02- Days, International Conference on Recent Trends in Engineering Science, Humanities and Management
Sri S Ramasamy Naidu Memorial College, Sattur, Tamil Nadu, India (RTESHM-17)
02nd-03rd February 2017, www.conferenceworld.in ISBN: 978-93-86171-18-4

A STUDY ON TECHNIQUES THAT IMPROVE THE STRENGTH OF ENCRYPTION ALGORITHMS BASED ON DATA STRUCTURES

S. M Farooq[1], S. Kiran[2], G. Iliyas[3]

[1]*PhD Research Scholar, Dept. of CSE, Y.S.R Engineering College of Yogi Vemana University, (India)*

[2]*Assistant Professor, Dept. of CSE, Y.S.R Engineering College of Yogi Vemana University, (India)*

[3]*Student III B.Tech CSE, Y.S.R Engineering College of Yogi Vemana University, (India)*

ABSTRACT

Achieving secure communication of through insecure channel is the goal of cryptography. Study of techniques of breaking the secure mechanisms to is referred as cryptology. Cryptography and cryptology go side by side to provide more secure techniques. Data structures plays key role in speed up mechanisms and in cryptography as well. The present paper studies about different data structures implemented in different scenarios of cryptographic techniques implementation. The paper further studies their role in increasing the efficiency and performance of the cryptographic technique. The study reveals that the data structures like Hight balanced trees can be used to generate cipher text. Data structs like B+ Trees can be used to store encrypted data to speed up the retrieval process of the database. Data structures with hash functions proved to very effective in achieving cryptographic primitives.

Keywords : Cryptography, Data Structures, Performance, Efficiency.

I INTRODUCTION

The fundamental goal of encryption algorithms is to achieve confidentiality, integrity, authentication and non-repudiation in different communication networks [10]. They are considered as security requirements of communication. Securely transmission of data without eavesdropping is the requirement of confidentiality. Reaching the data to intended receiver without tampering is the requirement of integrity. Checking the source of data as intended sender is the requirement of authentication. Claiming the source of data in case of denial of data generation is the requirement of non-repudiation. Cryptanalysis is the process of attacking cryptographic techniques to get the secret message [11]. Encryption algorithms are categorized as private and public key encryption

ISBN: 978-93-86171-18-4

algorithms. Public key cryptography is used in different web applications. It uses two different keys public and private for authentication as well as encryption. Private key cryptography is based on symmetric key (one) which is used for encryption as well as authentication. RSA (Rivest, -Shamir- Adelman), Data Encryption Standard (DES), Triple Data Encryption Standard (TDES), Advanced Encryption Standard (AES) are different popular encryption algorithms uses different key lengths that gives protection against brute force attack.

Encryption and decryption is a mathematical function that maps a set of plain text symbols to cipher text symbols and vice versa. Classical ciphers such as Caesar, Affine, Vigenere, and Hill cipher are built based on mathematical concepts like Modulo Arithmetic, GCD (Greatest Common Devisor), multiplicative inverse etc. The paper studies about different techniques that uses data structures as a medium for generating cipher text. We have many attempts in the literature that uses different data structures like complete binary tree, B+ trees, Dictionaries, Link based data structures, graphs.

II DATA STRUCTURES IN CRYPTOGRAPHY

In [1], complete binary tree is used as medium to generate cipher text. The plain text is converted into cipher text by applying mirroring, preorder traversals for five times, and the reverse operations gives the plaintext from cipher text. As the tree is constructed in by taking a chunk of 15 characters the height of the tree becomes balance which gives O (log n) time complexity when compare to O(n) searching time for an unbalanced tree, where n is considered as height of the tree. In [2], the authors try prove that the query performance on encrypted data using B+ trees varies very less but gives a feasible result. Generally, database information such as account details, personal information is confidential which should be protected from attacks. So, storing the data in encrypted form and retrieving it by decrypting is prone to timing constraints, since query evaluation is itself requires processing time to retrieve the data from database. So, the paper [2] proposed a framework that quickly implements query on encrypted data based on B^+ trees along with tree index. The two data structures that plays key role in the revocation is Merkle Hash Tree (MHT) and the 2-3 tree. MHT is based on one way hash functions [3]. Hash tree or merkle tree is a tree where non-leaf nodes are hash values generated by one way hash function. One way hash function generates a hash value for a given input. We can generate a hash value for a given input but for a given hash value it is computationally hard to get input value. Hash trees gives effective verification of the contents. Since the revocation process works on encrypted data, one way hash function is faster than normal digital signature for authentication. There are many applications of hash functions such as storage of passwords, data authentication as well as checking integrity etc. but in [3], the hash functions are used in generating tree indexes.

ISBN: 978-93-86171-18-4

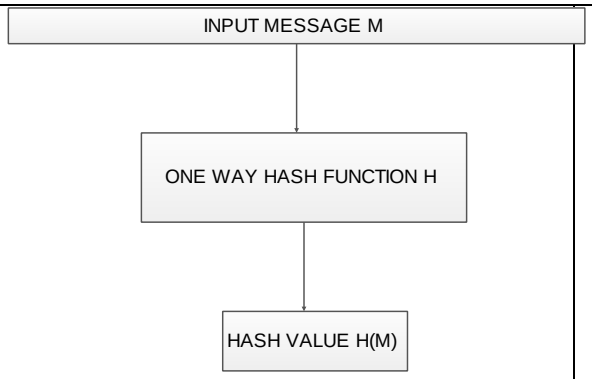

Figure 1 One Way Hash Function

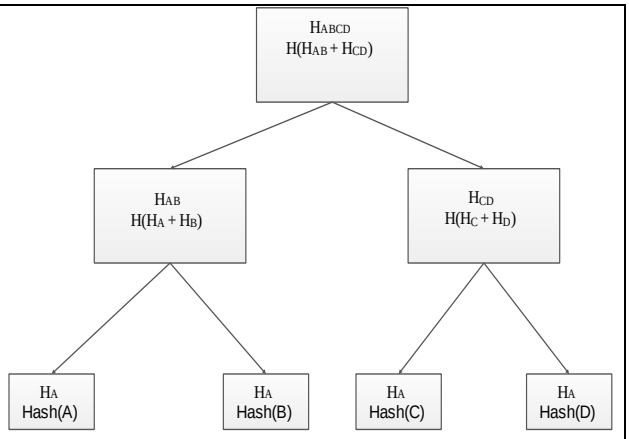

Figure 2 Merkle Hash tree

The Markle Hash tree is constructed by concatenation of leaf node hash values in its parent nodes. The process will be continued up to root node as shown in the figure. The MHT is constructed based on 2-3 tree [12]. The 2-3 tree gives O (log n) time complexity in searching, adding, removing a leaf node. In [4], the authors achieved fast retrieval of data from database using encrypting B+ trees. Generally, data is stored in encrypted form, then B+ tree is constructed to create indices. In [5], the authors implemented a negative database for security purpose. By implementing RSA and Hash algorithms the generic database is implemented based on EAV (Entity Attribute Value) data model. In [6], the authors proposed a protecting mechanism to kernel link data structures based on secure canary. The protecting mechanism secures the linked nodes using secure canary. The secure canary consists of information to be protected with stream cipher.

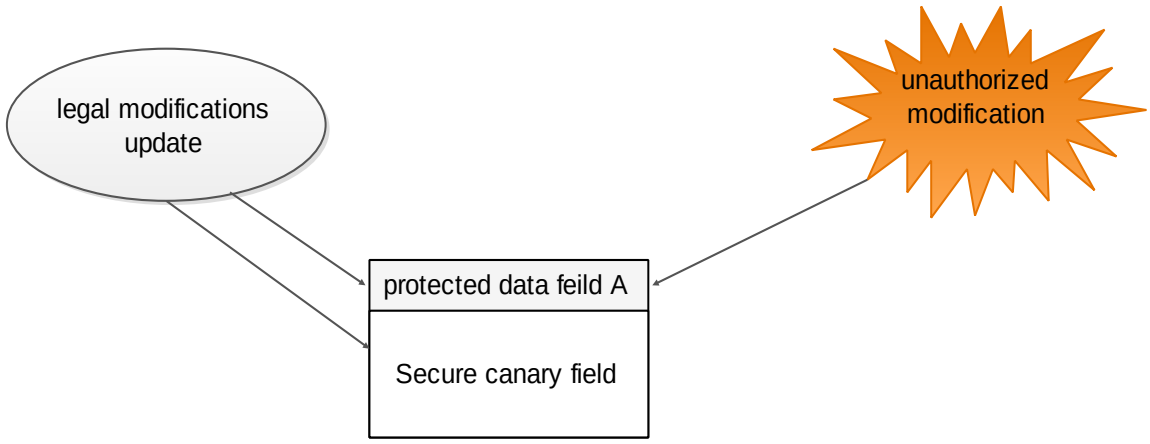

Figure 3 Securing kernel data structures based on secure canary field using stream cipher

ISBN: 978-93-86171-18-4

The above threat model illustrates about unauthorized update of the protected data field A of link node. Any update to link node A should be done through a facility called legal modifications. Some eavesdropper tries to modify the data by bypassing legal modifications then using secure canary field the update action is recorded and can be traceable. To calculate secure canary stream cipher is used. Here the assumption is the seed key of the stream cipher cannot be steal by unauthorized entity. The adaptive radix tree [7] is significantly faster than existing data structures like FAST [8] and B+ Tree [9] acted as high performance index data structure.

III VARIOUS APPLICATIONS

The tables summarized the various applications of data structures in cryptographic techniques.

Table 1. Summary Table

S.No	Data Structure	Application
1	Complete binary tree	Generating cipher text
2	Height balanced tree	Achieving less time complexity in searching, insertion and removal
3	Merkel Hash Tree	Authentication, encrypted data storage.
4	2-3 tree	Storing encrypted data
5	B+ Tree	Indexing
6	Link based data structures	Stream cipher to store encrypted data.

IV CONCLUSION

The paper studies various applications of data structures in cryptography. Various data structures which includes complete binary tree, height balanced tree, complete binary tree, merkle hash tree (MHT), 2-3 trees, radix trees are extensively used in cryptographic applications like generating cipher text, providing authentication and integrity, speed up the process etc. the study reveals that use of data structures with hash functions proved to very effective in achieving cryptographic primitives such as authentication and integrity.

REFERENCES

1] Nikhil Agarwal, Manoj Kumar, Dr. M.A Rizvi, Transposition Cryptography Algorithm using Tree Data Structure, ICICES2014, ISSN No. 978-1-4799-3834-6/14.

ISBN: 978-93-86171-18-4

2] Jose L. Muñoz, Jordi Forné, Oscar Esparza, Miguel Soriano, Implementation of an efficient authenticated dictionary for certificate revocation, Proceedings of the eight IEEE International Symposium on computers and communication (ISCC'03), 1530-1346/03 2003, IEEE.

3] R.C Merkle. A certified digital signature. In advances in Cryptology (CRYPT089). Lecture Notes in computer science, number 435, pages 234-246, Springer – Verlag, 1989.

4] Zheng-fei wang, Al-guo tang, Wei wang, Fast Query Over Encrypted Data Based on B^+ Tree, International Conference on Apperceiving Computing and Intelligence Analysis, **DOI:** 10.1109/ICACIA.2009.5361133, 2009 IEEE.

5] Gaurav Dubey, Vikram Khurana, Shelly Sachdeva, Implementing Security Technique on generic Database, Eighth International Conference on Contemporary Computing (IC3) DOI: 10.1109/IC3.2015.7346709 ©2015 IEEE.

6] Li Wang, Dinghao Wu, and Peng Liu, iCruiser: Protecting Kernel Link-Based Data Structures with Secure Canary, IEEE International Conference on software quality, reliability and security companion, DOI: 10.1109/QRS-C.2016.9

7] Victor Alvarez , Stefan Richter , Xiao Chen, Jens Dittrich, A Comparison of Adaptive Radix Trees and Hash Tables, IEEE 31st International Conference on Data Engineering (ICDE), DOI**:** 10.1109/ICDE.2015.7113370.

8] C. Kim, J. Chhugani, N. Satish, E. Sedlar, A. D. Nguyen, T. Kaldewey, V. W. Lee, S. A. Brandt, and P. Dubey, "Fast: Fast architecture sensitive tree search on modern cpus and gpus," in SIGMOD, 2010, pp. 339–350.

9] J. Rao and K. A. Ross, "Making B+-trees cache conscious in main memory," in SIGMOD, 2000, pp. 475–486.

10] J. Katz, Y, Lindell, Introduction to Modern Cryptography, second edition, CRC Press, 2015.

11] Douglas R Stinson, Cryptography Theory and Practice, Third edition, University of waterloo, Ontario, Canada.

12] A.V Aho, J.E. Hopcroft, and J.D. Ulman. Data Structures, Addison-Wesely, 1988.