Development of Mechanism of Integrity in M-Commerce Using Joint Signature Scheme

Sobia Shafiq, Dr. Malik Sikandar Hayat Khiyal, *Member IACSIT*, and Aihab Khan

***Abstract*—This research focuses on providing integrity in mobile transactions. The main contribution for this paper is to develop mechanism which can provide integrity in mobile payment methods by using joint signature scheme, as it uses fewer resources. Different hash functions and time stamping is used for providing integrity of data. The result obtained by this research shows that integrity issue is resolved in mobile commerce domain. Message as well as time stamp is send through trusted third party from sender to receiver. Experimental results shows that different hash functions used for hashing will produce different results, in context of time, memory and throughput and it also depends upon message size.**

***Index Terms*—Integrity, joint signature scheme, mobile-commerce, mobile-payment.**

I. Introduction

As world of technology is making heaps now a day, so need for security measures in such technology is also growing rapidly because without basic measure of information security such technology will become useless. Thus to have an efficient use of information technology today we have to make it secure. The purpose of this research is to ensure integrity in m-commerce, as integrity is the one of the most sensitive issue in m-commerce. Mobile commerce is about buying and selling of goods and services by using mobile phones without any restriction of location. In mobile commerce, consumer sends a payment request via an secure message service (SMS) text message to a short code and a premium charge is applied to their phone bill or their online wallet.

Different security issues are involved during transactions through mobile phones such as confidentiality, integrity, non-repudiation, origin authentication etc. All of the security features should be handled with great care for providing a secure mobile commerce. For security we can use a lot of techniques such as, digital signatures, hash functions,

Encryption/decryption, joint signatures etc. Among the entire techniques provided joint signature scheme has great importance for providing secure transactions because it is computationally low, less expensive and uses less resources.

This research is based on providing integrity of messages in mobile commerce and this is done by using joint signature scheme. In real world it may happen that during transaction real message is not received at the receiver end, or any intruder may change the message so the transactions must be secured and confidentiality should be supported in it. The main objective of this research paper is that to truly provide integrity mechanism using joint signature for secure mobile transactions.

Section II includes the related work Section III provides the description of the proposed framework. Section IV provides the proposed technique. Section V presents experimental results and analysis with the help of graphs. Section VI gives conclusion of system and gives the future work.

II. Related Work

Now a day's mobile commerce is gaining importance and different security services are related with mobile transactions, such as non-repudiation, authentication, for secure transactions. For this purpose He and Zhong [1], introduces joint signature scheme. In which two parties signs the message. This scheme is based on hash functions and digital signature methods. Joint signatures provide security services with low computational resources and communication cost. Singh and Khan [2] presents the security of joint signature. His proposed scheme presents the security of message by hybrid encryption method and digital signatures. His proposed scheme uses International Data Encryption Algorithm-Range Scaling Algorithm (IDEA-RSA) for encrypting message and RSA digital signature is used for getting digital signatures. This scheme uses "encrypt-then-sign" instead of "sign-then-encrypt" policy. Ji-yong and Ryu [3], this paper propose secure key checksum which is a secure mechanism using encryption to provide confidentiality and integrity. Issues of confidentiality and integrity are resolved in network encoding. For providing confidentiality secure key checksum utilizes encryption and for integrity it verifies encoded block. Ayub et al. [4], proposed a technique for origin authentication of digitally signed message by using joint signature scheme in mobile commerce. The issue of origin authentication in m-commerce is resolved by this technique. Proposed technique is efficient in mobile domain because it can be used with limited resources in mobile commerce. An experimental analysis shows that proposed technique overcomes the major drawbacks of traditional digital signed message. Apvrille et al. [5], is about providing integrity for archival of organizations. For coping with the problem of providing integrity a time stamped scheme is introduced. This paper proposed triple integrity mechanism that include data integrity, time integrity, copy integrity. This scheme is used in digital signatures.

Manuscript received March 20, 2012; revised May 5, 2012.

Sobia Shafiq is with the Software Engineering in Fatima Jinnah Women University (e-mail: sobiashafiq786@gmail.com).

Malik Sikandar Hayat Khiyal is with the head of academic APCOMS, Khadim Hussain Road, Lalkurti, Rawalpindi (e-mail: m.sikandarhayat@yahoo.com).

Aihab khan is with the Department of computing, Iqra University, Islamabad (e-mail: aihabkhan@yahoo.com).

III. Proposed Solution

The proposed framework model shown in fig 1 explains that key distribution center shares key between Message Sender (MS) and Message Receiver (MR).Message Sender (MS) sends Message, along with time stamp (TS), encrypt it then compute hash of it and send it to Message Receiver (MR) through Trusted Third Party (TTP). Symbols used in this paper are shown in table I.

TABLE I: Symbols.

MS	Message sender
MR	Message receiver
TTP	Trusted third party
M	Message being sent by the message sender
K1	Shared id key between message sender trusted third party and message receiver
H	Used for hash function
||	Sign of concatenation
PRTTP	Private key of trusted third party
PUTTP	Public key of trusted third party
HOAC	Hash origin authentication code
HMAC	Hash message authentication code
EP	Encryption
DP	Decryption
E(HOAC)	Encrypted hash origin authentication code
E(M)	Encrypted message
E(HMAC)	Encrypted hash message authentication code
TSA	Time stamp authority
TS	Time stamp
PR_{TSA}	Private key of time stamp Authority
HTS	Hash on computed time stamp

Fig 1 show that in doing mobile transactions three parties are involved i.e. Message Sender (MS), Trusted Third Party (TTP) and Message Receiver (MR). Secrete keys are shared between these three parties through a key distribution center. Message sender will send the message, time stamp and hash of the messages to the trusted third party where it is signed by Trusted Third Party and send to Message receiver.

Fig 2 represents the detailed view of integrity of both message as well as time stamp will be provided in mobile commerce using joint signature scheme.

The proposed model is shown in Fig. 2. The technical description of this model is explained below.

Technical Description of Proposed Model

ALGORITHM:

In this section we explain about the detail technical description of proposed model. Initially before starting the keys are being shared between Message Sender (MS) and Message Receiver (MR) through Trusted Third Party (TTP). The steps for proposed technique are explained below:

Step 1:

Message Sender (MS) sends an encrypted message to Message Receiver (MR) and gives a joint signature with the help of Trusted Third party (TTP).

Step 2:

In addition with message send to Message Receiver (MR) by Message Sender (MS) a Time Stamp (TS) is also send to Message Receiver (MR) for providing integrity for both time as well as message.

Step 3:

Firstly as user start entering the message, timer will get start to calculate the time stamp till encryption. Then one of the hash function is applied on the message i.e. SHA1, SHA512, ShA256, MD5.

Step 4:

Then encrypted message, hashed message as well as time stamp are concatenated and send to Trusted Third Party (TTP). TTP will behave as a linking party between Message Sender (MS) and Message Receiver (MR).

Step 5:

After receiving an encrypted message from TTP, message receiver decrypts that message with the same decryption technique by which it was encrypted as well as it will be compared for providing integrity of message. In the same way time stamp will also be compared after receiving the message for providing integrity of time.

Pseudo code of proposed model is

Algorithm: Hash Functions

SHA1

Algorithm for hash function SHA1 is shown in table II.

TABLE II: Algorithm for Hash Function SHA1.

```
UnicodeEncoding UE = new UnicodeEncoding();
        byte[] hashValue;
        byte[] message = UE.GetBytes(text);
        SHA1Managed hashString = new SHA1Managed();
        string hex = "";

        hashValue = hashString.ComputeHash(message);
        foreach (byte x in hashValue)
        {
           hex += String.Format("{0:x2}", x);
        }
        return hex;
```

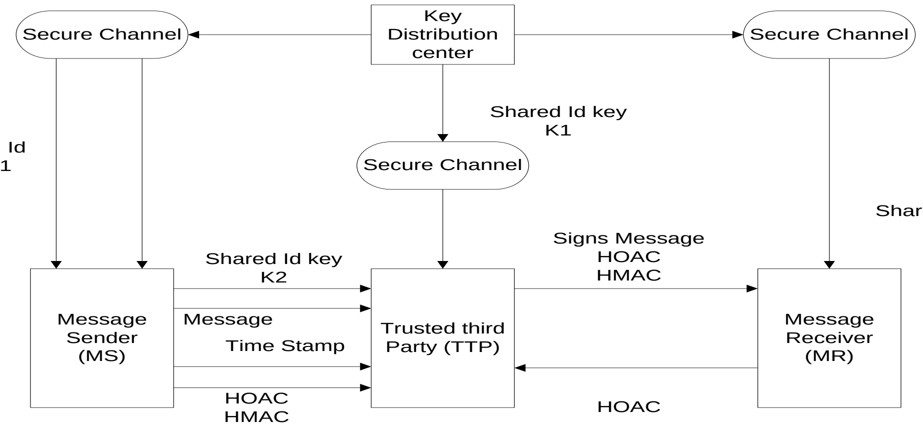

Fig. 1. Proposed framework for providing integrity in m-commerce.

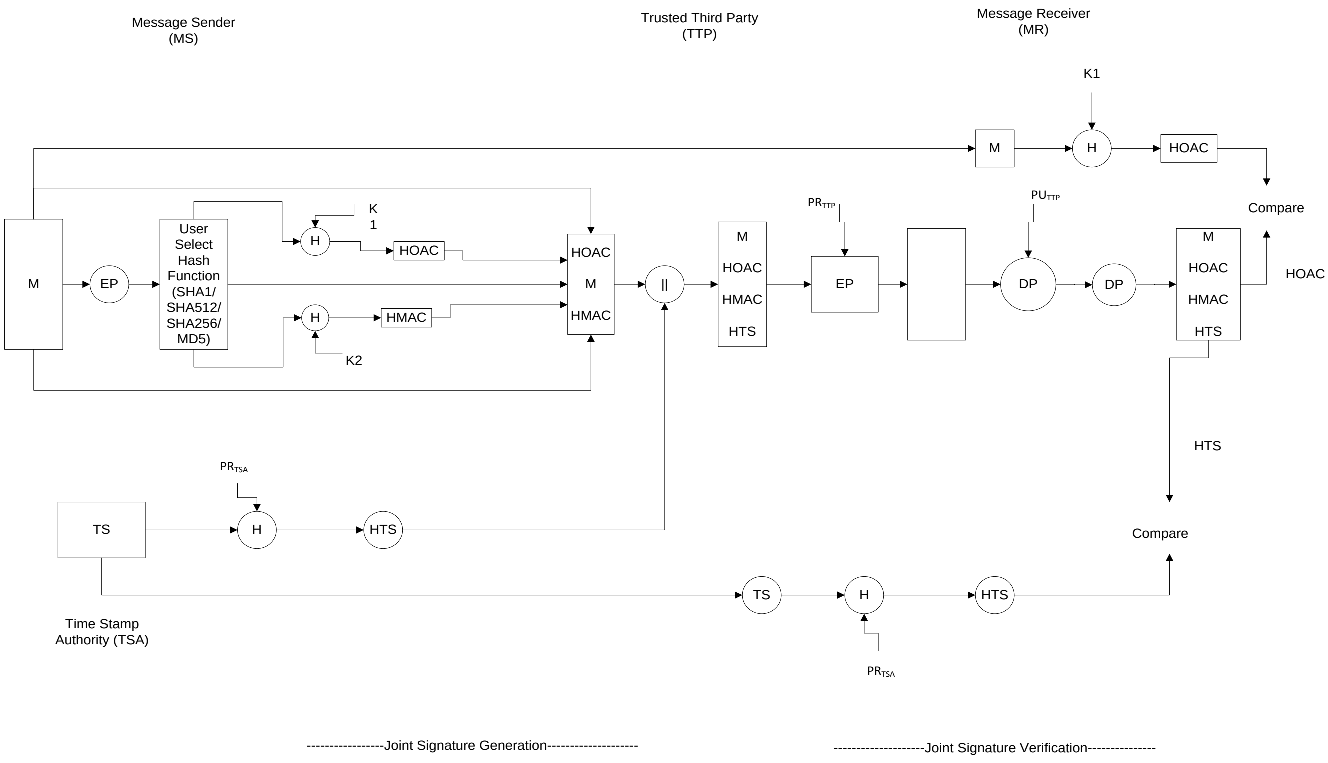
Message Sender (MS)
Trusted Third Party (TTP)
Message Receiver (MR)
K1
M
H
HOAC
Compare
K 1
M
EP
User Select Hash Function (SHA1/ SHA512/ SHA256/ MD5)
H
HOAC
H
HMAC
K2
HOAC
M
HMAC
||
M
HOAC
HMAC
HTS
PR_TTP
EP
PU_TTP
DP
DP
M
HOAC
HMAC
HTS
HOAC
HTS
PR_TSA
TS
H
HTS
Compare
TS
H
HTS
PR_TSA
Time Stamp Authority (TSA)
-----------------Joint Signature Generation--------------------
--------------------Joint Signature Verification---------------

Fig. 2. Proposed model for providing integrity of messages and time stamp in m-commerce.

SHA256

Algorithm for producing hash function SHA256 is shown in table III.

TABLE III: ALGORITHM FOR HASH FUNCTION SHA256.

```
UnicodeEncoding UE = new UnicodeEncoding();
         byte[] hashValue;
         byte[] message = UE.GetBytes(text);
         SHA256Managed hashString = new
SHA256Managed();
         string hex = "";

         hashValue = hashString.ComputeHash(message);
         foreach (byte x in hashValue)
         {
            hex += String.Format("{0:x2}", x);
         }
         return hex;
```

SHA512

Algorithm for producing hash function SHA512 is shown in table IV.

TABLE IV: ALGORITHM FOR HASH FUNCTION SHA512.

```
UnicodeEncoding UE = new UnicodeEncoding();
         byte[] hashValue;
        byte[] message = UE.GetBytes(text);
         SHA512Managed hashString = new
SHA512Managed();
         string hex = "";

         hashValue = hashString.ComputeHash(message);
         foreach (byte x in hashValue)
         {
            hex += String.Format("{0:x2}", x);
         }
         return hex;
```

MD5

Algorithm for producing hash function MD5 is shown in table V.

TABLE V: ALGORITHM FOR HASH FUNCTION MD5.

```
UnicodeEncoding UE = new UnicodeEncoding();
        byte[] hashValue;
        byte[] message = UE.GetBytes(text);
        MD5Managed hashString = new MD5Managed();
        string hex = "";

        hashValue = hashString.ComputeHash(message);
        foreach (byte x in hashValue)
        {
           hex += String.Format("{0:x2}", x);
        }
        return hex;
```

Algorithm: Time Stamp Function:

Algorithm for time stamping is shown in table VI.

TABLE VI: ALGORITHM FOR TIME STAMPING.

```
TimeSpan span = DateTime.Now.Subtract(da);
span.Milliseconds.ToString();
```

Algorithm: Encryption Function:

Algorithm for encryption is shown in table VII.

TABLE VII: ALGORITHM FOR ENCRYPTION USING AES.

```
RijndaelManaged        symmetricKey        =        new
RijndaelManaged();
```

First of all input is given to message sender, and timer got started, after input is entered encryption of entered message will be started and as encryption start, timer will get stop. Then one of the hash functions (SHA1, SHA256, SHA512, and MD5) will be applied on encrypted message. Then after this concatenation of encrypted message, hashed message, and time stamp will be occur. Then message will be send to message receiver through trusted third party. Message receiver after receiving the message does decryption of the message. And then after decryption message as well as time stamp will be compared for integrity.

IV. EXPERIMENTAL RESULTS

A. Graphical comparison of time consumed by Hash Functions:

Time consumed by different hash functions i.e. SHA256, SHA1, SHA512, MD5 are shown in fig 3. In fig 3 time is shown along y axis where as hash functions are plotted against x axis. SHA256 consume more time for hasing for input ‘shahidalaptop’, SHA1 consume more time with input shahidalaptop, SHA512 consume more time with ‘haniahpc’, where as hash function MD5 will consume more time with input ‘shahidalaptop’. So we can conclude that as message size is larger than hash functions SHA256, SHA1, and MD5 will consume more time for hashing. Least time is consumed by hash function SHA1and MD5 for input ‘sobiacar’

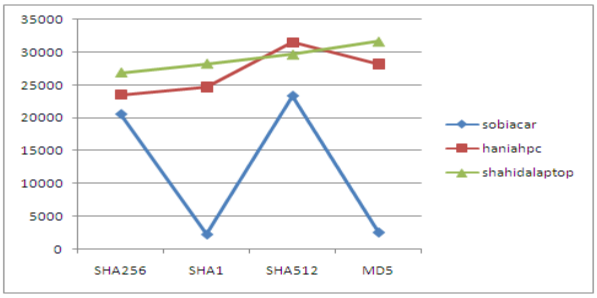

Fig. 3. Time consumed by hash functions.

B. Graphical Comparison of Memory Consumed by Hash Functions:

Memory consumed by different hash functions i.e. SHA256, SHA1, SHA512 and MD5 is shown in fig. 4 In fig. 4 memory in bites is taken along y axis where as hash functions are taken along x axis. Memory consumed by input ‘haniahpc’ is largest for hash function SHA256, SHa1, SHA512 and MD5 where least memory consumed by hash functions SHA256, SHA1, SHA512 and MD5 is for input ‘sobiacar’.

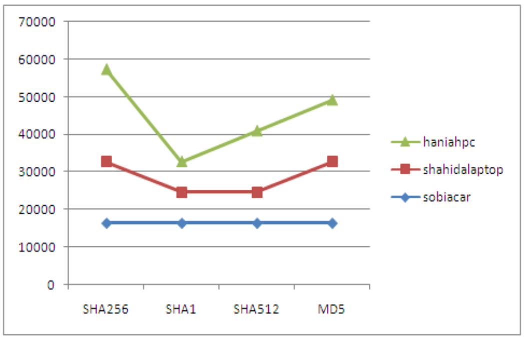

Fig. 4. Memory consumed by hash functions.

C. Graphical Comparison of Throughput Consumed by Hash Functions:

Throughput consumed by hash functions SHA256, SHA1, SHA512and MD5 is shown in fig. 5. In fig 5 throughput is taken along y axis where as hash functions are taken along x axis. Least through for hash function SHA256 is for input ‘shahidalaptop’ where least throughput for hash functions SHA1, SHA512, MD5 is for input ‘sobiacar’. Maximum throughput for all hash functions is for input ‘haniahpc’.

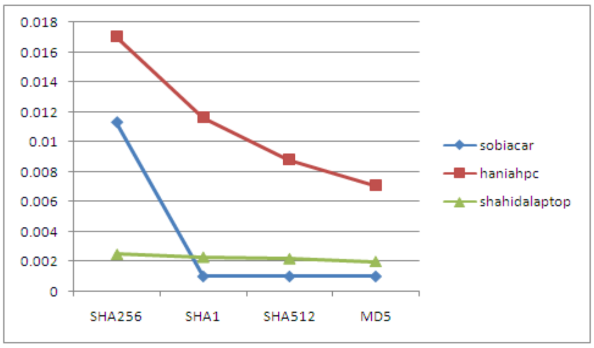

Fig. 5. Throughput consumed by hash functions.

D. Graphical Comparison of Time Consumed by AES Encryption:

Time consumed for encryption of message using AES is shown in fig 6. In fig 6 time in milliseconds is taken along y axis and inputs are taken along x axis. Time for encryption will change for each input. In our example time consumed by AES encryption method is maximum for input ‘shahidalaptop’ and minimum for input ‘haniahpc’.

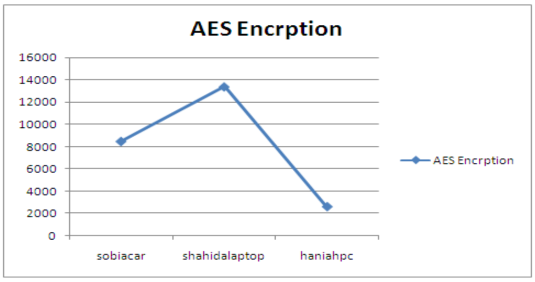

Fig. 6. Time consumed by AES encryption.

E. Graphical Comparison of Time Consumed by AES Decryption:

Time consumed for decryption of message using AES is shown in fig 7. In fig 7 time in milliseconds is taken along y axis and inputs are taken along x axis. In our example time consumed by AES decryption method is maximum for input 'sobiacar' and minimum for input 'shahidalaptop'.

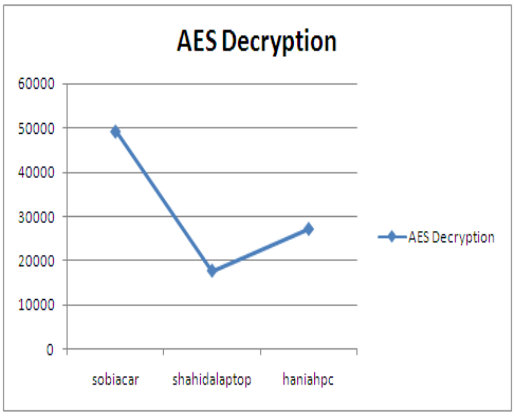

Fig. 7. Time consumed by AES decryption.

V. Conclusion and Future Work

In this paper, we present a model for providing integrity in mobile commerce. That model uses different hash functions for applying hash on the message as well as uses time stamping for providing integrity of time as well. By encrypting message and applying any one of the hash function on the message we achieve integrity of the message, and when we concatenate time stamp with encrypted message as well as hashed message we will get integrity for time also. That will make our integrity technique stronger. At the receiving end message will be compared to provide integrity, in this way no intruder can intrude that message which is sending along network. By this research we have overcome the issue of integrity in m-commerce with less resources being consumed.

In future this research work can be extended for different encryption and decryption techniques and copy integrity can also be done as future work of this research work.

References

[1] L.-S. He and N. Zhang, "A new signature scheme: joint-signature," in *Proceedings of the 2004 ACM symposium on Applied computing*, pp 807-812, March 14-17, 2004, Nicosia, Cyprus

[2] Y. P. Singh and M. A. Khan, "On the security of joint signature and hybrid encryption," *13th IEEE International Conf. on Communication*, vol. 1, pp. 4, 2005.

[3] J.-Y. Park and M.-S. Ryu "An intergarted security mechanism for network coding combining confidentiality and integrity," *ICACT'09 Proceedings of the 11th international conference on Advanced Communication Technology* vol. 1, pp 311-314, Feb 15-18, 2009

[4] S. Ayub, A. Khan, and M. S. H. Khayal "Origin Authentication of digitally signed message using joint signature scheme in mobile commerc," Int. Jour. of Latest Trends in Netw. and Comm, vol. 1, no. 1, pp. 6-12, June 2011.

[5] A. Apvrille, J. Hughes, and V. Girier, "Streamed or detached triple integrity for a time stamped secure storage system," in *proceedings of 1st Int. IEEE Security in Storage Workshop (SiSW)*, Greenbelt, Maryland, Dec 2002, IEEE Computer Society, pp. 53-64.

Biographies

Dr. M. Sikandar H. Khiyal born at Khushab, Pakistan. He is Haed of Academic APCOMS, Khadim Hussain Road, Lalkurti, Rawalpindi. He Served in Pakistan Atomic Energy Commission for 25 years and involved in different research and development program of the PAEC. He developed software of underground flow and advanced fluid dynamic techniques. He was also involved at teaching in Computer Training Centre, PAEC and International Islamic University. His area of interest is Numerical Analysis of Algorithm, Theory of Automata and Theory of Computation. He has more than hundred research publications published in National and International Journals and Conference proceedings. He has supervised three PhD and more than one hundred and thirty research projects at graduate and postgraduate level. He is member of SIAM, ACM, Informing Science Institute, IACSIT. He is associate editor of IJCTE and Co editor of the journals JATIT and International Journal of Reviews in Computing. He is reviewer of the journals, IJCSIT, JIISIT, IJCEE and CEE of Elsevier. He can be contacted at m.sikandarhayat@yahoo.com,.

Mr. Aihab Khan works in Department of computing Iqra University, Islamabad Pakistan. His research interests are in the field of Data mining, Data Warehousing as well as Information Security. He can be contacted at aihabkhan@yahoo.com.

Miss.Sobia Shafiq is a software engineer graduate from Department of Software Engineering, Fatima Jinnah Women University, Pakistan. She can be contacted at sobiashafiq786@gmail.com