.



```
-- Query event(endeNBPOS(r_1382,n_1382,x_1381,k_1382)) ==> (event(endPOSeNB(r_1383,n_1382,n_1381)) && event(beginPOSeNB(r_1383,n_1382,x_1380)))
Completing...
Starting query event(endeNBPOS(r_1382,n_1382,x_1381,k_1382)) ==> (event(endPOSeNB(r_1383,n_1382,n_1381)) && event(beginPOSeNB(r_1383,n_1382,x_1380)))
RESULT event(endeNBPOS(r_1382,n_1382,x_1381,k_1382)) ==> (event(endPOSeNB(r_1383,n_1382,n_1381)) && event(beginPOSeNB(r_1383,n_1382,x_1380))) is true.
-- Query event(endeNBPOS(r_1906,n_1906,x_1904,k_1905)) ==> event(beginNBPOS(r_1906,x_1904,k_1905))
Completing...
Starting query event(endeNBPOS(r_1906,n_1906,x_1904,k_1905)) ==> event(beginNBPOS(r_1906,x_1904,k_1905))
RESULT event(endeNBPOS(r_1906,n_1906,x_1904,k_1905)) ==> event(beginNBPOS(r_1906,x_1904,k_1905)) is true.
-- Query event(endNFCSeNB(r_2528,n_2527)) ==> (event(endeNBNFCS(r_2528,n_2528,y_2530)) && event(beginNBNFCS(r_2528,n_2528,k_2530)))
Completing...
Starting query event(endNFCSeNB(r_2528,n_2527)) ==> (event(endeNBNFCS(r_2528,n_2528,y_2530)) && event(beginNBNFCS(r_2528,n_2528,k_2530)))
RESULT event(endNFCSeNB(r_2528,n_2527)) ==> (event(endeNBNFCS(r_2528,n_2528,y_2530)) && event(beginNBNFCS(r_2528,n_2528,k_2530))) is true.
-- Query event(endNFCSeNB(r_3160,n_3158)) ==> event(beginNFCSeNB(r_3160,n_3160,k_3159))
Completing...
```

Fig. 22 Proverif results of both messages (8 & 11)

Same applied for message 9 & 10 as both are secured and mutual authentic.

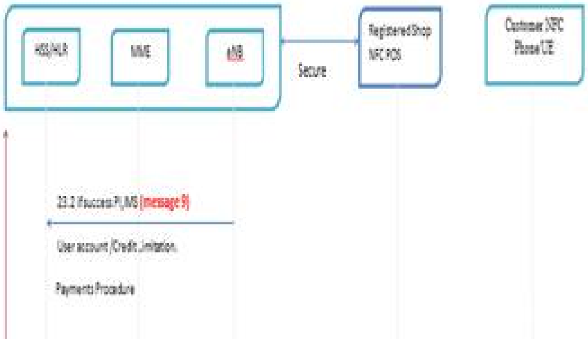


Fig. 23 Transaction execution stage, Message 9

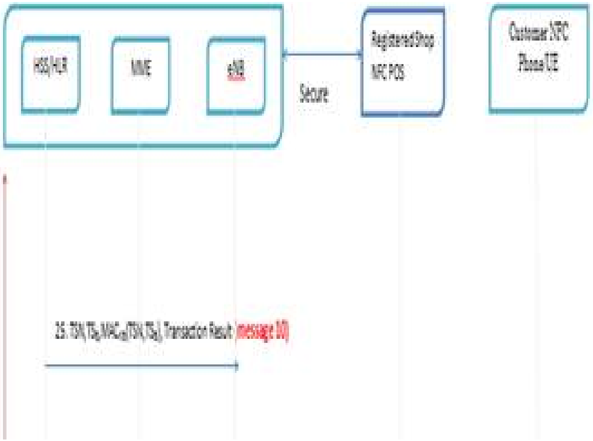
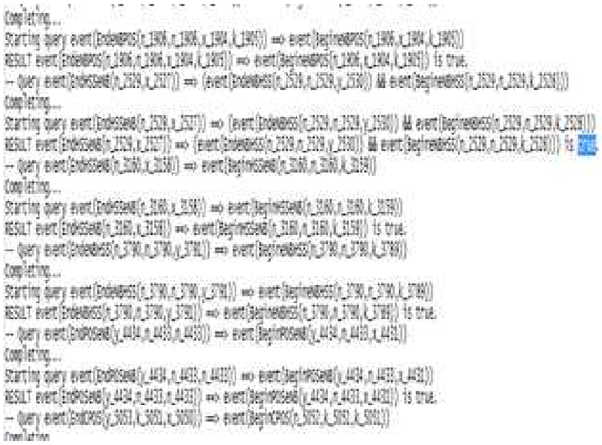

HSS/HLR
MME
eNB
Secure
Registered Shop NFC POS
Customer NFC Phone UE
25. TSN, TS_E, MAC_K(TSN, TS_E), Transaction Result (message 10)

Fig. 24 Transaction execution stage, Message 10

The proverif output for both message 9 & 10 shows true as the following screen shoot shows.



```
Completing...
Starting query event(endeNBPOS(r_1906,n_1906,x_1904,k_1905)) ==> event(beginNBPOS(r_1906,x_1904,k_1905))
RESULT event(endeNBPOS(r_1906,n_1906,x_1904,k_1905)) ==> event(beginNBPOS(r_1906,x_1904,k_1905)) is true.
-- Query event(endNFCSeNB(r_2528,n_2527)) ==> (event(endeNBNFCS(r_2529,n_2529,y_2530)) && event(beginNBNFCS(r_2529,n_2529,k_2528)))
Completing...
Starting query event(endNFCSeNB(r_2528,n_2527)) ==> (event(endeNBNFCS(r_2529,n_2529,y_2530)) && event(beginNBNFCS(r_2529,n_2529,k_2528)))
RESULT event(endNFCSeNB(r_2529,n_2527)) ==> (event(endeNBNFCS(r_2529,n_2529,y_2530)) && event(beginNBNFCS(r_2529,n_2529,k_2528))) is true.
-- Query event(endNFCSeNB(r_3160,n_3158)) ==> event(beginNFCSeNB(r_3160,n_3160,k_3159))
Completing...
Starting query event(endNFCSeNB(r_3160,n_3158)) ==> event(beginNFCSeNB(r_3160,n_3160,k_3159))
RESULT event(endNFCSeNB(r_3160,n_3158)) ==> event(beginNFCSeNB(r_3160,n_3160,k_3159)) is true.
-- Query event(endeNBNFCS(r_3790,n_3790,y_3791)) ==> event(beginNBNFCS(r_3790,n_3790,k_3789))
Completing...
Starting query event(endeNBNFCS(r_3790,n_3790,y_3791)) ==> event(beginNBNFCS(r_3790,n_3790,k_3789))
RESULT event(endeNBNFCS(r_3790,n_3790,y_3791)) ==> event(beginNBNFCS(r_3790,n_3790,k_3789)) is true.
-- Query event(endPOSeNB(y_4434,n_4433,n_4433)) ==> event(beginPOSeNB(y_4434,n_4433,x_4431))
Completing...
Starting query event(endPOSeNB(y_4434,n_4433,n_4433)) ==> event(beginPOSeNB(y_4434,n_4433,x_4431))
RESULT event(endPOSeNB(y_4434,n_4433,n_4433)) ==> event(beginPOSeNB(y_4434,n_4433,x_4431)) is true.
-- Query event(endNFCPOS(y_5053,k_5053,x_5050)) ==> event(beginNFCPOS(n_5052,k_5053,k_5051))
```

Fig. 25 Proverif results of both messages (9 & 10)

Message 12 shows the final result to the client whither success or failure.

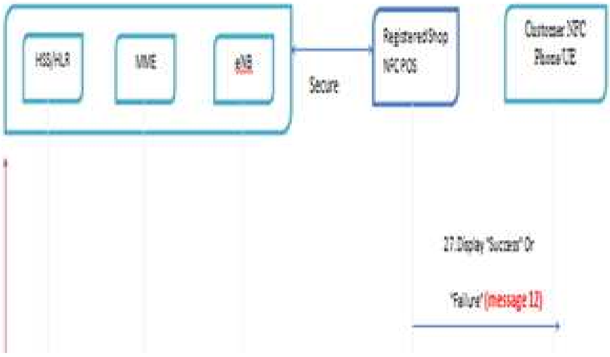


Fig. 26 Transaction execution stage, Message 12

저 자 소 개

Ebrahim AL-Alkeem

Holds Master of Science in Information security and bachelor degree in communication engineering from Khalifa University. In addition, he is in the final stage to word completing his PhD. He has also completed Specific coursework covering health care security, IoT security, and governance related issues. He has more than 12 years' of experience in governance and security related subject, his responsibility is to overseeing the security architecture and compliance of the government entities. Ebrahim presented much international and domestic speech and published many papers in security related field. In 2015, Ebrahim won Tamayaz Excellence Award, Appreciation award in the category of inventions of information technology & smart services, as well he is an active member of many of engineering association across the world.

Chan Yeob Yeun

Received his M.Sc. and Ph.D. in Information Security from Royal Holloway, University of London, in 1996 and in 2000, respectively. After that, he joined Toshiba TRL in Bristol. Then, he became a Vice President at LG Electronics, Mobile Handset R&D Center in 2005. He was responsible for developing the Mobile TV technologies and its security. He left LG Electronics in 2007 and joined at KAIST in Korea until August 2008 and moved to Khalifa University. He is currently serving as an Associate Professor of Electrical and Computer Department and an active member of Information Security Research Center. He currently enjoys lecturing M.Sc. in Information Security Courses at Khalifa University. He has published 28 journal papers, 71 conference papers, 2 book chapters and 10 international patent applications. He is also serving as several Editorial Board members of International Journals and a steering committee member of ICITST conference series and he is a senior member of the IEEE.

Yousef Al-Hammadi

Joined UAE University in May 2013 as assistant professor in information security. His research interests cover many areas in the fields of information security and algorithm. His research combines both theoretical and practical aspects, addressing applications on a number of domains, such as communication and protocols. Current research includes: ICT education and curriculum, access control systems, hash function, factoring and primality testing. Yousef accomplished PhD in information security, especially in the field of cryptography from Queensland University of Technology, Australia in 2006, Master degree in computer and information systems from University of Detroit, USA, in 1998, and BSc from UAE University in 1991 in Mathematics and computer science.

Hyun Ku Yeun

Received his BSc. and Ph.D. in Applied Mathematics from The University of Sheffield, in 1996 and in 2000, respectively. He became Mathematics and Physics faculty at Higher Colleges of Technology in the UAE since August 2011. He currently serves as Course Development Team Leader of Engineering Statistics for Bachelor of Engineering course and enjoys teaching various level of Mathematics in the Engineering department. His research interests are in areas of Fluid Mechanics, Number Theory, Cryptography and Convergence Technology such as Cloud Computing Security, Vitalization Security, Wireless Network and Mobile Security and Security for Internet of Things. The Data modelling, Statistical approach and Computer programming are also the favourite interest of his. He has two international patents in the field of ID based Cryptosystem, a number of conference and journal papers.

Young-Ji Byon

Received his BASc (2003) degree from the Department of Mechanical and Industrial Engineering at the University of Toronto; and MASc (2005), and PhD (2011) degrees from the Department of Civil Engineering at the University of Toronto, specializing in Transportation Engineering. From 2009 to 2010, he was a visiting researcher at the Universidad de Chile. From 2010 to 2011, he was a postdoctoral research fellow at the University of Calgary. He is currently an Associate Professor in the Department of Civil Infrastructure and Environmental Engineering at the Khalifa University of Science and Technology, Abu Dhabi, UAE.