© 2016 IJSRSET | Volume 2 | Issue 6 | Print ISSN: 2395-1990 | Online ISSN : 2394-4099
Themed Section: Engineering and Technology

Secure SMS Method Based on Social Networks

Saeed Yazdanpanah, Saman Shojae Chaeikar

Sama Technical and Vocational Training College, Islamic Azad University, Khorramabad Branch, Khorramabad, Iran

ABSTRACT

Nowadays, Short Message Service (SMS) communications play important role in daily lives. Authentication, reports and personal communications are the common instances of SMS applications. The message signals are being transmitted in the air without any security measures. Hence, the message content could easily be intercepted without requiring complicated infrastructure. To protect confidentiality and integrity of the messages, various techniques are devised. Some early techniques rely on extra or upgraded hardware equipments and impose high implementation cost. However, the modern techniques mostly are applicable on smart phones and work in application layer. In this research, the proposed technique utilizes the profile information and users' communications in social networks to generate two dynamic keys for SMS encryption and decryption. Cost of secure session establishment has dramatically reduced. The analysis results also show high level of user satisfaction and long secure lifetime for exchanged messages.

Keywords : SMS Security; Short Message Security; Secure Communication; Encryption; Cryptography

I. INTRODUCTION

Various techniques are devised which aim to secure SMS communications in Global System for Mobile communications (GSM) networks. The first category targets changing GSM structure to achieve security. Hossain et al. [1] offers restructuring GSM protocol in transport layer to secure the communications between the mobile equipment and the connected GSM station. If applied, the offered technique would bring security into infrastructure level and accordingly strengthen security of SMS communications. However, due to large required investment and implementation difficulties the proposed technique is not applicable.

The second category concentrates on implementation of security features in application layer. Due to increasing computational power of mobile equipments, this category has become more applicable. The resultant frameworks could be classified according location of storing cryptographic keys and implementation of security schemes. A solution is to store the application and its cryptographic keys in programmable SIM cards. This is used in the techniques introduced by IPCS Group [2] and Rongyu et al. [3]. Programmable SIM cards have advantages like tamper resistance SIMs and transferability between mobile equipments with no data loss. However, there are disadvantages like limited computational powder which constrains implementation of advanced security schemes. An alternative to programmable SIM cards is using SIM cards as cryptographic key storage and the mobile equipment as the processing unit [4,5]. Although implementing this technique is easy, but there are many difficulties when communicating with SIM cards from other operators.

The counted difficulties could be overcome when bringing security into application level. MIABO [6], SafeSMS [7] and the techniques in [8] and [9] use pre-shared passphrase or public key cryptography to implement security in application layer. Since these techniques are applicable on Person-to-Person communications only, their application is restricted. Additionally, security relies on safety of key distribution. Another solution in application layer is Client-Server model in which the engaged entities could be human or authorities like security agencies or CAs. Examples of these techniques are SMSec [10] and Trusted-SMS [11].

II. METHODS AND MATERIAL

2.1 Technical Background

2.1.1. SMS Architecture

In process of delivering a short message in GSM, it would be exchanged between a Mobile Station (MS) and a Base Transceiver Station (BTS). Messages are being routed from a Message Service Center (MSC) to a Short Message Service Center (SMSC). The message would be stored in SMSC until is delivered or its validity time is elapsed. Its recipient could either be MS or SMS gateway. Gateways are the servers which are in connection with one or more SMSCs for generating SMS application for users. The applications could be icon delivery, ring tone, bank services, entertainment or such services [12].

Consumer technologies always are dealing with security issues. The major SMS security issues are:

- Insecure nature of SMS
- Focusing on people rather than technology for breaching security

SMS content is open to the mobile network operators. Therefore, SMS is not a convenient channel for secure communications [13]. It might be hard to hack into telecom provider systems to access SMS communications, but it is easily possible to sniff messages and intercept them. Alternatively, privileged staffs in telecom companies are an excellent source for accessing exchanged messages [14]. Deficiencies of SMS protocols generate huge technical security gaps which might be exploited by attackers. The main SMS protocol vulnerabilities are:

- SMS interception: in wired networks, over the air
- Snooping: forwarding network elements at the store, on device
- Spoofing: own SMS gateway, in application of commercial tools
- GSM attacks, the SMS protocols: the weakest security link is mobile phone. Leaving the mobile phones unattended might result in snooping of confidential and private messages

Maximum load of exchanged SMS data equals with 140 octet or in other words 1120 bits (14 octets=140*8bits). Length of the messages would vary based on the applied character set. With a 7-bits character set the message length could be up to 160 characters. This value would be reduced to 140 characters when using 8-bits characters and to 70 characters for 16-bits characters. Long messages could be split into several messages but a portion of the space in each SMS would be occupied for message reassembling data. The messages might be delivered in different order. The overhead data in each message defines messages reassembling order.

2.1.2. SMS Security

To secure a system, there are principal security rules to be met. Depend on application, observing all or some of the conditions might be compulsory. The principals of data security are [15,16]:

- Confidentiality: keeping unauthorized parties away from accessing private information. Interception is a confidentiality attack instance.
- Integrity: prevention of unauthorized parties from data manipulation. Examples of integrity attacks are recording, replaying and modification.
- Availability: providing data access for authorized parties when needed. Instances of availability attacks are denial of service and inception.
- Authenticity: prevention of message content manipulation by unauthorized parties. Fabrication is an example of authenticity attack.

Depend on the applied method, there are two types of passive and active attacks [17,18]. Passive attack is monitoring or eavesdropping of transmitted messages. Intruders' aim is to acquire the transmitted information for traffic or content analysis. Since passive attacks do not alter data, they are not easily detectable. Passive attack is to modify or create the data stream. This could be classified into denial of service, replay, masquerade and message modification [19,20]. Unlike passive attacks, there are countermeasures against active attacks. However, they are difficult to prevent as physical protection for communication equipments and channels is required.

2.1.3. Hash Functions

A hash function turns an arbitrary length text into a fixed length text called hash, hash value, hash code or hash sum. A cryptographic hash function is a one way function which converts a text into a hash value while the value is not reversible to the original text. In

cryptographic hash functions the input value is called message and output value is called message digest or digest. Below are the main principal properties of a hash function [21,22]:

- Computing the hash is easy
- The hashing process is not reversible
- Changing the message without affecting the hash value is impossible
- There are no two different messages with identical hash value

Hash functions have wide application in information security like digital signature and Message Authentication Code (MAC) [23,24]. Table 1 below presents name and technical specifications of important hash functions.

Table 1 : Name, output size and internal states of the important hash functions

Function	Output size (bit)	Internal size	Block size	Length	Word size	Rounds
GOST	256	256	256	256	32	256
HAVAL	256/224/192/160/128	256	1024	64	32	160/128/96
MD4	128	128	512	64	32	48
MD5	128	128	512	64	32	64
SHA-0	160	160	512	64	32	80
SHA-1	160	160	512	64	32	80
SHA-224,SHA-256	256/224	256	512	64	32	64
SHA-384, SHA-512, SHA-512/224, SHA-512/256	384/512/224/256	512	1024	128	64	80
SHA-3	224/256/384/512	1600	1600-2*bits	-	64	24
SHA3-224	224	1600	1152	-	64	24
SHA3-256	256	1600	1088	-	64	24
SHA3-384	384	1600	832	-	64	24
SHA3-512	512	1600	576	-	64	24
Tiger(2)-192/160/128	192/160/128	192	512	64	64	24

2.2. Secure SMS Based on Social networks (SSS)

This research proposes a new technique which is structured based on user profile and exchanged text messages between social network users for securing their SMS communications. Hash of the user profile information and exchanged messages constructs encryption and decryption keys. Each user utilizes two separate dynamic keys for encryption and decryption processes. SSS is GSM operator independent and requires up to three SMS exchanges for establishing a secure connection.

Each time SSS application gains access to the internet, it would synchronize its database with the latest updates of user profiles and exchanged messages. Up to 1KB of user profile data and the stored exchanged text messages would be utilized in key generation process. If no data exists, SSS generates and shares a random text to set the grounds of key agreement. The process and data flow for key generation is illustrated in Figure 1 below.

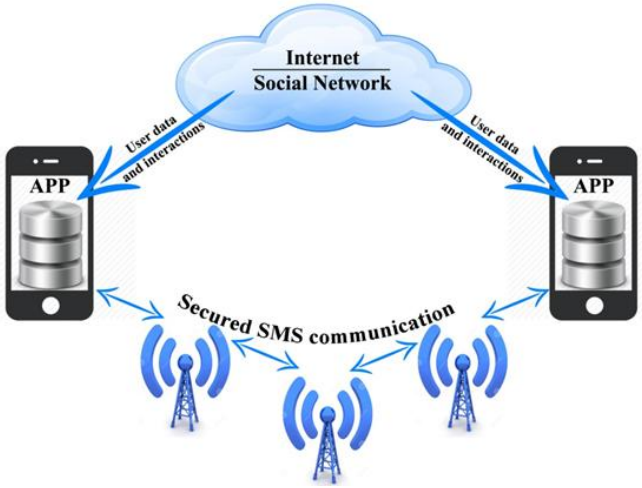

Figure 1: SSS data flow for key generation

2.2.1. Encryption and Decryption in SSS

The required cryptographic hash function for SSS should meet two criteria. Firstly, it should be secure enough to protect confidentiality of the encrypted messages. And secondly, length of the generated key should be proportion of the SMS length. In case of incompatibility between these, the process might face two difficulties. The message size either should be reduced to the key length or the biggest proportion of the key smaller than 1120 bits, or if the chosen key is longer than 1120 bits, the encrypted message should be split in multiple messages which results in increasing cost and implementation difficulties.

Among the studied hash functions in Table 1, the five functions of RIPEMD, HAVAL, SHA-1, SHA-0 and Tiger (2) have the hash length of 160 bits. Dividing SMS length (1120 bits) into hash length (160 bits) of the counted functions reveals that the key should be utilized up to 7 times to encrypt/decrypt the message. All the five counted hash functions meet the criteria to be utilized in SMS encryption. Between these, SHA-0 and SHA-1 are designed by National Security Agency (NSA) of the United States and are CRYPTREC and FIPS PUB 180-4 certified [25]. Since SHA-1 is newer, is the chosen hash function for encrypting SMS messages in SSS.

2.2.2. Social Networks in SSS

SSS records the user profile information and the message communications between users within the selected social networks to be utilized as key generation factors. The keys getting updated based on the latest exchanged data between the users on the chosen social network. Additionally, involving date and time parameters in key generation process enhances dynamicity of the keys and prevents replay attack.

2.2.3. Key Generation Factors and Process

There are four factors involved in key generation process. The first one is user profile data. Fields of this factor varies based on the chosen social network. The second factor is the users' phone number. SSS generates unique encryption/decryption hash keys for each phone number. This is the key generation factor which generates the uniqueness of the keys for the used phone line. Combination of time and date is the third factor involved in hash key generation. This combination is the dynamic item which changes every hour and bans the replay attacks. The fourth factor is exchanged messages between the users on the chosen social network. The latest 1KB will be utilized in key generation.

Each user generates a couple of keys. The first key uses sender's information and the second key uses receiver's information. At transmission time, sender encrypts the message using recipient's key. Recipient also decrypts the message using its own key. Encryption key of user A (E_{KA}) is the same as decryption key of user B. This also applies on decryption key of user A (D_{KA}) and encryption key of user B (E_{KB}). Table 2 below describes the notations used in key generation process.

Table 2: Descriptions of notations in SSS keys

Notation	Description
	Encryption key of user A
	Encryption key of user B
	Decryption key of user A
	Decryption key of user B

Below describes the process and factors involved in key generation process.

$$E_{KA} = SHA-1\,(user\ profile\ B + phone\ number\ B + unified\ date\ \&\ hour + selected\ exchanged\ messages)$$

$$E_{KB} = SHA-1\,(user\ profile\ A + phone\ number\ A + unified\ date\ \&\ hour + selected\ exchanged\ messages)$$

$$D_{KA} = SHA-1\,(user\ profile\ A + phone\ number\ A + unified\ date\ \&\ hour + selected\ exchanged\ messages)$$

$$D_{KB} = SHA-1\,(user\ profile\ B + phone\ number\ B + unified\ date\ \&\ hour + selected\ exchanged\ messages)$$

Since hour is a factor in key generation process, maximum lifetime of the generated keys is 60 minutes. Therefore, if a message is encrypted in last valid seconds of current key and get delivered in lifetime of the next key, it would not be decrypt-able. This might also happen due to operator delay in message delivery or unavailability of the recipient. To overcome this issue, the previous key also would be applied in decryption process, if the message is not decrypt-able with current key. If the message was not yet decrypt-able, the message would be recognized as invalid.

SSS communicating users might be located in two different time zones. This will result in generating incompatible keys. To prevent it, UTC time and Gregorian calendar are chosen as references. Local time and date firstly would be converted into the reference format and then be applied within the process.

2.2.4. Key Agreement

Key agreement in SSS might face three conditions. First condition is when storages of both mobile devices are synchronized and generate identical keys without overhead message exchange. Since this is unknown if

both parties are using synchronized databases, user A sends an encrypted message to user B based on latest update of its database. Decrypting the message by user B shows that the databases are synchronized. In this condition no key agreement message is exchanged. In Figure 2 below these steps are labeled as 1 and 2.

Second condition is when the received message is not decrypt-able. To establish a secure session, user B sends its latest database update time and date to user A. If the data of the received date and time existed in the database, user A will encrypt the messages using the received data from user B. In this case, cost of key agreement is two exchanged messages. These key agreement steps are labeled as 3, 4 and 5 in Figure 2 below.

Third condition happens when database of user B is fresher than user A. To resolve this issue, user A sends its latest update date and time to user B. Afterward, without any extra message exchange, the key agreement is performed and the users will be able to communicate securely. Steps of key agreement in third condition are labeled as 6, 7 and 8 in Figure 2 below.

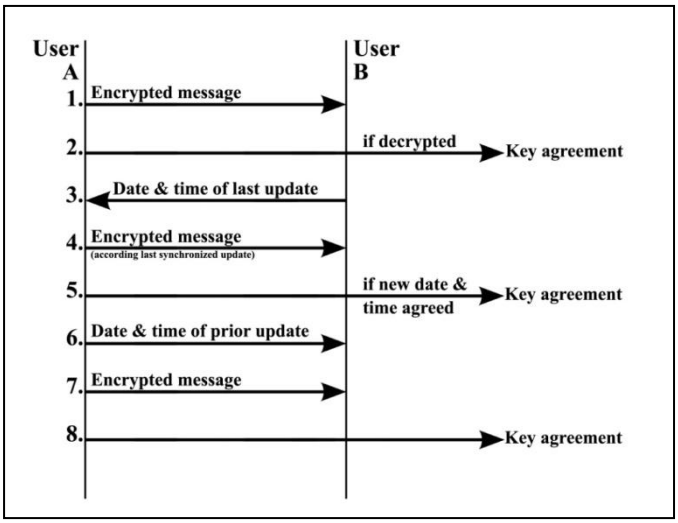

Figure 2: Key agreement steps in SSS

Upon connecting to the internet, SSS updates its database to be synchronized with other users. Therefore, establishing secure connection without internet access in situations like communicating with a new user, lack of records of exchanged data or technical failures is impossible. Exchanging a randomly generated message between the users prepares the grounds for establishing a secure connection.

2.2.5. Session Establishment and Termination

A secure session always starts with a message from user A to B. The key agreement process starts upon exchanging the first message and users can establish a secure channel afterward. Although the session establishment starts after first message, but there is no sign for last message. Session termination automatically happens one hour after last exchanged message. In other words, session termination happens between 60 to 119 minutes after last message exchange. Importance of session termination is for next session establishment. Before terminating a session, the exchanged messages between the users or profile updates does not affect the key generation process.

III. RESULTS AND DISCUSSION

SSS generates the encryption and decryption keys according user profiles and exchanged messages in the chosen social network. The process is dynamic and mobile operator independent. The main advantages of SSS are:

- Reducing cost of key agreement process to up to 3 messages
- Utilizing dynamic fresh keys for encryption/decryption
- Utilizing separate keys for encryption and decryption
- Replay attack resistant
- Using phone number dependent keys
- Mobile operator independent
- CA independent

SSS is evaluated in three approaches of key security lifetime, cost effectiveness and user satisfaction. For security lifetime, minimum secure life of the generated keys is calculated. In economical approach, SSS is compared with cost of establishing a Diffie-Hellman secure channel. For user satisfaction evaluation, the results are calculated according the 50 questionnaires distributed between the users which experienced working with the SSS developed prototype.

3.1. Security Lifetime

SSS uses symmetric keys for encryption and decryption. The research by Lenstra and Verheul [26] shows that symmetric keys with length of 109 bits are secure beyond 2050. SSS uses 160-bits length

encryption/decryption keys. Therefore, safety of the keys is guaranteed beyond 2050.

3.2. Economic Evaluation

SSS does not rely on supporting infrastructures like certificate authorities (CAs) or mobile operators. Therefore, in economical approach SSS is compared with Diffie-Hellman which generates a secure channel through exchanging session establishment messages. For this comparison, number of exchanged messages for establishing a secure channel is compared.

In Diffie-Hellman, through a mathematical process, a couple of asymmetric keys secure the communication. To establish a Diffie-Hellman secure channel, 7 messages are needed to be exchanged [27]. In SSS, exchanging between 0 to 3 messages is needed. Table 3 below compares SSS and Diffie-Hellman in term of imposed message exchange cost for establishing a secure channel. The analysis results in Table 3 reveal that SSS works 57.14% to 100% more economical than Diffie-Hellman.

Table 3: Costing comparison between SSS and Diffie-Hellman for establishing a secure channel

SSS key agreement round	Number of SMSs in SSS	Number of SMSs in Diffie-Hellman	SSS Cost Improvement
SSS - condition 1	0	7	%100
SSS – condition 2	2		%71.43
SSS – condition 3	3		%57.14

3.3. SSS User Satisfaction Evaluation

To evaluate SSS user satisfaction, a prototype developed and distributed between 50 users. The users were randomly selected from university students. Upon experiencing working with SSS prototype, they requested to answer a questionnaire. The questionnaire analysis results are presented in Table 4 below.

The questionnaire is constituted from 4 questions which study users' internet usage, membership in social networks, using smart phones and satisfaction of users in using SSS. Due to pre-requirements of using SSS, in satisfaction evaluation only the questionnaires are used which have positive responses for the first three questions in Table 4. Since the users with negative answers in first three questions had not enough qualification to judge the product, their questionnaires are ignored in final evaluation.

Table 4: SSS user satisfaction questionnaire

No.	Question
1	Are you member of a social network?
2	Are you using smart phone?
3	If you are using a smart phone, how often you use internet?(daily, weekly, monthly, I do not use)
4	Are you willing to use SSS for securing your text message communications?

Table 5: Statistical analysis of SSS evaluation questionnaires

Question	Quantity	Total users	Percentage
No use of social networks	5	50	10%
No use of smart phones	6	50	12%
No use of internet	2	50	4%
Number of questionnaires with negative answers for questions 1, 2 or 3	7	50	14%
Number of questionnaires with positive responses for questions 1, 2 and 3	43	50	86%
Number of questionnaires with positive responses for questions 1, 2 and 3 and positive evaluation of SSS	39	43	90.69%

The statistics in Table 5 shows that 90.69% of the users which meet the pre-requirements evaluated SSS as secure software for protecting their text message communications. However, 9.31% of the qualified users were not interested in using SSS. Percentage of the users which do not meet SSS pre-requirements is 14%.

IV. CONCLUSION

This research proposes a new technique for securing text message communications (SMS). The designed and developed technique has been evaluated in three approaches of key security lifetime, cost effectiveness and user satisfaction. In term of secure lifetime, the used keys are guaranteed to remain safe beyond 2050. In cost effectiveness evaluation, depend on the conditions, the proposed technique is 57.14% to 100% more economic than similar practices. In applicability and user satisfaction approach, 90.69% of qualified users evaluated SSS positively.

V. REFERENCES

[1] A. Hossain, S. Jahan, M. Hussain, M. Amin, and S. Shah Newaz, "A proposal for enhancing the security system of short message service in GSM," in Anti-counterfeiting, Security and Identification, 2008. ASID 2008. 2nd International Conference on, Aug. 2008, pp. 235–240.

[2] IPCS Group "IPCryptSim SMS Encryption", http://www.ipcslive.com/pdf/IPCSSMS.pdf, online visited July 2009.

[3] H. Rongyu, Z. Guolei, C. Chaowen, X. Hui, Q. Xi, and Q. Zheng, "A PK-SIM card based end-to-end security framework for SMS," Computer Standards & Interfaces, vol. 31, no. 4, pp. 629–641, 2009.

[4] M. Toorani and A. Beheshti Shirazi, "SSMS - A secure SMS messaging protocol for the m-payment systems," in Computers and Communications, 2008. ISCC 2008. IEEE Symposium on, July 2008, pp. 700–705.

[5] M. Hassinen, K. Hypp¨onen, and K. Haataja, "An Open, PKI-Based Mobile Payment System," in ETRICS, 2006, pp. 86–100.

[6] U. Chirico, http://www.ugosweb.com/miabo/, online visited July 2009.

[7] M. Hassinen, "SafeSMS - end-to-end encryption for SMS," in Telecommunications, 2005. ConTEL 2005. Proceedings of the 8th International Conference on, vol. 2, 15-17, 2005, pp. 359–365.

[8] D. Lisonek and M. Drahansky, "SMS Encryption for Mobile Communication," in Security Technology, 2008. SECTECH '08. International Conference on, Dec. 2008, pp. 198–201.

[9] M. Hassinen and S. Markovski, "Secure SMS messaging using Quasigroup encryption and Java SMS API," in SPLST, 2003, pp. 187–200.

[10] J. L.-C. Lo, J. Bishop, and J. Eloff, "SMSSec: An end-to-end protocol for secure SMS," Computers & Security, vol. 27, no. 5-6, pp. 154–167, 2008.

[11] A. Grillo, A. Lentini, G. Me, and G. Italiano, "Transaction Oriented Text Messaging with Trusted-SMS," in Computer Security Applications Conference, 2008. ACSAC 2008. Annual, Dec. 2008, pp. 485–494.

[12] Saeed Yazdanpanah, Saman Shojae Chaeikar, Mazdak Zamani, Reza Kourdi, "security features comparison of master key and IKM cryptographic key management for researchers and developers," International Conference on Software Technology and Engineering, 3rd (ICSTE 2011), 2013, Kuala Lumpur, Malaysia.

[13] Mojtaba Alizadeh, Wan Haslina Hassan, Mazdak Zamani, Touraj Khodadadi, Saman Shojae Chaeikar, "A Prospective Study of Mobile Cloud Computing," International Journal of Advancements in Computing Technology(IJACT), Vol. 5, No. 11, pp. 198 ~ 210, 2013. Publisher: IACIT. Journal URL: http://www.aicit.org/ijact/home/index.html , Paper URL: http://www.aicit.org/IJACT/ppl/IJACT3136PPL.pdf

[14] Saman Shojae Chaeikar, Aizah Bt Abdul Manaf, Mazdak Zamani, "Comparative analysis of Master-key and Interpretative Key Management (IKM) frameworks," Cryptography and security in computing, ISBN: 978-953-51-0197-6, Publisher: Intech. March 2012, pp. 302-218, New York, USA.

[15] N. Saxena and A. Payal, "Enhancing Security System of Short Message Service for M-Commerce in GSM," International Journal of Computer Science & Engineering Technology (IJCSET), ISSN: 2229-3345 vol. 2, no. 4, April 2011, pp. 126-133.

[16] Saman Shojae Chaeikar, Mazdak Zamani, Christian Sunday Chukwuekezie, Mojtaba Alizadeh, "Electronic Voting Systems for European Union Countries," JNIT: Journal of Next Generation Information Technology, Vol. 4, No. 5, pp. 16 ~ 26, 2013. Publisher: IACIT.

[17] W. Stallings, "Cryptography and Network Security," 4th Ed., Prentice Hall, 2005, pp. 58-309.
[18] A. Nadeem, M. Y. Javed, "A Performance Comparison of Data Encryption Algorithms," First IEEE International Conference on Information and Communication Technologies, 2005, pp. 84- 89.
[19] Saman Shojae Chaeikar, Shukor Abd Razak, Shohreh Honarbakhsh, Hossein Rouhani Zeidanloo, Mazdak Zamani, Farhang Jaryani, "Interpretative Key Management (IKM), A Novel Framework," iccrd, pp.265-269, 2010 Second International Conference on Computer Research and Development, 2010, Kuala Lumpur, Malaysia.
[20] Saman Shojae Chaeikar, Mohammadreza Jafari, Hamed Taherdoost, Nakisa Shojae Chaei Kar, "Definitions and Criteria of CIA Security Triangle in Electronic Voting System," International Journal of Advanced Computer Science and Information Technology (IJACSIT). Vol. 1, No.1, October 2012, Page: 14-24, ISSN: 2296-1739, © Helvetic Editions LTD, Switzerland.
[21] Hamed Taherdoost, Shamsul Sahibuddin, Meysam Namayandeh, Neda Jalaliyoon, Alaeddin Kalantari, Saman Shojae Chaeikar, "Smart Card Adoption Model: Social and Ethical Perspectives," International Journal of Research and Reviews in Computer Science (IJRRCS). Vol. 3, No. 4, August 2012, ISSN: 2079-2557. Pages 1792-1796. © Science Academy Publisher, United Kingdom.
[22] Mojtaba Alizadeh, Saeid Abolfazli, Mazdak Zamani, Sabariah Baharun, Koichi Sakurai, "Authentication in Mobile Cloud Computing: A Survey," Journal of Network and Computer Applications, Volume 61, 2016, PP 59-80.
[23] Mojtaba Alizadeh, Mazdak Zamani, Sabariah Baharun, Azizah binti Abdul Manaf, Koichi Sakurai, Hiroki Anada, Shehzad Ashraf Chaudhry, Muhammad Khurram Khan, "Cryptanalysis and Improvement of "A Secure Password Authentication Mechanism for Seamless Handover in Proxy Mobile IPv6 Networks"," Plos One, Volume 10, Issue 11, 2015, PP 1-22.
[24] Mojtaba Alizadeh, Sabariah Baharun, Mazdak Zamani, Touraj Khodadadi, Mahdi Darvishi, Somayyeh Gholizadeh, "Anonymity and Untraceability Assessment of Authentication Protocols in Proxy Mobile IPv6 Survey," Jurnal Teknologi, Volume 72, Issue 5, Pages 31-34.
[25] Mojtaba Alizadeh, Mazdak Zamani, Sabariah Baharun, Wan Haslina Hassan, Touraj Khodadadi, "Security and Privacy Criteria to Evaluate Authentication Mechanisms in Proxy Mobile IPv6," Jurnal Teknologi, Volume 72, Issue 5, 2015, Pages 27-30.
[26] Arjen K. Lenstra, Eric R. Verheul, Selecting Cryptographic Key Sizes, Journal of Cryptography
[27] Saeed Yazdanpanah, Saman Shojae Chaeikar, "IKM-based Security Usability Enhancement Model," IRACST - International Journal of Computer Science and Information Technology & Security (IJCSITS), ISSN: 2249-9555, Vol. 2, No.4, August 2012. Pages 852-858. International Research Association of Computer Science & Technology, India.