NUST Journal of Engineering Sciences, Vol 9, No 2, 2016, pp 35-41

Security Protocol for NFC Enabled Mobile Devices Used in Financial Applications

Osama Faridoon, Abdul Ghafoor
School of Electrical Engineering and Computer Science, National University of Sciences and Technology (NUST)
Islamabad, Pakistan
12msccsofaridoon@seecs.edu.pk, abdul.ghafoor@seecs.edu.pk
Received: 5 September 2016 Accepted: 5 November 2016

Abstract

The fostering of NFC in everyday tasks and with growth in applications involving contactless transactions based on NFC, there is a requirement from users and industry to address the security issues affecting mobile payments. The current NFC security standards are inadequate to address most of the security concerns such as privacy infringements, unauthorized access to financial data, theft of mobile data exchanged between terminal and mobile device. In this paper, we designed a NFC based security protocol for financial applications, which addresses security requirements holistically and provides local and remote mutual authentication, confidentiality, integrity and non-repudiation. After designing, we verified our protocol using Scyther and established that it protects against spoofing attack, man-in-the-middle attack, replay and skimming attacks. It ensures the secrecy of transaction data, privacy of the users and also ensures that only authenticated and authorized NFC device holder and PoS terminals are securely exchanging financial data to perform the transaction. Furthermore, we developed a prototype system using java technology to show that the solution is practical and works according to our verified and designed specification.
Keywords: NFC, authentication, certificate, confidentiality, mobile transactions security

Introduction

The number of applications and services provided by contactless cards are increasing day by day. Therefore a person needs to carry more than one contactless card for different applications and institutions. A more practical solution for providing flexibility and scalability is to adopt Near Field Communication (NFC) enabled devices, which provides Secure Element (SE) for keeping all the applications secure and independent of each other in our mobile phone. Furthermore, the associated data of each application should be managed by the individual applications and should not be shared with each other. For smooth and seamless experience, both in proximity and in internet based remote applications, NFC enabled mobile phones are considered more practical solution.

The ever increasing processing power and storage with reduced cost has made smart phones fast, ubiquitous and part of our everyday life. With NFC being part of billions of smart phones, mobile payment market is increasing exponentially. According to ABI, a technology market research company, the number of NFC enabled devices will reach close to 1.95 billion in 2017 [1]. A number of different payment systems including MasterCard [2], Google Wallet [3], etc are using NFC feature because of its simplicity and convenience. But with varied financial services like payment systems for banks, billing, ticketing, buying medicine for a patient with history of disease etc, concerns about authenticity, confidentiality, privacy, integrity and reliability of the systems are arising.

The NFC security standards ECMA-385 and ECMA-386 provide only confidentiality and integrity. There is no protection from attacks against authentication, authorization and privacy mechanisms. As stated in [4], the privacy of users during payments is not ensured due to fixed keys used for confidentiality.

Our solution addresses the security requirements of mobile contactless payment systems including remote and local mutual authentication, confidentiality, privacy, authorization, integrity and non-repudiation. It is a generalized solution which can be implemented for any payment system and is independent of third party solutions. In this paper, section 2 describes the concept of mobile payment system, analyzes different options for secure element and dwells upon mobile payment security requirements. In section 3, related work to the security solutions and protocols for NFC is described. In section 4, the proposed architecture and protocol designed is explained and evaluated respectively. Section 5 is about proof of concept implementation. A comparison with other solutions is given in section 6. In section 7 of the paper, conclusion is given.

Background

NFC has evolved from RFID and is inherently insecure. Ecma International European association for standardizing information and communication systems developed a security protocol for NFC which lacks in a number of aspects like mutual authentication and privacy. There are a number of security vulnerabilities and attacks on NFC ecosystem comprising of NFC Endpoint Hardware, the NFC link/ communication and the backend infrastructure. These attacks include relay attack, cloning and skimming attack, eaves dropping, man in the middle, fuzzing attack, privacy, and denial of service attack. These attacks exploit vulnerabilities in all the operating modes of NFC consisting of reader/writer mode, peer to peer mode and card emulation mode, NFC Data Exchange Format (NDEF) and application loading and personalization using Over the Air (OTA) link.

Secure Element

According to Global platform [6] a Secure Element (SE) is an architecture which resists tampering and is used for

securely hosting applications and their cryptographic credentials. It is a secure microcontroller containing a processor, operating system, different types of memories, crypto engines, sensors, timers, random number generators and communication ports[7]. In Secure Element (SE) different applets representing various physical smart cards are executed on virtual machines.

Generally the form factor for Secure Element (SSE) can be divided into Removable and Non Removable Elements. Removable form includes Universal Integrated Circuit Card (UICC) and micro SD whereas non removable is in the form of embedded chip. Each of the form has its own usage scenarios along with specific and pros and cons.

Due to strong security, remote access control and over the air provisioning of the application and credentials, the most suitable choice for SE is sim based. The embedded chips for secure element also provide strong security and protection from different attacks but lacks in flexibility and scalability, as it involves a tedious task of transferring applications from one handset to the other [8]. Micro SD card option for SE is independent of mobile operators and Trusted Service Managers (TSMs) but it lacks standards and specifications yet. Also it will require multiple cards slots in a handset for multiple applications [9]. In cloud based secure elements the sensitive data is stored on the cloud in the encrypted form and is transferred to the mobile phone for transaction; decrypted and then transferred to the reader or PoS terminal [10]. Although it solves the problem of limited space and computing but the bottle neck is the network latency and encryption decryption complexity at the mobile phone. In addition, it gives a large surface area for launching attack. The concept of Host Card Emulation (HCE) introduced by Google does not rely on secure element and hence not dependent on mobile network operators, equipment manufacturers and TSMs. It allows the applications on android device to both emulate NFC card and NFC reader by allowing the applications to directly communicate with the NFC antenna [11][12]. The security of Google's HCE is yet to be proved, especially in case of multiple security sensitive applications using NFC.

The more secure an application or a solution is, the less easiness and flexibility it provides. In the end it all comes to the compromise between cost, ease and security of different solutions. We also need thorough standardization and certified security test-ing of various solutions of Secure Element (SE) implementation.

Mobile Payment System

Mobile payment refers to the use of mobile device to pay bills instead of paying with credit cards. Proximity mobile payments involve transactions, having consumer interacting with a PoS terminal using mobile device like contactless NFC payment [5].An NFC payment constitutes an NFC enabled mobile phone personalized with a payment application and account respectively, waved or held near a PoS terminal capable of contactless payment. As mobile phones have modest cost and widespread ownership, the integration of contactless payment in mobile handsets is the main reason of its wide acceptance as the dominant mode of payment. Also the builtin display and keyboard is very handy for the confirmation, code entry or activation/deactivation of the payment functionality. Since the payment processing infrastructure for the proximity contactless payment is the same as that for contact payments, confidentiality, authentication, integrity, authorization, non-repudiation and privacy are the security requirements of the mobile NFC contactless payments transactions taking place.

Literature Review

Security in NFC is yet to be matured and is in the developing stage. Most of security solutions for NFC are scenario based and their dependence on third party makes them less promising. Also the standardization and certification process for NFC security is not yet completed. There is a need to reorganize the roles of different entities in NFC ecosystem, giving more independence and flexibility to developers.

In [13] Husni et.al described a protocol for NFC mobile payment with the assumption that the merchant has the knowledge of payment amount. This very assumption limits the applicability of the protocol, as the transaction amounts can vary according to requirement. The protocol is vulnerable to man-in-the-middle attacks and also lacks privacy as only a part of message is encrypted using a fixed key. It does not provide authentication of the entities involved as well. The establishment of a shared secret is a difficult task and also hinders the scalability. In [4] Hasoo et.al take care of privacy of the users in electronic payments by hiding the public key with the pseudonyms in the NFC security protocol, which prevents the attacker from profiling users. Another solution is provided for mutual authentication between NFC device and Point of Sale (PoS) Terminal based on trusted third party verification of NFC phone and PoS by Ceipidor et.al [14]. The verification depends on the validation of timestamp and shared symmetric key, which can be easily exploited by man-in-the-middle and DoS attacks. The whole authentication process is subject to the security of Authentication Server and its availability. In [15] author gives security protocol for the mobile payment system using GSM network.

In Europay, MasterCard and Visa Corporation EMVCo security provided in contactless payment is similar to that of chip card based payment. The mobile device plays the role of card. It does not play the role of PoS terminal. EMV secures the con-tactless payment in a similar way as in chip card based payment systems. There is no authentication of the PoS terminal. Only card is authenticated [16] either online or offline. For offline authentication, the IC card data signed with the private key of the issuer is verified by the terminal. In case of online authentication [17], the dynamic data for verification to be performed by the terminal includes that provided by the IC card, the terminal and the transaction specific data. The public key certificate of the IC card is also verified by the terminal.

In [18] Nadra et.al proposed architecture for payment in cafeteria. The users need to register themselves. Privacy is not catered as the data is not encrypted. The system gives no protection from man in the middle attack. Scalability is also an issue. The key generation and revocation is not addressed.

Google wallet provides security by requiring a PIN to access application and disabling antenna upon locking of the screen or completion of the transaction [8]. It limits the app to access only a part of your credit card information which is stored in encrypted form on the servers. It also allows online disabling of the Google wallet account and clearing the transaction data [19]. However, still there are issues which can cause security breach and need to be addressed. According to Nelenkov [20] during installation of the Google Wallet application, application protocol data units (APDUs) are not completely encrypted and hence can be subject to man-in-the-middle attack. Also the access control for SE needs more restrictions on the applets and APDUs for them. The MIFARE manager applet, which is part of the application, is susceptible to attacks on authentication and encryption [20]. In the new versions of the application, the intention is to store and verify the pin inside Secure Element (SE) but still pin ownership issues can arise in case of any breach. [19].

In [21], Mainetti et.al described the solution to store the encrypted credit card information in the memory of the android phone instead of the Secure Element (SE). The Point of Sale PoS terminal acts as a Relaying Party (RP) to forward the encrypted file to the payment gateway which is a server in front of financial institution. The file is first encrypted using public key of the gateway server and then using symmetric key. The symmetric key is derived from the PIN, entered by the user. The key distribution, confirmation and revocation for public key encryption are not elaborated in detail. Also the authentication of the user and authorization is based on the pin which can be subject to brute force attack.

In [22] Urien et.al presented authentication protocol for peer-to-peer payment transactions based on Logical Link Control Protocol (LLCP) secured by Transport Layer Security (TLS) in a retail environment. LLCP is an OSI layer-2 protocol which enables peer-to-peer bidirectional communication between two NFC enabled devices [23]. The authentication of the customer is done by signing in with NFC enabled phone at the sign in terminal upon entering the store. Then customer scans the bar codes on different items to be purchased with his smart phone. For payment the customer brings his phone close to the PoS Terminal, which processes the transaction. In this way, it removes the bottle neck of scanning the items at the PoS terminal. During the initial sign-in process, the keys are transferred using elliptic curve based public key cryptography. Cha and Kim described a payment system for retailers based on message digest authentication and tokenization for providing privacy [24]. Based on sensitive data related to the customer, a token is generated to replace the original information. The original data is stored in the encrypted form on payment gateway server. The tokens are exchanged between client and the payment gateway server encrypted symmetrically. Although this approach is simple but the overall system security is dependent on the security of the server and the symmetric encryption.

In [25], the trial of a contactless mobile payment via pay pass terminal compatible with EMV is presented. The access to payment application is given on entering the PIN. The user is authenticated using combined data authentication for online transaction. The author used common criteria to evaluate the security of the NFC link between application and PoS terminal and the results presented in the paper were satisfactory.

As evident, the existing solutions and work is scenario based and lack of system, hardware or platform independence. Most of the solutions provide protection from only common limited attacks to mobile payment systems. If any work caters the privacy, it lacks the authentication aspect of the mobile payments. Also mutual authentication is lacking, which is very important aspect of financial transactions.

Design of Architecture for NFC Based Financial Transactions

Our designed protocol consists of the entities as shown in the Figure 1. These entities are: Identity Management System, Certification Authority, Authentication and Authorization Server, Mobile PoS terminal, and NFC enabled mobile phone.

— **Identity Management System (IDMS)** In this system, it is assumed that all valid users of the payment system are registered in IDMS. It maintains identities, required information about the users, and information about NFC mobile. It manages all the steps of Identity Management Life Cycle. It also holds the database of users PIN which are stored in hashed and encrypted format.

— **Certification Authority (CA)** CA issues public key certificates to all NFC mobiles used by the customers and also the mobile PoS terminals. The verification of users and PoS terminal's certificates is also done by CA. During the personalization of card and PoS terminal, the CA uses its private key to sign the sensitive data and the public key certificates. The signed data and the CA's public key are loaded in the secure element of mobile device.

— **Authentication and Authorization Server** Based on our security protocol, this server authenticates the NFC mobile applet of the customers and also authorizes the transactions. The users are authenticated on the basis of their verification against the IDMS identity database and Certification Authority. We used certificate and challenge based authentication. The authorization of transactions to the users is managed by using XACML 3.0 standard.

— **Mobile PoS Terminal** Financial applications are network sensitive and require end to end node security. For financial wireless applications in a distributed environment, a security mechanism at the application layer is critical. Mobile Point of Sale (MPoS) terminal contains our application, which carries financial transaction over NFC. The application is securely stored in Secure Element (SE) and the transactions are secured over the wireless link through our protocol. The mobile PoS terminal is certified. It passes the certificate of the customer and his identity to authentication and authorization server for verification.

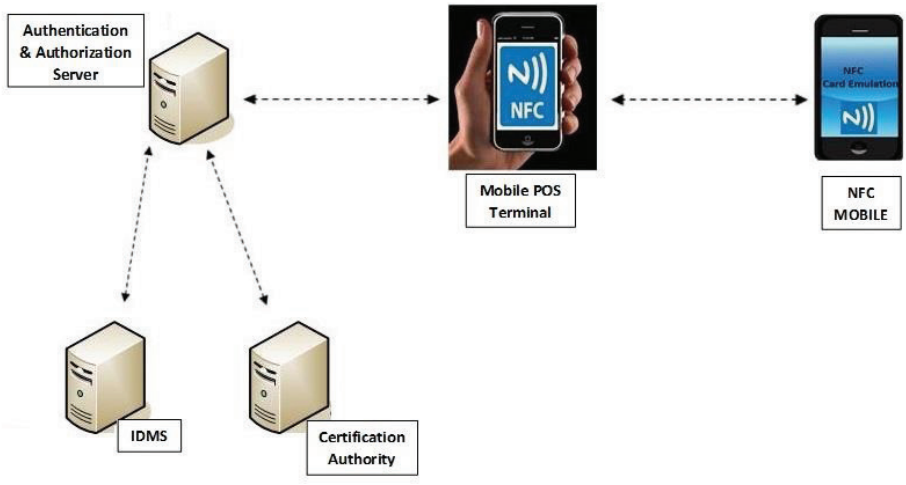

Fig. 1: Diagram for NFC based Financial Transactions Architecture

It also forwards PIN hash to authentication and authorization server for verification and hence authorizes the transaction. Once the certificates are verified and pin is matched and challenge response is complete, it allows the transaction and also secures the link between itself and NFC mobile through encryption.

— **NFC Enabled Mobile Phone** NFC enabled mobile phone's NFC chip card contains an applet that processes the requests from the PoS application.

Security Protocol for NFC Enabled Mobile Devices

When NFC enabled mobile comes in the range of Mobile PoS terminal, the terminal initiates the communication by generating a field and requests for the certificate. The card responds by sending its certificate for authentication. The initial communication comprising of certificates exchange and challenge response is for the mutual authentication of NFC mobile and terminal. Once both the ends are authenticated and the keys have been verified, then the one time session key is sent by the terminal to the NFC mobile using public key encryption. The symmetrically encrypted pin is sent by the NFC mobile to the terminal, authenticating it locally to carry out the transaction. The transactions are encrypted and hashes are sent to ensure the integrity of the transactions. Authentication and authorization server requests updates after regular interval from CA and IDMS. CA and IDMS respond by sending the latest changes in certificates and identities respectively. Authentication and authorization server updates its database.

As NFC mobile is in the range of mobile PoS terminal, the PoS terminal requests the card for certificate. The NFC mobile sends its certificate containing public key of the card and ID of the card and applet. The PoS terminal forwards the IDs and certificate to the Authentication and Authorization server for verification.

The PoS then sends its certificate. NFC mobile applet extracts the certification authority root key and authenticates the POS terminal's certificate, validating that the certificate is issued by the correct issuer and is authentic. This verifies the certificate and ID and activates the relevant applet. Once the NFC mobile applet certificate is verified, the PoS terminal sends the challenge to the NFC mobile and also retains it.

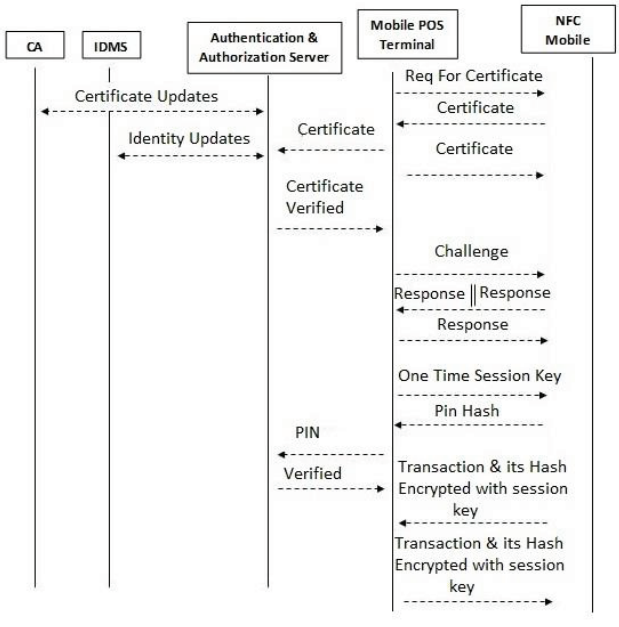

Fig. 2: NFC based Security Protocol

Challenge from the PoS terminal: $T_N \mid T_t \mid T_{ID}$

Where T_N is nonce generated by the terminal, T_t is the timestamp and T_{ID} is the identity of the PoS terminal. The NFC mobile applet signs the challenge with its private key using RSA. And also generates a challenge for the PoS terminal and sends it to the terminal.

$$C_{ID} \mid RSA_{Sign}\{(T_N \mid T_t \mid T_{ID}), K_{Priv}\} \mid C_N \mid C_t$$

Where C_N is nonce generated by the card, C_t is the timestamp and C_{ID} is the identity of the NFC mobile applet.

The terminal decrypts and verifies the response. It then signs the challenge with its private key using RSA and sends it to the card.

$$RSA_{Sign}\{(C_N \mid C_t \mid C_{ID}), K_{Priv}\}$$

The NFC mobile applet decrypts the response and verifies it. After the verification of responses from both the sides, the authentication, both remote (via certificates) and local authentication (via challenge response) is completed. Challenge response ensures that the entity is available and willing, whose certificate is being verified providing nonrepudiation. It also ensures whether the person or the device having the certificate also possess the corresponding private key. This process also helps avoid fake, forged and stolen certificates. There are two steps involved in the verification of certificate. The first step is to verify originality of certificate by signature verification. Then revocation and expiry of the certificate is checked.

The PoS terminal then generates One Time Session Key and sends it to the NFC mobile for use in encryption during the session. The key is sent, first signed with the private key of the terminal and then encrypted with the public key of the NFC card.

RSA_{ENCRPT} {RSA_{Sign} (One Time Session Key, K_{Priv}), K_{Pub}}

Once the one time session key is exchanged, rest of the session is encrypted using AES. For local authentication the NFC mobile applet calculates the PIN hash using message digest algorithm SHA and sends it encrypted with the session key. Both AES and SHA have NIST approved lightweight variants [28].

$$AES_{ENCRYPT} \text{ (Pin Hash, } K_{Session})$$

The terminal decrypts it and sends it to the Authentication and Authorization server for verification against the hash database. All of the financial transactions are sent encrypted with One Time Session Key using AES along with their hash calculated with using SHA.

$$AES_{ENCRYPT} \text{ (Transaction +Hash, } K_{Session})$$

Implementation of Designed System

The proof of concept consists of a mobile PoS application for Android operating system developed in Java. The corresponding java applet has been developed by using java card framework 3.0 Classic Edition uploaded.

The PoS application detects (discovers) the target card and initializes java applet. After establishing connection, PoS Terminal sends APDU to fetch the certificate stored in the smart card. The sequence of commands and response APDUs are as follow:

// code of CLA byte in the command APDU header for Security Protocol NFC

```
final static byte SP_CLA = (byte) 0x80;
```

// codes of INS byte in the command APDU header
//Certificate request form APP from PoS

```
final static byte CERT_REQ_FROM_APP =
(byte) 0x10;
```

//Challenge from PoS

```
final static byte CHALLENGE = (byte)
0x20;
```

//Session Key from PoS

```
final static byte SESSION_KEY = (byte)
0x30;
```

//Certificate request form CARD from PoS

```
final static byte CERT_REQ_FROM_CARD =
(byte) 0x10;
```

Since first request from mobile PoS terminal is request for certificate so the NFC card will fetch the certificate from its buffer and then sends back to the terminal. The send certificate function of our Applet performed this task as shown in following piece of code.

```
case CERT_REQ_FROM_APP:
                sendCerticate(apdu);

      return;
```

After exchanging certificates, the PoS Terminal sends challenge to the NFC card. The NFC card's applet received this challenge and signs with private key corresponding to the card's certificate. We implemented this function in signChallenge-WithRSA(apdu) as specified in following code.

//if incoming APDU request is to sign challenge.

```
case CHALLENGE:
{m_sign=Signature.getInstance(Signature
.ALG_RSA_SHA_PKCS1, false);
```

// initialize signature with private key and a signature mode

```
m_sign.init(m_privateKey,
Signature.MODE_SIGN);
```

// encrypting incoming challenge

```
short signLen= m_sign.sign(buffer,
ISO7816.OFFSET_CDATA, (byte) num-
Bytes,m_ramArray, (byte) 0);
```

//sending encrypted challenge

```
short le = apdu.setOutgoing();
apdu.sendBytes(m_sign, signLen, le);}
```

The other functions specified in the design of protocol are implemented in the same fashion.

Analysis of Protocol: Evaluation and Comparison

After designing security protocol for NFC mobile used in financial application, we analyzed it with Scyther , which is an automated security verification tool. The complete formal code for verification is available. The results generated through Scyther are shown in Fig. 3. In this verification, we assumed that all the credentials and sensitive data are safely stored in tamper resistant chip (SE) at POS terminal as well as NFC mobile. The adversary has access to the mobile PoS terminal and the NFC link between the PoS terminal and the NFC mobile.

Table 1: Comparison of NFC based Protocols for Financial Transactions

	Mutual Authentication	Confidentiality	Integrity	Non Repudiation	Authorization	Prior Knowledge	Third Party	Privacy
Husni et.al		✓	✓		✓	✓	✓	
Hasoo et.al		✓	✓				✓	✓
Ceipidor et.al		✓			✓		✓	
Ali et.al		✓	✓		✓		✓	
EMVCo		✓	✓		✓			
Nadra et.al			✓					
Google Wallet		✓	✓		✓		✓	
Mainetti et.al		✓			✓			
Urien et.al	✓	✓	✓		✓	✓		✓
Cha & Kim	✓	✓	✓			✓		✓
Pasquet et.al		✓	✓		✓			

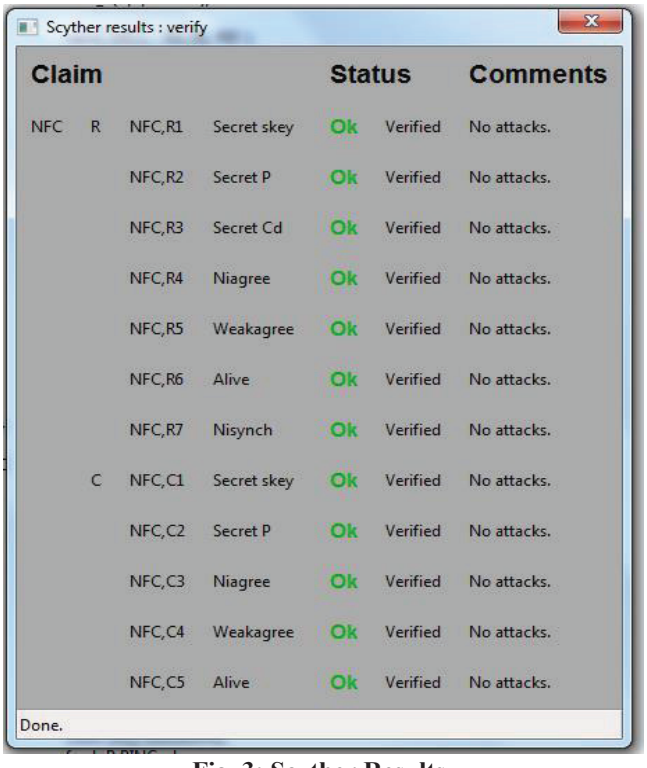

Fig. 3: Scyther Results

We specified in it, that the adversary can modify and create messages between the two ends in order to tamper the messages and launching replay attack as man-in-the-middle. The secrecy attribute of the verification expresses that the confidentiality of the session key, PIN and the transaction data over an untrusted network is secured by our protocol during execution.

Authentication is the guarantee that the intended and aware partner is communicating in the network and as the protocol runs, the messages are transmitted exactly in the desired sequence. It is described by the properties of aliveness and synchronization. Aliveness property ensures that the communicating parities, both NFC mobile and PoS terminal are present and responding to each other. Synchronization is the exact exchange of the messages between the corresponding runs, for every entity as described in the protocol. It is possible, by ensuring that each entity performs its described role in the protocol. Agreement property of tool describes that, both the entities after successful corresponding execution, agree on the values of data variables. Satisfying aliveness, injective synchronization, non- injective synchronization and non-injective agreement property ensures the protection against man-in-the-middle attack, spoofing, skimming and replay attacks. Signed messages are used to meet the non-repudiation requirement of the protocol. As shown in the table below, compared to other protocols [5] [6] [9] [10] [12] for NFC based financial applications, our protocol provides a solution to most of the existing problems for NFC transactions such as confidentiality, mutual authentication, integrity, authorization and non-repudiation. Specifically mutual authentication is considered the most important aspect for any financial transaction. Our protocol not only ensures that the certificates are verified but also makes sure that the entity claiming a public key has the corresponding private key. In other words, it does not only authenticate the existence of an entity but also ensures its willingness to communicate and nonrepudiation. The one time session key provides privacy to the user, in case the attacker is observing the shopping preferences and its intervals. Another advantage of our protocol is that, it is not limited to any specific domain or environment and can be applied for healthcare system, NFC immigration system and Physical Access Control etc. Furthermore, the designed protocol is not dependent on any third party for its functioning as in case of [7] [8] [13] [14].

Conclusion

The designed security protocol is based on some common and extended security features which help to increase the reliability of NFC based systems. Furthermore, it may be beneficial to the financial organizations in increasing their user's trust, for secure usage of their mobile devices for financial transactions. As a proof of concept, we designed and implemented our solution for android based NFC mobile devices and successfully deployed it in our local environment to test its correctness and behavior. In addition to that, we verified designed approach through formal verification tools and found that it resists against various potential attacks launched against authentication and secrecy of exchanged data. At the end of paper, we also provided a comprehensive comparison of our protocol with other NFC based financial protocols. We found that the mutual authentication, confidentiality, integrity, authorization and non-repudiation services help to protect against most of the security attacks related to mobile

financial transactions. Since this protocol is flexible, generalized and reliable, so the whole system is not depended on the third parties and any prior knowledge

REFERENCES

1. A. Research, "https://www.abiresearch.com/press/nfc-will-come-out-of-the-trial-phase-in-2013-as-28," 2014.
2. Mastercard, "http://newsroom.mastercard.com/press-releases/mastercard-to-use-host-card-emulation-hce-for-nfc-based-mobile-payments," 2014.
3. Wikipedia, " http://en.wikipedia.org/wiki/Google_Wallet," 2014.
4. H. Eun, et al., "Conditional privacy preserving security protocol for NFC applications," IEEE Transactions on Consumer Electronics, vol. 59, no. 1, 2013, pp. 153-160.
5. S.C. Alliance, "The mobile payments and NFC landscape: A US perspective," Smart Card Alliance, 2011, pp. 1-53.
6. Globalplatform, "http://www.globalplatform.org/mediaguideSE.asp.," 2014.
7. S. allaince, "http://www.smartcardalliance.org/resources/webinars/Secure_Elements_101_FINAL3_032813.pdf," 2014.
8. O. Ghag and S. Hegde, "A comprehensive study of google wallet as an NFC application," International Journal of Computer Applications, vol. 58, no. 16, 2012.
9. C. Li, et al., "A trusted virtual machine in an untrusted management environment," IEEE Transactions on services computing, vol. 5, no. 4, 2012, pp. 472-483.
10. N. world, "http://www.nfcworld.com/2012/09/25/318059/inside-secure-to-offer-cloud-based-nfc-secure-element-solution/."
11. 2014, "http://developer.android.com/about/versions/kitkat.html#44-hce."
12. , "http://tomnoyes.wordpress.com/2013/11/01/hce-kills-isis/," 2014.
13. E. Husni, et al., "Efficient tag-to-tag Near Field Communication (NFC) protocol for secure mobile payment," Proc. Instrumentation, Communications, Information Technology, and Biomedical Engineering (ICICI-BME), 2011 2nd International Conference on, IEEE, 2011, pp. 97-101.
14. U.B. Ceipidor, et al., "KerNeeS: A protocol for mutual authentication between NFC phones and POS terminals for secure payment transactions," Proc. Information Security and Cryptology (ISCISC), 2012 9th International ISC Conference on, IEEE, 2012, pp. 115-120.
15. T. Ali and M.A. Awal, "Secure mobile communication in m-payment system using NFC technology," Proc. Informatics, Electronics & Vision (ICIEV), 2012 International Conference on, IEEE, 2012, pp. 133-136.
16. C. Markantonakis and K. Rantos, "On the life cycle of the certification authority key pair in EMV'96," Proceedings of Euromedia'99, 1999, pp. 125-130.
17. S.C. Alliance, "Card Payments Roadmap in the United States: How Will EMV Impact the Future Payments Infrastructure?," Book Card Payments Roadmap in the United States: How Will EMV Impact the Future Payments Infrastructure?, Series Card Payments Roadmap in the United States: How Will EMV Impact the Future Payments Infrastructure? , ed., Editor ed.^eds., September, 2012, pp.
18. , "http://www.google.com.pk/wallet/faq.html#tab=faq-security. ," 2014.
19. Nelenkov, "http://nelenkov.blogspot.com/2012/08/exploring-google-wallet-using-secure.html#!/2012/08/exploring-google-wallet-using-secure.html.."
20. L. Mainetti, et al., "IDA-Pay: an innovative micro-payment system based on NFC technology for Android mobile devices," Proc. Software, Telecommunications and Computer Networks (SoftCOM), 2012 20th International Conference on, IEEE, 2012, pp. 1-6.
21. P. Urien and S. Piramuthu, "LLCPS and SISO: A TLS-based framework with RFID for NFC P2P retail transaction processing," Proc. RFID (RFID), 2013 IEEE International Conference on, IEEE, 2013, pp. 152-159.
22. N. Forum, " http://members.nfc-forum.org/specs/spec_list/," 2014.
23. B. Cha and J. Kim, "Design of NFC Based Micro-payment to Support MD Authentication and Privacy for Trade Safety in NFC Applications," Proc. Complex, Intelligent, and Software Intensive Systems (CISIS), 2013 Seventh International Conference on, IEEE, 2013, pp. 710-713.
24. M. Pasquet, et al., "Secure payment with NFC mobile phone in the SmartTouch project," Proc. Collaborative Technologies and Systems, 2008. CTS 2008. International Symposium on, IEEE, 2008, pp. 121-126.
25. S.tool, "www.cs.ox.ac.uk/people/cas.cremers/scyther/," 2014