DISCUSSION OF A KEY EXCHANGE PROTOCOL BETWEEN UN-KEYED SIM CARD AND SERVICE PROVIDER

[1]KEREM OK, [2]CEM CEVIKBAS, [3]MOHAMMED ALSADI, [4]VEDAT COSKUN, [5]BUSRA OZDENIZCI

Abstract— Advances in mobile communication technologies, smart cards and Smartphones enables companies to offer valuable services to mobile users such as mobile payment, ticketing, loyalty applications and so on. Obviously, for enabling secure services, end-to-end encryption between a Service Provider and a SIM card is an important requirement. Keyed SIM cards have the required infrastructure for secure key generation and exchange; however un-keyed SIM cards do not provide the required structure. This study aims to present a novel key exchange protocol for un-keyed SIM cards; hence SIM card and Service Provider can perform end-to-end data encryption. Security and threats discussion of the proposed protocol is provided as well.

Keywords— Key exchange protocol, SIM card, Smart card, Symmetric, End-to-end Encryption, Security.

I. INTRODUCTION

With the advancements in mobile communication technology, many value added services are provided to users through Smartphones. At the same time with the evolution of smart card technology, SIM cards store more data, perform faster calculations and provide advanced security functions. Latest SIM cards provide advanced security services such as public key cryptography, mobile signatures, and remote key generation. So that, Service Providers (SPs) are willing to provide value added services on SIM cards such as mobile payment, ticketing, loyalty which requires a secure end-to-end encryption between SIM card and SP.

Today, corresponding security keys are embedded to the SIM cards at manufacturing phase; hence SIM cards provide secure end-to-end encryption between SIM card and SP. However, most SIM cards those dispensed to the users today do not have embedded keys that can be used for end-to-end encryption between SIM card and SP [1].

End-to-end encryption between SP and the SP's application on SIM card is a must requirement for secure services. A SP needs to be sure that no one can modify the communication conducted with the user. Data should be appropriately encrypted using a secure protocol using a proper key length. Considering the property of the SIM cards, using symmetric encryption protocols [2-7] is favorable.

In keyed SIM cards, MNOs initially issue SIM Cards to the users. As an SP wishes to offer a secure service via SIM Card, it makes an agreement with the MNO to use a specific slot –say, slot n– of the SIM card. After the agreement is signed, MNO notifies CI about the agreement. Then, CI shares the corresponding slot key with the SP. After SP gets the key, it can install an application to the slot n of the SIM card; and SP can communicate with the application running in the specified slot securely by using the slot key (Kn). In order for a SP to communicate securely with an un-keyed SIM card, there exist two solutions.

The first alternative is to replace un-keyed SIM cards with keyed ones. Actually this option creates an enormous cost due to the production cost of the contemporary keyed SIM cards and the cost of shipping & handling the keyed cards to users. Hence, a lower cost and yet satisfactory solution is still required.

The second alternative is to build an infrastructure for generating required keys at both sides using a key exchange protocol which is the aim of this study. However, this option is not a straightforward solution though. There are some difficulties through the protocol. First of all, un-keyed SIM cards have limited storage capacity and their programming capabilities are also limited as well.

In this paper, we describe our proposed end-to-end key exchange protocol between an un-keyed SIM card and Service Provider in which both parties are not equipped with an encryption key at the beginning. Then we briefly discuss the security issues behind our proposed protocol.

II. A SECURE KEY EXCHANGE PROTOCOL

In this section, we describe the proposed protocol, which creates the required infrastructure for un-keyed SIM cards. We name overall protocol as *SIMSec*, the application that implements the protocol on the SIM card as *SIMSec Card Application*, and the application on the SP server as *SIMSec Server Application* [8].

There exist some key exchange protocols in the literature, which perform interactive key generation solutions for keyed SIM cards only; those protocols do not consider the constraints of un-keyed SIM cards. The memory size and the computing power in un-keyed SIM cards are so small that the corresponding SIM card application should be small in size, should be efficient, and should use only the

functions that the card provides. *SIMSec* protocol is developed specifically taking into account of these SIM cards' constraints.

The *SIMSec* protocol is given in study [8]. We shortly describe the protocol's important issues; the values and hash functions used in the protocol first.

In the first process, the SP generates a random 10 characters long value, V and then sends this value to the SIM card.

Then, the SIM card randomly generates a private Diffie-Hellman secret value, a. Afterwards; it calculates g^a and computes the hash of $ID_{SIM} \| V \| g^a$ values. Then it sends this hash value with g^a to the SP.

If the SP receives the packet from the SIM card in the pre-set time period after the exchange of V, it randomly generates a private Diffie-Hellman secret, b.

Then the SP calculates hash value X' which is expected to be same as X. If two hash values are not equal, then the SP terminates the protocol.

If two hash values are equal, the SP authenticates the SIM card by calculating g^b and $(g^a)^b$ values. The SP calculates Y value as the hash of $ID_{SIM} \| V \| g^a \| g^b \| (g^a)^b$ as well as calculates K by computing hash of $ID_{SIM} \| V \| (g^a)^b$ values. After calculating the K value, SP sends Y and g^b values to the SIM card.

After the SIM card receives the packet, it calculates $(g^b)^a$ which should give the same result with $(g^a)^b$ that is calculated by SP. Then, SIM card calculates the Y' value which should be equal to the Y value that is calculated by SP. The only difference between Y' and Y is that when calculating Y', $(g^b)^a$ is inputted to the hash function instead of $(g^a)^b$. Then, SIM card calculates same K value and selects the K value as a key. Note that all g^a, g^b, $(g^a)^b$, $(g^b)^a$ calculations are in mod p.

p is a public variable which is used as the modulus in all computations. For a secure key exchange, the p value is selected to be a prime number with a length of at least 1024 bits [9].

g is a public variable that is used as a base variable in exponentiation operations.

a is generated randomly by the SIM card, and it remains private. The SIM card uses this value as a power in exponentiation. For a secure key exchange, the length of a value must be at least 384 bits [10].

b is generated randomly by the SP, and it remains private. SP uses this value as a power in exponentiation. For a secure key exchange, the length of b value must be at least 384 bits [11].

V is 10 characters long random variable. In *SIMSec* protocol, V value is sent to the SIM card owner using a different communication channel than other messages.

ID_{SIM} is the identification data of the SIM card. Unique digits of IMSI and ICCID are used to form ID_{SIM} value and the length of ID_{SIM} is 96 bits. Since both MNO and SIM card are capable of calculating this value, SIM card calculates this value itself. When required, SP receives ID_{SIM} value from MNO via their existing secure communication channel before the protocol starts. This secure communication channel generally exist between MNOs and SPs; otherwise they need to set it up.

H_1 implements a hash function and uses 128 Least Significant Bits (LSB) of the output. SP and SIM card should agree on the hashing function that they will use when developing the *SIMSec* application. Following data is inputted to the each hash function in the provided order:

1. 32 bit function type (1 for H_1)
2. 32 bit value for the main input's bit length
3. Main input which is the concatenation of ID_{SIM}, V and g^a (mod p)

H_2 implements a hash function and uses 128 LSB bit of the output. Following data is inputted to the hash function in order:

1. 32 bit function type (2 for H_2)
2. 32 bit value for the main input's bit length
3. Main input which is the concatenation ID_{SIM}, V, g^a (mod p), g^b (mod p), and $(g^b)^a$ (mod p)

H_3 implements a hash function and uses 168 LSB bit of the output. Following data is inputted to the hash function in order:

1. 32 bit function type (3 for H_3)
2. 32 bit value for the main input's bit length
3. Main input which is the concatenation ID_{SIM}, V, $(g^b)^a$ (mod p)

SIM card and Service Provider uses an SMS channel that is controlled by the MNO. Only V value is exchanged via an alternative communication channel. On the other hand, the exchange of ID_{SIM} value between MNO and Service Provider is performed using a secure communication channel. This secure communication channel generally exist between MNOs and Service Providers.

III. SECURITY DISCUSSION

There are some possible threats that can challenge *SIMSec* protocol. While key is being exchanged between SIM card and Service Provider, an intruder may eavesdrop the communication and try to hack the generated key. In order to address all possible threats, the following security requirements are identified:

- Confidentiality of the Key: In *SIMSec* protocol, an unauthorized third party including MNO should not be able to discover the key, thus should not be able to listen or alter the communication after the key is established. The key exchange protocol is developed based on Diffie-Hellman methodology. In this methodology, calculation of $(g^b)^a$ or $(g^a)^b$ values

by an unauthorized third party without knowing numbers a and b is not possible. In order to generate the key, the attacker needs to calculate at least one of these numbers; so the confidentiality of the key between SIM card and Service Provider is ensured.

- Data Integrity: In *SIMSec* protocol, the receiver should recognize any modification to the data from an unauthorized third party. Service Provider and SIM card checks the incoming hash value with the calculated one and recognize the modification in the data, if any. As a result, the protocol satisfies the integrity of the data between SIM card and Service Provider.
- Authentication of SIM card to Service Provider: Service Provider authenticates the SIM card in the protocol by using V value in hash functions. Since V value is exchanged via a secondary authenticated channel and only the user of the SIM card has access to this channel, Service Provider authenticates the SIM card. SIM card's id value is also used in hash functions. SIM card's id value is private and known only by the SIM card, the MNO, and the other Service Providers that SIM card run the protocol with. The MNO shares this data with the requesting Service Provider via a secure channel maintained between them prior to the protocol. When the SIM card sends a packet to Service Provider, the packet also includes the hash of SIM card's id; Service Provider checks the hash and ensures that the V value is used by the claimed SIM card.
- Authentication of Service Provider to SIM card: SIM card authenticates Service Provider by using V value in hash functions. Since this V value is exchanged via a secondary authenticated channel and only the SIM card user and the Service Provider knows the value, the SIM card authenticates the Service Provider.
- Man in the Middle (MITM) Attack Protection: In the key exchange protocol, an unauthorized third party should not be able to perform a MITM attack. In *SIMSec* protocol, V value is created by the Service Provider for one time use only and exchanged with the SIM card user from a secondary communication channel as described in previous sections. The MNO does not control this channel and is not able to retrieve V value from this communication channel. As it is seen in the protocol, when the SIM card sends a value to the Service Provider and when the Service Provider sends a value to the SIM card, V value is used in hash functions. If an unauthorized party including MNO tries to perform a MITM, it needs to guess this value, which has around 2^{60} possibilities (10 characters long, 64 possibilities for each character). Since V value is used by the Service Provider and the SIM card for one time only, and if the Service Provider or the SIM card identifies an unequal hash, the receiving party terminates the communication and a new V value needs to be generated. As a result, using V value protects the protocol from a possible MITM attack from an unauthorized party including the MNO. As the computing capabilities of attackers increase in time, the length of the V value can be increased accordingly.
- Replay attack Protection: In the protocol, the SIM card and the Service Provider accepts only one packet for exchanged V value. Thus, performing a replay attack by repeating a packet is not possible, since the receiver will reject it. Moreover, there is a small period that V value can be used, so when an unauthorized party performs a replay attack by delaying a packet, it is rejected after this period ends.

CONCLUSION

In the literature, the existing studies on SIM cards are mostly for un-keyed cards. In this paper, we provide a novel key exchange protocol between an un-keyed SIM card and a Service Provider. When *SIMSec* protocol is performed, a symmetric key is created at both sides that can be used to encrypt data. The protocol allows SIM card and Service Provider to perform secure applications on SIM cards such as mobile payment services.

REFERENCES

[1] Meeting notes with Turkcell Technology, November 2014

[2] B. Schneier, J. Kelsey, D. Whiting, D. Wagner, C. Hall, N. Ferguson, *The Twofish encryption algorithm: a 128-bit block cipher*, New York, NY: John Wiley & Sons, Inc., 1999.

[3] J. Daemen, V. Rijmen, *The design of Rijndael: AES-the advanced encryption standard*, Secaucus, NJ: Springer, 2002

[4] B. Schneier, "Description of a new variable-length key, 64-bit block cipher (Blowfish)," In *Fast Software Encryption*, R. Anderson, Ed. U.K: Springer, 1994, pp. 191-204.

[5] W. Stallings, W, "The advanced encryption standard," *Cryptologia*, vol. *26(3)*, pp. 165-188, July 2002

[6] D. Coppersmith, "The Data Encryption Standard (DES) and its strength against attacks," *IBM journal of research and development*, vol *38(3)*, pp. 243-250, May 1994

[7] Barker, William C.; Barker, E. NIST Special Publication 800-67 Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher Revision 1, 2012.

[8] K. Ok, V. Coskun, C. Cevikbas and B. Ozdenizci, "Design of a Key Exchange Protocol between SIM Card and Service Provider", *23rd Telecommunications forum TELFOR 2015*, Belgrade, SERBIA, 24-26 November 2015, pp. 281-285.

[9] *cdma2000 Packet Data Services: Wireless Local Area Network (WLAN) Interworking - Access to Internet*. 3GPP2 Standard X.S0028-100-0, 2007.

[10] *Over-the-Air Service Provisioning of Mobile Stations in Spread Spectrum Systems*, TIA Standard TIA-683-D, 2006.

[11] *Password-authenticated key exchange (PAK) protocol*, ITU Standard X.1035, 2007.