I. J. Computer Network and Information Security, **2012, 1, 45-51**
Published Online February 2012 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijcnis.2012.01.06

Enhancing Hash Function Selection Techniques Based on Message Contents

Ali Saeed[1], Muhammad Khalil Shahid[2]
Department of Telecommunication Engineering
PTCL Institute of Communication Technologies, H-9, Islamabad, Pakistan
[1]alisaeed01@yahoo.com, [2]khalil.shahid@ptcl.net.pk

Abstract- In Hash based Security systems two major factors that are mostly relied upon are Strong Hash function and the selection procedure of the hash function from a given pool. This paper aims at exploiting maximum available resources a message possesses, intrinsically, that can accommodate greater number of hash functions references. It provides a simple, low cost- easy to implement technique that will be able to make systems available with random hash functions' selection ability. With the given technique the security level will be enhanced along with greater availability of hash functions. The truly variable nature of contents of messages can be exploited in order to secure messages beyond measure. In case of a single communication stint, if one hash function is compromised the next hash function for next block will be selected truly randomly and cannot be predicted. A summary of already in use techniques is also discussed in order to prove the proposition distinct and practicable. In proposed technique it is proven that it has ability to accommodate greater number of hash functions. Further, the hash function selection methodology has been provisioned with a technique to be message-dependent; the security cannot be compromised owing to truly randomness of the selection procedure.

Index Terms- HMAC, Hash Function Table, Hash Functions' Random Selection Techniques

I-Introduction

The prime necessity in the world of computers and data communication is the provision of ways to check message integrity. The authentication and integrity aspect of a communication process has developed rapidly in recent years. The methods employed have matured for the fulfillment of requirements for given scenarios.

A mechanism that involves use of secret key is generally known as preservation of message integrity and authentication check through MAC (message authentication codes). In layman terminologies, the MAC is basically a short piece of information used to authenticate a message.

The authentication process when coupled with the message integrity is interchangeably called MIC message integrity code. Hash based message authentication codes are a further step forward in the same field of message authentication and integrity. The hash based security systems have three main ingredients:
A message which needs to be kept integrated and its authenticity should be proven. Second ingredient is the hash function which acts on a message under a specific algorithm or procedure. Last, a secret key known to the sender and the receiver [1].

A specific mathematical or logical operation or algorithm could act on variable message size message and produce a code. This code once concatenated with the message at the sender's end will be sent to the receiver. Receiver on reception will remove the code and evaluate the message to ensure its integrity. If the evaluation of message results in same MAC generation as received with original message, the message will be accepted.

The advantage of hash functions usage encircles its simplistic approach to work with following constraints [2]

- Ability to manipulate variable data size
- Fixed length output
- Easy to calculate once key available
- Easy in implementation in hardware and software
- Intrinsic property to be one-way
- Collision resistance

In specific scenarios, authentication that used conventional encryption got obsolete because of computation complexity constraints and same came out to be case of MAC bases systems. The usability of hash-based message authentication codes became pervasive because of the ease of using it in Public Key environments.
A typical one-way hash function works in an extremely simple flow. The introductory example is given in below figure to further illustrate the system [3].

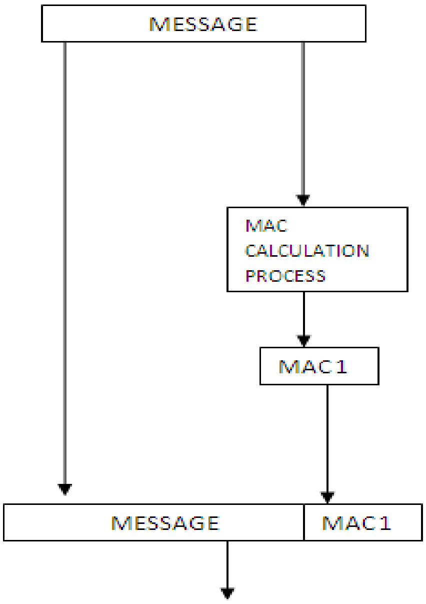


Figure I. Sender side calculation of MAC. MAC Calculation process involves input of Message and a secret key (not displayed here)

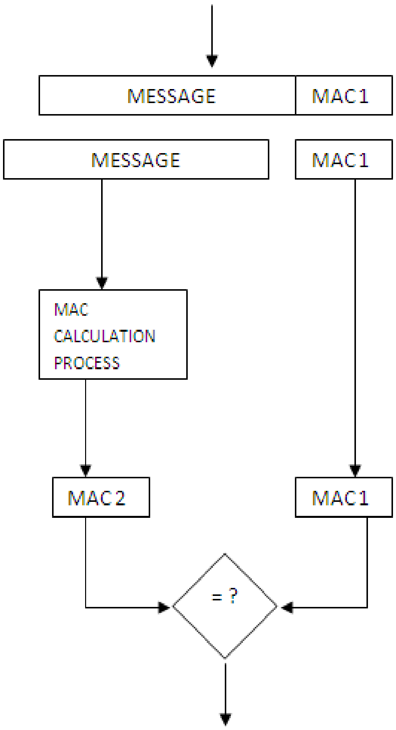


Figure II. Receiving side Evaluation of MAC and integrity and authentication check

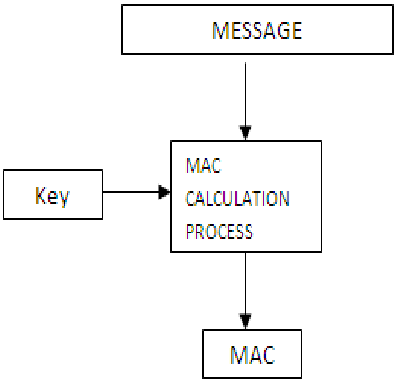


Figure III. Message and Key as the inputs to the algorithm for the calculation of MAC

After the introduction, a skimmed overview of related, pertaining, initial work of this paper will be discussed. In section III, currently in-use techniques and a way forward for new techniques is discussed. The evolution and need for the newer techniques are discussed to make reader get complete hold of the under-discussion techniques. In the last subsection of the same section, the new proposed techniques, along with its implementation results, is discusses. The newer technique is objectively observed for the capacity it can accommodate and the modifications and additional resources required for the newer techniques implementation. The IV section discusses results obtained and the future work that should be done in order to test the developed system on the computational complexity grounds.

II- Related Work

The aspect of greater security depends on the selection of hash function from within a given set of functions. They are operative once the selection procedure ensures greater capacity to address good number of hash functions. Further intelligent selection is also an aspect of securing the system [4].

Ross Anderson discussed the problem in the multicast distribution scenario in data communication. The problems in such systems state the hash function reliability and complex selection methodology so that the authentication and integrity of the message could be ensured [5].

A. Ballardie and J Crowcraft discussed the problem in the multicast security schemes. The threats and counter measures were discussed that existed even with the use of such authentication methods. The chances of compromising of hash functions were the central theme of the paper [6].

Albert Meixner and Andreas proposed enhancement of hashes (visual hashes) which are key dependent and the dependency made the scheme used for selection greatly responsible for security process [7].

Yedidya Hilewitz and Yiqun Lisa Yin, also discussed selections schemes based on messages. The message contents can provide the reference for the respective hash function in the hash table that could be evaluated to check the authentication and integration service [8].

It is worth mentioning that the computational complexity in such schemes could put additional constraints on the security mechanism and the systems. To simplify this process and to ensure lower computational complexity the comparative analysis is required as discussed in work of D. Sharmila and R. Neelaveni. A comparative analysis and evaluation method was given that should be implemented to evaluate the already given and now being proposed system [9].

III- Methodology

In first part of this portion, preferred and currently in practice scenarios are discussed. Finally we discussed a scheme that has been successfully implemented and populated with a number of hash functions. The hash table in proposed scheme has a proven greater accommodation capacity along with ability to rearrange the hash function table dynamically. The rearrangement and hash function selection process is made contents dependent. Thus the need to use secret key or any other additional over head has been nullified.

1. Hash function selection on the base of whole array, single dimension scheme

The diagram illustrates the procedure adopted for the function selection from the table. The message plays a central role in providing the key. From a given set of functions the pointer comes from message.

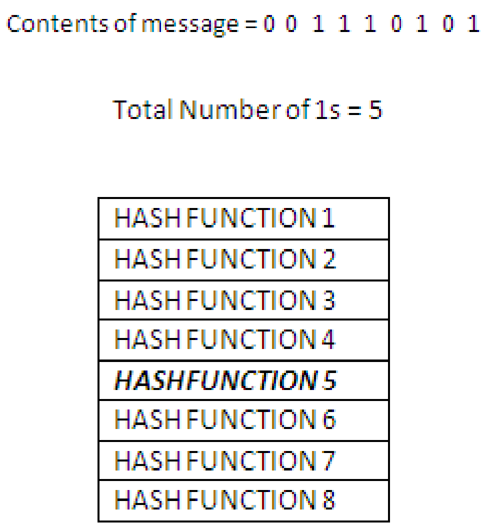


Figure IV. Calculation of HF on the base of contents of the message

The process can be summarized as

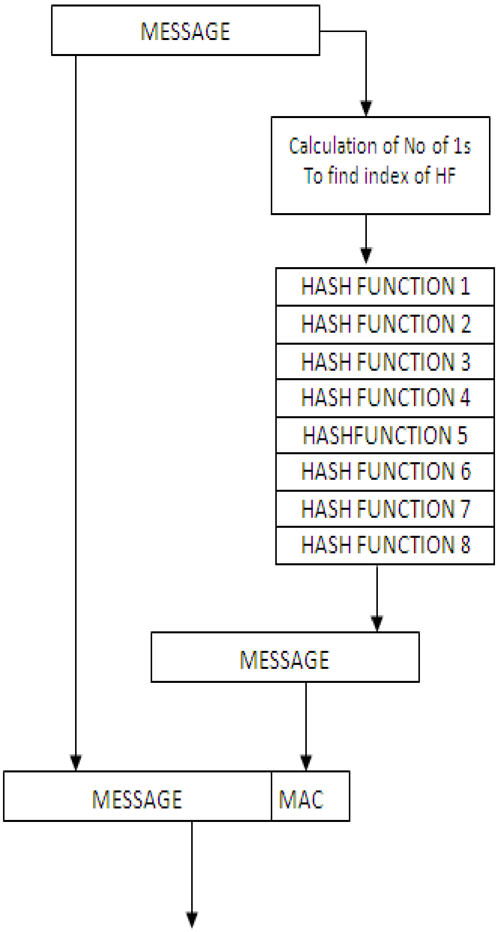


Figure V. Sender side of the content based hash function index finding

- Calculation of number of ones in a message
- This calculated value provides the reference for the available function values
- The referred hash function is used to calculate the authentication code for the given message
- The MAC is later appended to the message and sent over the transport layer
- On the reception side the MAC and message are separated
- On the base of message the reference is found out and Hash function is selected
- The selected hash function is used to calculate the MAC
- If the this calculated MAC is the same as that which is received, the message is accepted and rejected otherwise

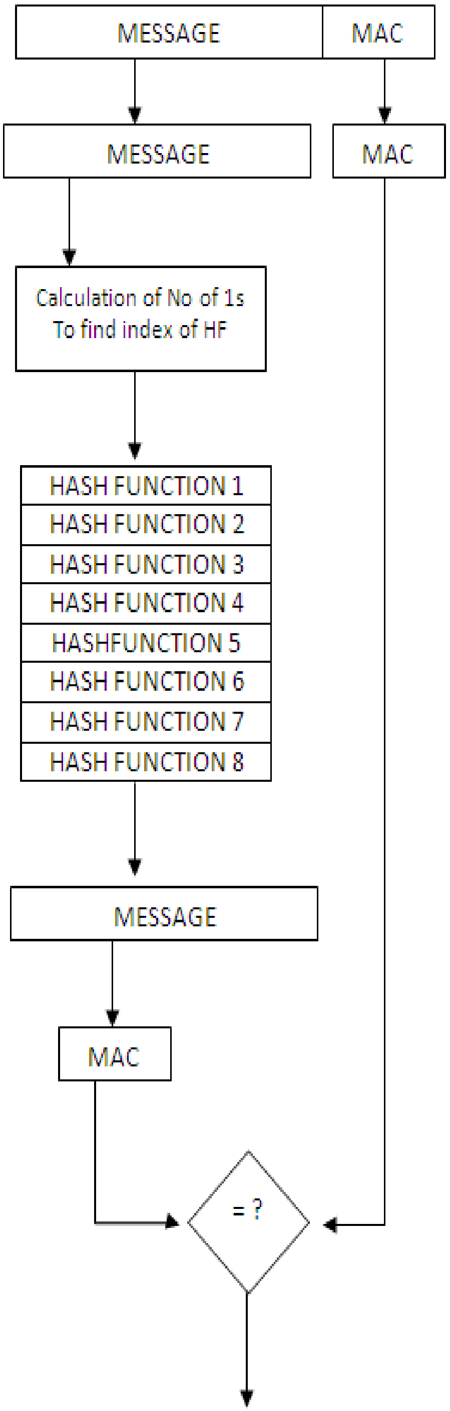


Figure VI. Receiving side of the content based hash function index finding

2. Hash function selection on the base parts of array, bi-directional scheme

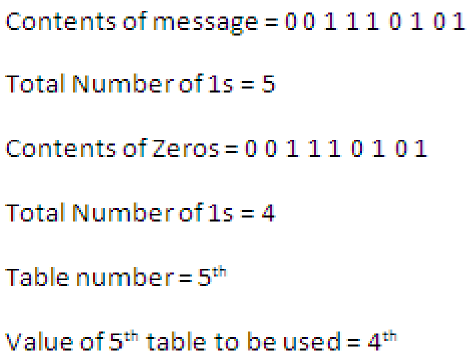
This scheme is also effective as far as making system dependent upon a random selection process. This random value is very difficult to compromise. On top of it, it is very difficult to compromise the hash function. In case compromise is accomplished, single communication duration can still be made secured as the upcoming message chunk will be with different hash function as the hash function depends upon the message contents.

3. Expanding Hash function table, introduction of Transpose method in bi-directional scheme

This scheme has following provisions in addition to the previously proposed schemes

- The original message contents be used for hash function table selection i.e. the number of ones present in the message content
- After the selection of table the number of zeros in the message is used to find the reference for the hash function
- This hash function is used to calculate the authentication code
- Before the selection of hash function the arrays are provided with the option to refresh in accordance with the message contents. Here the process of transposition is added in order to frustrate already doomed attacker
- The hash functions that are available in ‘n’ number of tables and ‘n’ in number in each table provide great flexibility to use any hash function.
- The transpose is a functionality that can be done through a handshaking procedure in order to refresh the scheme time and again.

The proposed model has been successfully implemented in normal C paradigm. The rest of the processes can be summarized as follows:



Contents of message = 0 0 1 1 1 0 1 0 1

Total Number of 1s = 5

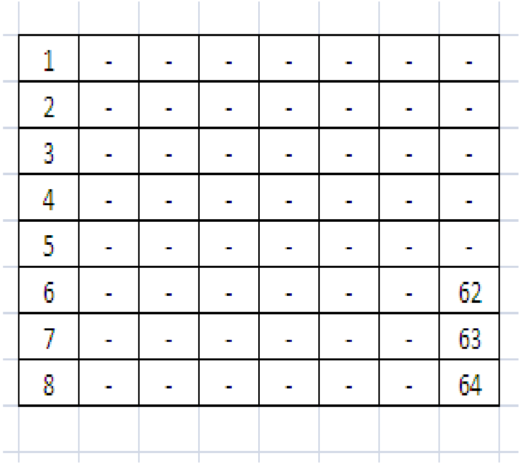
Contents of Zeros = 0 0 1 1 1 0 1 0 1

Total Number of 1s = 4

Table number = 5th

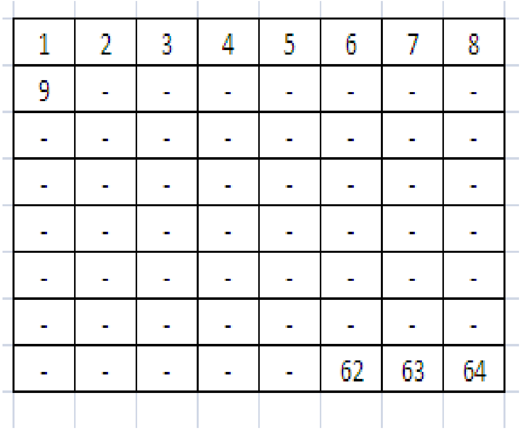
Value of 5th table to be used = 4th

Figure VII. Calculation of Table number and the hash function to be used in figure the Total number of 1s is an ODD number therefore the entries will be transposed



1	-	-	-	-	-	-	-
2	-	-	-	-	-	-	-
3	-	-	-	-	-	-	-
4	-	-	-	-	-	-	-
5	-	-	-	-	-	-	-
6	-	-	-	-	-	-	62
7	-	-	-	-	-	-	63
8	-	-	-	-	-	-	64

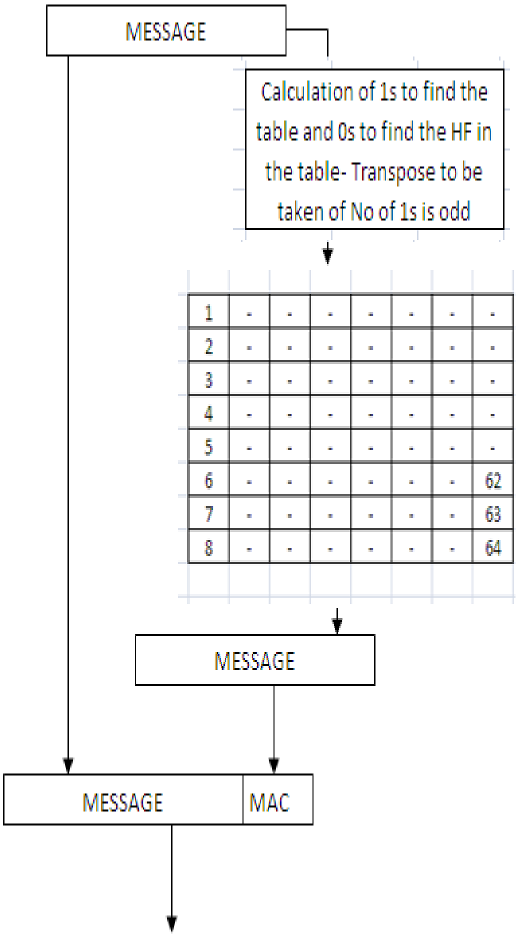
Figure VIII. Original Arrangement at the reception of the message



1	2	3	4	5	6	7	8
9	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-
-	-	-	-	-	62	63	64

Figure VIII. Transpose taken as per direction of the contents of the message (ODD no of 1s)

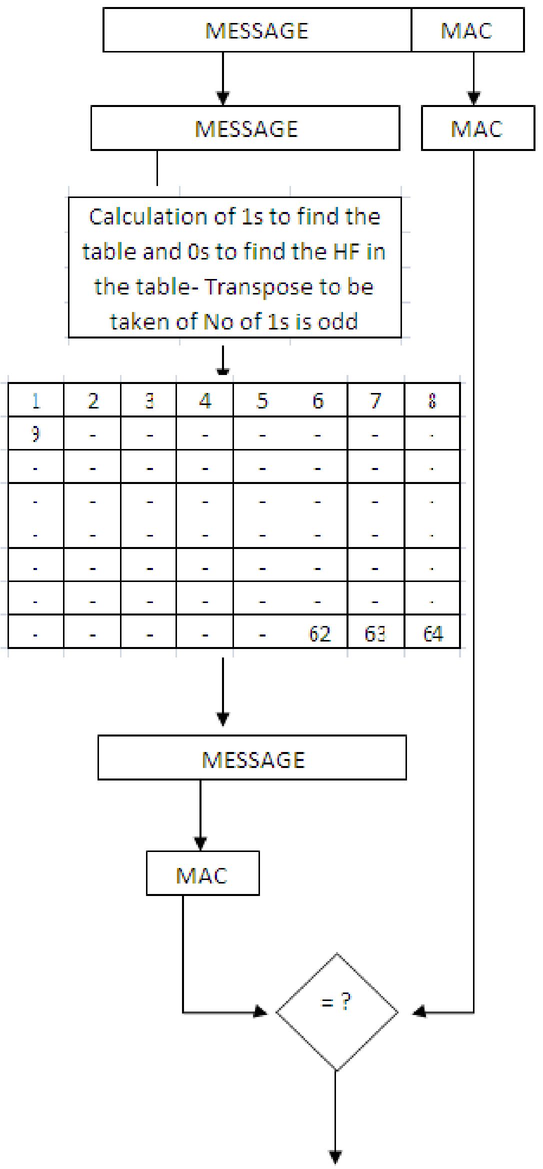
The message is set to count number of ones in it. Then the message is checked for zeros and number of ones in the message again calculated. The first operation will enable the selection of the table from where the HF is to be found out. The zeros in the message are calculated to tell the scheme of hash being used to find out the MAC/HMAC. This means that if the message size is 8 bits the options to evaluate the message hash are 8*8 i.e. 64 in number.


MESSAGE
Calculation of 1s to find the table and 0s to find the HF in the table- Transpose to be taken of No of 1s is odd
MESSAGE
MESSAGE
MAC

1	-	-	-	-	-	-	-
2	-	-	-	-	-	-	-
3	-	-	-	-	-	-	-
4	-	-	-	-	-	-	-
5	-	-	-	-	-	-	-
6	-	-	-	-	-	-	62
7	-	-	-	-	-	-	63
8	-	-	-	-	-	-	64

Figure IX. Sender Side

The transposition to further strengthen the security is implemented and the hash functions are swapped for enhancing the security of the message.


MESSAGE
MAC
MESSAGE
MAC
Calculation of 1s to find the table and 0s to find the HF in the table- Transpose to be taken of No of 1s is odd
MESSAGE
MAC
= ?

1	2	3	4	5	6	7	8
9	-	-	-	-	-	-	.
-	-	-	-	-	-	-	.
-	-	-	-	-	-	-	.
-	-	-	-	-	-	-	.
-	-	-	-	-	-	-	.
-	-	-	-	-	-	-	.
-	-	-	-	-	62	63	64

Message accepted if both the MACs are same. Otherwise rejected

Figure IX. Receiving Side

Since the number of hash functions is equal to the number of tables thus transpose will result into a new arrangement, eventually frustrating any integrity or authenticity of the meesage.

IV- Conclusion

The algorithm proposed and tested after implementation proves to be low cost practicable solution and thus extends the ability to accommodate maximum number of hash functions available. It further provides enhancement of security by refreshing the list of already in use hash functions. If under same arrangement any apprehension needs to be thwarted the transpose of the arrangement will prove to be best possible solution to frustrate the attacker. The security level increases with the use of strong hash function and a dynamic selection method.

It will be important to evaluate the performance of the proposed system in comparison with already implemented systems. Further work shall include the computational complexity analysis of the system on the same model as proposed in hash-based system comparative analysis researches [10].

References

[1] Network Security Essentials, 2/E William Stallings ISBN-10: 0130351288 ISBN 13: 9780130351289 Publisher: Prentice Hall Copyright: 2003. Page(s): 42- 46

[2] John Hopkins University database. "*Reading Guide 2: Keying Hash Functions for Message Authentication*". Hyrum Mills, Chris Soghoian, Jon Stone, Malene Wang September 10, 2004

[3] D.W. Engels, "*Security & Privacy Aspects of Radio Frequency Identification Systems*", Lecture Notes in CS, Springer Berlin / Heidelberg, ISSN 0302-9743 (Print) 1611-3349 (Online), Volume 2802/2004

[4] Croonen, N., Theuwissen, H. (2002): *Table Lookup: Techniques Beyond the Obvious*, Paper 11-27, SUGI 27

[5] Ross Anderson, Markus K, "*low cost attacks on tamper resistant devices*" IWSP: International Workshop on Security Protocols LNCS 1997

[6] A. Ballardie and J.Crowcroft, "*Multicast-Specific Security Threats and Countermeasures*", Proc. ISOC symp. Net. and Distrib. Sys. Sec., San Diego, CA, Feb 1995

[7] Albert Meixner and Andreas Uhl, "*Security Enhancement of Visual Hashes Through Key Dependent Wavelet Transformations*" Department of Computer Science, Duke University, USA

[8] Hilewitz, Y., Yin, Y.L., Lee, R.B.: "*Accelerating the Whirlpool Hash Function Using Parallel Table Lookup and Fast Cyclical Permutation.*" In FSE(2008) 173-188

[9] D. Sharmila and R Neelaveni, "*Performance Evaluation of VHDL Implementation of SAFER+ and AES algorithm for Bluetooth security system.*" ICGST-CNIR Journal, Volume 9, Issue 1, July 2009

[10] Aihab khan, M Iqbal, "*Performance evaluation of hash algorithms for integrity and fatabase archives*" JATIT. 31st Aug 2011 Vol. 30 No.2

Ali Saeed Received Bachelor degree from University of Engineering and Technology Peshawar, Pakistan in 2007. He is currently pursuing his Masters in telecommunication engineering at PTCL Institute of Communication Technologies, Islamabad, Pakistan. His experience chiefly relates to Next Generation Networks, where he has been in technical management of C5-NGN deployment and commission in Pakistan. His research interests are communication networks.

Dr. Muhammad Khalil Shahid received the Bachelor and Master degree from University of Engineering & Technology Lahore and University of Engineering & Technology Taxila respectively. In 2008, he received the PhD degree from Beijing University of Posts & Telecommunications. Currently, he is working as Associate Professor in Institute of Communication Technologies PTCL Islamabad. His research interests include Optical and Wireless Communication, Communication Network Security and Telecom Regulations.