International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2016): 79.57 | Impact Factor (2017): 7.296

A Secure End-To-End Protocol for Secure Transmission of SMS

Vaidehi V. Mantri[1], M. Y. Joshi[2]

[1, 2]SRTMUN University, MGM's College of Engineering, Nanded

Abstract: ***Nowadays, short message service (SMS) is being used in many daily life applications, including healthcare monitoring, mobile banking, mobile commerce etc. Short message service (SMS) is the text communication service component of mobile communication system by using standardized communication that allow to exchange of short text messages between mobile phone devices. Short message service (SMS) plays a vital role in various different fields such as in the medical field, mobile banking etc But the traditional SMS service offered by some of the network operator does not provide the encryption to the information before the SMS has been transmitted.The objective of this dissertation work is to implement an EasySMS protocol which is an efficient and secure protocol, which provides an end-to-end secure communication through SMS between end users.The main components in the Easy SMS protocol are two mobile station,authentication server (AS) which stores all the symmetric keys shared between authentication server (AS) and the respective mobile station (MS),Registration authority which stores all the information related to the mobile subscriber.There are two different phases in the first phase there is only the authentication in the second phase there is actual message exchanged between the two mobile phones within the expiry time.Easy SMS protocol is divided in two different scenarios .First scenario both the mobile station belongs to the same authentication server (AS).Second scenario both the mobile station belongs to the different authentication server (AS).Various encryption algorithms such as AES, Triple DES also hash function that is MD5 algorithm is used which provides data authentication to the short message service (SMS).Easy SMS protocol prevent from various attack which includes SMS disclosure, Over the air modification, Replay attack, Man-in-the–middle attack and impersonation attack.***

Keywords: Authentication, Security, SMS, Symmetric key, Over-the-air

1. Introduction

Short Message Service (SMS) is one of the fastest and easy communication channels to transmit the information across the world. According to the history of the Short Message Service (SMS) on the December 3, 2013 short message service (SMS) has successfully completed its 21 years of success. It was found that on the December 3,1992 the first short message service (SMS) has been sent by the Vodafone network the person who sent the first short message service(SMS) was the Neil Papworth from the UK .The short message service SMS are used in our day to day life as a communication medium such as in the Transportation Information System [3], MobileDeck [4], SMSAssassin [5], SMS-based web search such as SMSFind [6], Monitoring Community Health Worker Performance [7], private health facilities using SMS [8], participation in elections through SMS [9], in Crime Scene Investigation and many more.

Sometimes, we send the confidential information like password, pass code, banking details and private identity to our friends, family members and service providers through an SMS. But the traditional SMS service offered by various mobile operators surprisingly does not provide information security of the message being sent over the network. In order to protect such confidential information, it is strongly required to provide end-to-end secure communication between end users. SMS messages are transmitted as plaintext between mobile user (MS) and the SMS center (SMSC), using wireless network. SMS contents are stored in the systems of network operators and can be read by their personnel.

The EasySMS protocol prevents the SMS information from various attacks including SMS disclosure, over the air (OTA) modification, replay attack, man-in-the-middle attack, and impersonation attack. This EasySMS sends lesser number of transmitted bits, generates less computation. The above requirements can be accomplished by proposing a protocol called EasySMS which provides end-to-end security during the transmission of SMS over the network.

2. Literature Survey

Previously, various authors have proposed different techniques to provide security to the transmitted messages .An implementation of a public key cryptosystem for SMS in a mobile phone network has been presented in [10] but, the security analysis of the protocol has not discussed. A secure SMS is considered to provide mobile commerce services in [11] and is based on public key infrastructure. A framework Secure Extensible and Efficient SMS (SEESMS) is presented in [12], which allows two peers to exchange encrypted communication between peers by using public key cryptography. Another new application layer framework called SSMS is introduced in [13] to efficiently embed the desired security attributes in SMS to be used as a secure bearer for m-payment systems and solution is based on the elliptic curve-based public key that uses public keys for the secret key establishment. An efficient framework for automated acquisition and storage of medical data using the SMS based infrastructure is presented in [14] and the results conclude that the proposed SMS based framework provides a low-bandwidth, reliable, efficient and cost effective solution for medical data acquisition. The [11] and [13] generate shared key for each session but also generate huge overheads and not suitable for the real world applications.

In all [10]–[14], it is not clear whether the proposed approaches are able to prevent SMS against various attacks. All the above mentioned approaches/protocols/frameworks generate a large overhead as they propose an additional framework for the security of SMS. Due to physical limitations of the mobile phones, it is recommended to develop a protocol which would make minimum use of computing resources and would provide better security. However, implementation of framework always increases the overall overhead which is not much suitable for the resource constraints devices such as mobile phones. we compared our proposed protocol with the existing Easy SMS and PK-SIM protocols. The reason for chosen these protocols for comparison is that these are the only existing protocols which do not propose to change the existing architecture of cellular networks. They wanted to compare our proposed protocol with some existing protocols devoted to provide end-to-end SMS security with symmetric key cryptography, but there is no such protocol exists. Both protocols are having two phases similar to the proposed protocol and are based on symmetric as well as asymmetric key cryptography while the proposed protocol is completely based on symmetric key cryptography.

The Easy SMS protocol can be used to secure an SMS communication sent by Java's Wireless Messaging API while the PK-SIM protocol proposes a standard SIM card with additional PKI functionality. Both protocols are based on client-server paradigm, i.e., one side is mobile user and the other side is authentication server but they do not present any scenario where an SMS is sent from one mobile user to another mobile user. The Easy SMS protocol does not illustrate the security analysis.

2.1 Problem Definition

Short message service (SMS) is being used in our day to day life.There are different uses of short message service (SMS) for example to send the banking details, license number, pass code etc. But some of the mobile operators does not provide information security they have poor network security whenever the short message service (SMS) is being sent over the network. In order to overcome the security problem the efficient and the secure protocol named Easy SMS is been used which provides an end to end secure transmission of SMS. By using the different cryptographic algorithm we can prevent the different types of attack.

3. Security Goals and Proposed Solution

3.1 Attack Model

In the attack model we will discuss the different scenario of the different attack along with the different possibility where the malicious mobile station (MS) will be able to know the authentication information it will also mislead to the legitimate mobile station (MS) .As we know that the short message service (SMS) is sent as plain text to the short message service center (SMSC) so there are the chances that the network operator can access this easily during the transmission of the short message service (SMS) message. The over the air interface between the mobile station (MS) and the BTS is been protected by the very weak encryption algorithm (such as A5/1 or A5/2), so that the attacker can also get the algorithm information easily which is stored in the short message service (SMS) message or it can also send by making the changes in the short message service (SMS) information easily. The attacker can also try to generate the keys which are been used in the authentication protocol. The attacker can also try to delay the conversation which is taking place between the mobile station (MS1) and (MS2) and also it will be able to capture the important information which is been send when the protocol is been executed which is sent in the message sent before with the attack form the attack is the replay attack. For getting the authentication it will send the information known to the server or it can also modify the information which is send in the sequence to get the authentication. There is a attack which is the man-in-the-middle attack which is also possible when an mobile station (MS) is connected to the BTS with the help of the wireless network used. It will stop the session which is generated by the authenticated mobile station (MS).With the victim the attacker will establish its own new connection with both of the victims. On the active connection it will eavesdrop the session. It will modify and also intercepts the messages. The intruder can also inject the false information it can also intercept the transmitted message which is send between both of the mobile station (MS).

In such circumstances where the communication is taking place between the weak encryption algorithm or in the unexpected format. Whenever the attacker gets the secret key then all the above discussed is possible or some clue about the secret key. The key exchange phase when executes then there is the possibility of these type of attack it will try to know the session key too. Over the network if the proper integrity is not maintained then there are the chances of the impersonation of the mobile station (MS) and the authentication server (AS). The intruder can pretend like a legitimate MS and ask to the AS for valid authentication tokens in order to make the AS believe that originate from the authentic MS. Similarly, he/she can also show him (her) self like a valid AS and ask legitimate MS to send the information in order to make the target MS believe that originate from a genuine AS.

3.2 Proposed protocol

A new protocol named EasySMS protocol with two different scenarios which provide end-to-end secure transmission of information in the cellular networks. First scenario is illustrated in Fig. 3.1 where both MS belong to the same AS, in other words share the same Home Location Register (HLR) while the second scenario is presented in Fig. 3.2 where both MS belong to different AS, in other words both are in different HLR. There are two main entities in the EasySMS protocol. First is the Authentication Server (AS), works as Authentication Center (AuC) and stores all the symmetric keys shared between AS and the respective MS. In this paper, we refer AuC as the AS. Second entity is the Registration Authority (RA) which stores all the information related to the mobile subscribers.
The abbreviations and symbols are:

1. **MS:** Mobile station referring user
2. **AS:** Authentication server referring AuC
3. **RA:** Registration authority
4. **IDMS:** International mobile subscriber identity of MS(128-bits)
5. **Q:** New session identifier(28-bits)
6. **R_c/N_c/N_a/N_s:** Random Number(128-bits)
7. **Pf:** Private Port Number(16-bits)
8. **ReqNo:** Request Number(8-bits)
9. **SK/SK_MS:** Symmetric key shared between MS and AS(128-bits)
10. **DK1:** Delegation key(256-bits)
11. **MAC/H:** Message Authentication Code/Hash(64-bits)
12. **SQ:** sequence number(28-bits)
13. **T_i:**Timestamp(64-bits)
14. **CertSAG:** Certificate of SecurityAcess Gateway(40-bits)
15. **PK:** public key of the server (128-bits)
16. **UAKey:** Primary Key(128-bits)
17. **Expiry/ExpT:** Expiry Time(64-bits)
18. **SK_AS-CA:** Symmetric key shared b/w AS and CA/RA(128-bits)
19. **SK_AS1-AS2:** Symmetric key shared b/w AS1 and AS2(128-bits)
20. **F1:** Message Authentication Code Function

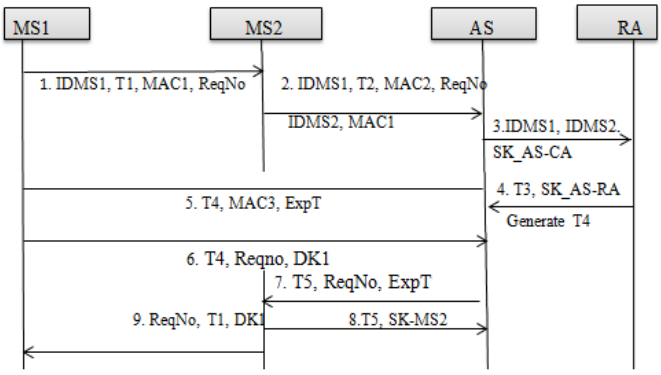
3.3 Scenario-1: Both MS Belong to Same AS

This scenario is presented in Figure 1 where MS1 sends a message to MS2 and both MS belong to the same AS. This scenario is subdivided into two phases.

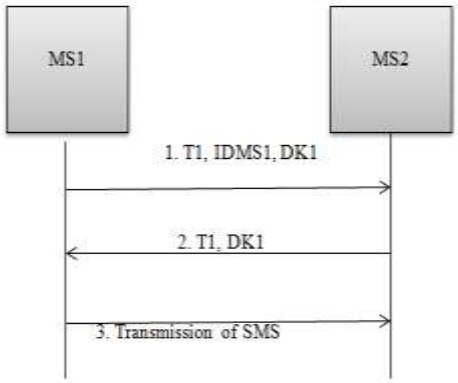
Phase-1: (1) First, the mobile user who wants to send the SMS (say MS1) transmits an initial request to other mobile user (say MS2) for the connection. This initial request consists of International Mobile Subscriber Identity (IMSI) of MS1 (say IDMS1), a timestamp T1, a request number ReqNo and a message authentication code MAC1 = f1SK1 (IDMS1 ||ReqNo). Here, SK1 is a symmetric key shared between the MS1 and the AS2.On receiving the message from MS1, the mobile user who receives this request (say MS2) computes the MAC2 = f1SK2 (IDMS2||T2||MAC1). Then MS2 sends a message to the AS containing the IDMS1, IDMS2, T2, MAC1, ReqNo and MAC2 where IDMS2 is the IMSI of the MS2. The SK2 is a symmetric key shared between MS2 and the AS. With this message, the MS2 requests to the AS to check the validity of the IDMS1. When the AS receives a message from the MS2, it computes the MAC2' = f1SK2 (IDMS2||T2||MAC1) and compares it with the received MAC2. If it holds then the AS sends not only the IDMS1 but also the IDMS2 to the CA/RA along with a timestamp T3 using a symmetric shared key between AS and CA/RA (say SK_AS-CA) to validate the identity of both MS. If, MAC2 and MAC2' are not equal then the connection is terminated. Next, the CA/RA checks the validity of both entities and sends the reply back to the AS with the received timestamp T3. On receiving the message from the CA/RA, if the AS finds any of the entities is invalid then the connection is simply terminated and MS1 needs to send a fresh connection request. If both entities are valid then the AS generates a new timestamp T4, an expiry time to authenticate MS1 (say ExpT), a delegate key DK1 generated from the SK1 using a function f2 and a new message authentication code MAC3=f1SK1 (T4||ExpT||ReqNo) and DK1= f2SK1 (T4||ReqNo). Then the AS sends (T4, MAC3, and ExpT) to the MS1. After receiving the message from AS, the MS1 first computes MAC3' and compares it with the received MAC3, where MAC3'=f1SK1 (T4||ExpT||ReqNo).If both are same then MS1 computes the DK1. Next, MS1 sends T4 and the corresponding ReqNo to the AS encrypted with the DK1 key. The AS checks the received T4 with its stored value and confirms ReqNo. If both are correct then the authentication of MS1 is completed. Thereafter, the AS sends DK1 to the MS2 along with a new timestamp T5, ExpT and ReqNo after encrypting all using the SK of MS2 (SK_MS2) which is a shared key between AS and MS2. The MS2 simply confirms the reception of DK1 key by replying to the AS, the T5 encrypted with the SK of MS2. MS2 also sends ReqNo and T1 to the MS1 encrypted with DK1 so that MS1 can verify the correctness of T1 and ReqNo. This message also verifies the successful reception of DK1 by the MS2.

Phase-2: Once both MS have a shared secret symmetric key, they can exchange the message information in a secure manner using a suitable and strong cryptographic algorithm like AES/ MAES. After phase-1, a session is generated which provides the secure communication between both MS for a specified time period ExpT. In this time period the same DK1 key is used to provide ciphering between MS1 and MS2 but after the ExpT time the session gets expire and MS1 needs to send a fresh request to MS2 with a new request number ReqNo with the same procedure of phase-1. Within the ExpT, the following steps are used for the communication between both MS:

The MS1 sends the IDMS1 and a timestamp (say Ti) to the MS2 encrypted with symmetric key of MS1 i.e., DK1. MS2 decrypts the message using the same DK1 key and checks the validity of IDMS1 and verifies whether Ti <= ExpT. If both are correct then MS1 is successfully authenticated and proved as a valid user for the connection. Then MS2 replies the same received Ti encrypted with DK1 as an acknowledgement to MS1. Secure SMS communication between both MS takes place



(a)



(b)

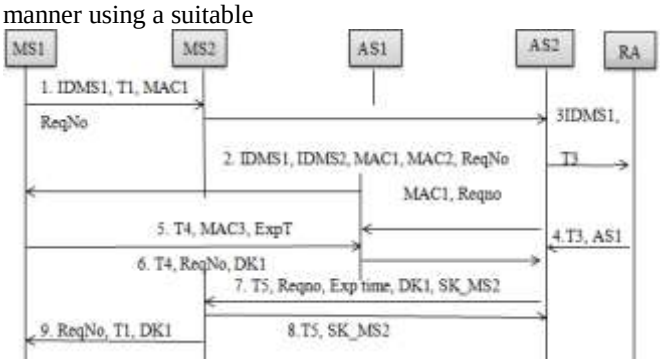
Figure 1: EasySMS Protocol Scenario 1: (a) Phase-1; (b)Phase-2

3.4 Scenario-2: Both MS Belong to Different AS:

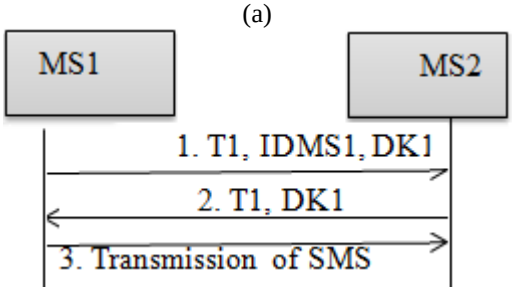
This scenario is presented in Figure 2 where MS1 sends a message to MS2 while both MS belong to the different AS. This case is one where both mobile users are located in the geographically far areas and they have different authentication centers. It may be the case where both MS are of different service providers so they genuinely have different authentication centers. This scenario is also subdivided into two phases.

Phase-1: It is same as presented in step-1 of scenario-1. Here, SK1 is a symmetric key shared between MS1 and AS1. The MS2 passes (IDMS1, IDMS2, ReqNo, T2, MAC1, and MAC2) to the AS through which it is connected (say AS2). The SK2 is a symmetric key shared between MS2 and the AS2. With this message, the MS2 requests to the AS2 to check the validity of the IDMS1. The MS2 stores the timestamp T1 in the memory which was received from the MS1.The AS2 computes the same as presented in of scenario-1 and checks whether MAC2? =MAC2'.The CA/RA checks the validity of both entities and sends the reply back to the AS2 with the received timestamp T3 and the identity of AS to which MS1 belong (say AS1). The AS2 checks the same as in scenario-1 step-5, if both entities are valid then the AS2 sends (IDMS1, ReqNo, MAC1) to the AS1 through a secure channel or using a symmetric key shared between AS1 and AS2 (say SK_AS1-AS2). We assume that all AS communicate with each other using the pre-computed symmetric shard keys. When the AS1 receives the message from the AS2, it computes MAC1'= f1SK1 (IDMS1||ReqNo) and compares MAC1' with the received MAC1. If both are different then the connection is terminated. If both are same then the AS1 generates a new timestamp T4, an expiry time to authenticate MS1 (say ExpT), a delegate key DK1 generated from the SK1 of MS1 using a function f2, and a MAC3, where MAC3 = f1SK1 (T4||ExpT||ReqNo) and DK1 = f2SK1 (T4||ReqNo). Then the AS1 sends (T4, MAC3, and ExpT) to the MS1.After receiving the message from AS1, MS1 repeats the same as in scenario-1 and sends (T4, ReqNo) to the AS1 encrypted with DK1 key. The AS1 checks T4 and ReqNo as in scenario-1.Then AS1 convey the confirmation of the authentication of MS1 by sending a message (ReqNo, ExpT, and DK1) to the AS2 using SK_AS1-AS2 key. The AS2 sends DK1 to the MS2 along with a new timestamp T5, expiry time ExpT and request number ReqNo after encrypting all using the SK of MS2 (say SK_MS2) which is a shared key between the AS2 and the MS2. Step: 10 MS2 repeats the same as in scenario-1 step-8, and sends encrypted reply of T5 to the AS2. It is same as in scenario-1 step-9.

Phase-2: Once both MS have a shared secret symmetric key, they can exchange the message information in a secure manner using a suitable



(a)



(b)

Figure 2: EasySMS Protocol Scenario 2: (a) Phase-1; (b)Phase-2

and strong cryptographic algorithm like AES/ MAES (explained later). After phase-1, a session is generated which provides the secure communication between both MS for a specified time period ExpT. In this time period the same DK1 key is used to provide ciphering between MS1 and MS2 but after the ExpT time the session gets expire and MS1 needs to send a fresh request to MS2 with a new request number ReqNo with the same procedure of phase-1. Within the ExpT, the following steps are used for the communication between both MS:

The MS1 sends the IDMS1 and a timestamp (say Ti) to the MS2 encrypted with symmetric key of MS1 i.e., DK1.MS2 decrypts the message using the same DK1 key and checks the validity of IDMS1 and verifies whether Ti <= ExpT. If both are correct then MS1 is successfully authenticated and proved as a valid user for the connection. Then MS2 replies the same received Ti encrypted with DK1 as an acknowledgement to MS1. Secure SMS communication between both MS takes place.

4. Security Analysis with Confidence Interval

We have also calculated the range of confidence interval, considering it 95% for each algorithm with 160 characters as input because the reported margin of error is typically about twice the standard deviation. Confidence interval is an

interval estimate of a population parameter and is used to indicate the reliability of an estimate. It represent the range of confidence interval (high & low range values) for both encryption (E_low_interval, E_high_interval) and decryption (D_low_interval, D_high_interval) of the message (SMS) with 160, 320, 480, 640 and 800 characters in length for DES, Triple-DES with 2-keys, Triple-DES with 3-keys and AES algorithms where all times are in nanoseconds. We have used t-distribution to calculate the confidence interval because it computes confidence intervals for large 'n' (100 samples in our analysis) if the data is not normally distributed. In this process, the SMS size from 160 to 800 characters is evaluated where more than 160 characters in an SMS is split and concatenated with another SMS. Thus, transmitted message can contain a range of 1120 to 56000 bits where each character is mapped with 7-bit ASCII value. A low standard deviation indicates that the data points tend to be very close to the mean, whereas high standard deviation indicates that the data points are spread out over a large range of values. Since, the AES algorithm is strict to its output range; hence, it is best among them.

4.1 Bandwith Utilization

This subsection evaluates the bandwidth utilized by all three protocols and compares them with respect to each other. It presents the bandwidth utilization of EasySMS Protocol with respect to SMSSec and PK-SIM protocols. It can be easily concluded that on an average, the EasySMS Protocol reduces 51% and 31% of the bandwidth consumption during the authentication process as compared to SMSSec and PK-SIM respectively, while the number of authentication requests is considered as 10, 50, 100, 200, 500, and 1000. Similarly, shows that EasySMS protocol reduces 62% and 45% of the message exchanged in comparison both protocols respectively.

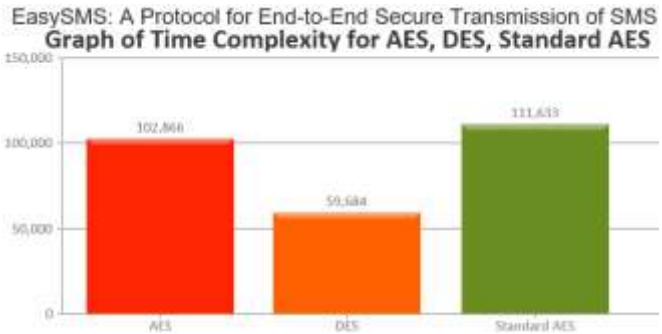


Figure 3:.Graph of time complexity for AES, DES and Standard AES

Table1: Calculation of Execution time

t_aes (ms)	t_des(ms)	t_stdaes(ms)
102866	59684	111633
150753	72354	113930
160376	70766	120510
75279	52055	81021

Average Time AES: 122318.5ms
Average Time DES: 63714.75ms
Average Time stdaes: 106773.5ms

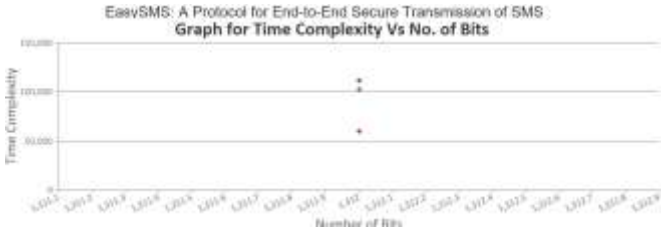


Figure 4: Graph of Time Complexity versus Number of Bits

5. Conclusion

It is an effective protocol for end-to-end secure transmission of secure message service (SMS). This Protocol utilizes bandwidth efficiently and also reduces the message exchange. There are two phases first phase Authentication Phase in this phase of Easy SMS protocol both the mobile stations authenticate each other, after the authentication is successful there is the actual message exchanged between two mobile stations which is done appropriately. Messages are encrypted at the sender side and they are decrypted at the receiver side. In the database of the Authentication Server (AS) the symmetric keys are securely stored. Symmetric algorithms are used they are faster as compared to asymmetric algorithms. The execution time of AES algorithm is 102.866sec and DES algorithm is 59.684sec. DES algorithm takes minimum time to encrypt.

6. Future Scope

In the future work, the protocol can be improved along several directions. The Protocol can be implemented in real time. Malware attack can be reduced since it is implemented for next coming years. Since the technology of SMS is increasing every year, security will be the highest priorities which enable the user to communicate or receive any confidential message for the purpose of verification as well as to reduce the tendency for leakage of information. Since the protocol is stronger and harder to crack, newly related attacks can be prevented by this protocol.

References

[1] R. E.Anderson *et al.*, "Experiences with a transportation information system that uses only GPS and SMS," in *Proc. IEEE ICTD*, no. 4, Dec. 2010.
[2] D.Risi and M.Teófilo, "MobileDeck: Turning SMS into a rich user experience," in Proc. 6th MobiSys, no. 33, 2009.
[3] K.Yadav,"SMSAssassin: Crowdsourcing driven mobile-based system for SMS spam filtering," in Proc. Workshop Hotmobile, 2011, pp. 1–6.
[4] J. Chen, L. Subramanian, and E. Brewer, "SMS-based web search for low-end mobile devices," in Proc. 16th MobiCom, 2010, pp. 125–135
[5] B. DeRenzi *et al.*, "Improving community health worker performance through automated SMS," in Proc. 5th ICTD, 2012, pp. 25–34.
[6] M. Densmore, "Experiences with bulk SMS for health financing in Uganda," in Proc. ACM CHI, 2012, pp. 383–398.

[7] J. Hellström and A. Karefelt, "Participation through mobile phones: A study of SMS use during the Ugandan general elections 2011," in Proc. ICTD, 2012, pp. 249–258.
[8] M. Hassinen, "Java based public key infrastructure for SMS messaging," in Proc. 2nd ICTTA, 2006, pp. 88–93.
[9] S. Wu and C. Tan, "A high security framework for SMS," in Proc. 2nd Int. Conf. BMEI, 2009, pp. 1–6.
[10] A. De Santis, A. Castiglione, G. Cattaneo, M. Cembalo, F. Petagna, and U. F. Petrillo, "An extensible framework for efficient secure SMS," in Proc. Int. Conf. CISIS, 2010, pp. 843–850.
[11] M. Toorani and A. Shirazi, "SSMS—A secure SMS messaging protocol for the m-payment systems," in Proc. IEEE ISCC, Jul. 2008, pp. 700–705.
[12] 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," Comput. Standard Interf. vol. 31, no. 4, pp. 629–641, 2009.
[13] Johnny Li-Chang Lo, Judith Bishop, J.H.P. Eloff "SMSSec: An end-to-end protocol for secure SMS" journal homepage: www.elsevier.com/locate/cose
[14] J. Choi1, J. Kim, J. Sung, S. Lee, and J. Lim, "Related-key and meet -in-the-middle attacks on triple-DES and DES-EXE," in Computation Science and Its Applications (Lecture Notes in Computer Science), vol. 3481. Berlin, Germany: Springer-Verlag, 2005, pp. 567–576.
[15] E. Biham, "Design tradeoffs of the AES candidates," in Asia crypt (Lecture Notes in Computer Science). New York, NY, USA: Springer-Verlag, 1998.
[16] R. Rischpater, "Messaging with wireless API," in Beginning Java ME Platform. New York, NY, USA: A press, 2009, pp. 373–407.

Author Profile

Vaidehi V.Mantri received B.E and M.E degree in Computer Science in 2014 and 2017 respectively.

M.Y.Joshi received B.E, M.E and Ph.D. in Computer Science from SRTMUN University Nanded