International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2019 IJSRCSEIT | Volume 5 | Issue 2 | ISSN : 2456-3307
DOI : https://doi.org/10.32628/CSEIT1952186

A Study on Data Hiding Technique Using Prime Numbers

[1]K. Ravikumar, [2]R. Raja
[1]Assistant Professor, Department of Computer Science, Tamil University (Established by the Govt. of. Tamilnadu), Thanjavur, Tamil Nadu, India
[2]Research Scholar, Department of Computer Science, Tamil University, Thanjavur, Tamil Nadu, India

ABSTRACT

In a globe full of communications, it is necessary to guard the data in arrange to make certain the isolation of the users. There is a lot of responsive data present in the mesh and it requirements. A novel data hitting technique is future, as an development over the Fibonacci LSB data-hiding technique. The traditional LSB technique is the simplest, but using this method it is promising to implant only in few bit-planes, since picture excellence becomes radically misshapen when embed in higher bit-planes. Battista et al. proposed an development over this by using Fibonacci putrefaction modus operandi and generate a deferent set of practical bit-planes all mutually, thereby growing the numeral of bit-planes. we propose two new hashing functions, prime modulo and odd-multiplier displacement, that are resistant to pathological behaviour and yet are able to eliminate the worst-case disagreement performance in the L2 cache. We show that these two schemes can be implemented in fast hardware using a set of thin totalling operation, with insignificant disintegration in the L2 hoard. We assess the scheme on 23 reminiscence intensive applications. It produces different secret message text for the same plain text which confuses the hacker to attain the unique text and makes it near impossible to break it.

Keywords : Prime Factorization, Cryptography, PI, DNA, Cipher Text, Plain Text, Key, Encryption, Decryption.

I. INTRODUCTION

The meticulous depiction generate a diverse set of (virtual) bit-planes overall, apposite for embed purpose. They not only allow one to embed secret communication in advanced biplanes but also do it without much deformation, with a much better steno-image excellence, and in a dependable and protected manner, assurance resourceful recovery of underground message.[1] A qualified routine study flanked by the traditional Least major Bit (LSB) method, the Fibonacci LSB data-hiding method and our planned schemes has been done. Examination indicates that image quality of the steno-image unseen by the method using Fibonacci putrefaction improve alongside that using simple LSB replacement means, while the same using the major putrefaction way improve considerably touching that using Fibonacci decomposition system.[2]

As for conventional computation, there are some key separate algorithms easy to get to such as Rectangular Sieve (QS) , Number Field Sieve (NFS)[2] , Pollard's p-1 method [3] , H. C. William's p+1 method [6], etc. There are also parallel computing methods to solve factor factorization problem such as using hardtop [7]. However, for big numbers, factorization is hard to get. In this paper, we present

a novel method which constructs suitable solution tree by modulo and prunes search branches by equations set and other methods such random, and we also present a parallel strategy by using huge number of nodes.

Data embedding or water marking algorithms necessarily have to guarantee the following:

1.Presence of embedded data is not visible.

2.Ordinary users of the document/image are not abetted by the watermark,

i.e., a normal user does not see any ambiguity in the clarity of the document/image.

3.The watermark can be made visible/retrievable by the creator and pos-sibyl the authorized recipients when needed; this implies that only the creator has the mechanism to capture the data embedded inside the doc-unmet/image.

4.The watermark is midcult for the other eavesdropper to comprehend and to extract them from the channels.

Cryptography

It is the process of generating cipher text from plain text with the help of a key where the cipher text generated is usually present in an incomprehensible format so that it is protected from any kind of unauthorized access. Key plays a major role in the level of encryption. The more complex the key and the encryption process, the more difficult it becomes to break the encrypted text. Cryptography is of two types

Symmetric Key Cryptography

As the name suggests, it is a kind of cryptography where a symmetric key is used i.e. same key is used for both encryption as well as decryption of a text [5]. Some of widely used and most common symmetric encryption techniques are AES, DES, 3- DES, and Blowfish.

AES

Advanced Encryption Standard is introduced to overcome the drawbacks present in DES. In 2001, NIST recommended to replace DES. Its supports data of block length 128 bits and 3 types of combinations of key length of 128, 192 and 256 bits and the number of rounds for processing them is respectively.

Asymmetric Key Cryptography

As the name suggest, it is a kind of cryptography where an asymmetric key is used i.e. two different keys are used namely public key for encryption and a private key for decryption. It is also commonly known as public key cryptography. This is a irreversible cryptographic technique. RSA, El Gamma, Duffy-Hellman Key exchange, Digital Signature, Elliptical Curve Cryptography(ECC) are some of the most common and widely used asymmetric encryption algorithms.

RSA

RSA algorithm is named after its authors Ron Rivest, Adi Shamir and Leonard Adleman. It is the most widely used asymmetric algorithm. It uses two different keys. A public key for encryption and a private key for decryption. It is known for its high security and irreversibility i.e. it is nearly impossible to trace back the private key from public key.

In fact the first and the second methods are the only ones to be considered when a rigorous proof of primality is required. For the first method, the problem of factoring the special function of s can be overcome by generating random factorizations instead of random numbers to be tested for primality. Williams and Schmid, Buhler, Crandall and Penk, and chiefly Plaisted propose such a method which appears to be quite general and, with some adaptations, suitable for our purposes. Thus we have adapted their method and extended their results.

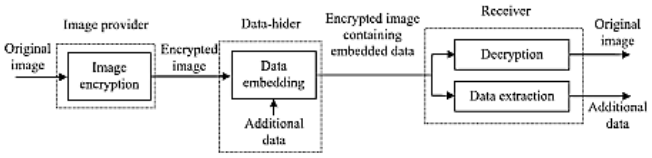
II. METHODS AND MATERIAL

EEA-PDP is an encryption algorithm which is used to encrypt the users data. It is a symmetric encryption algorithm which uses the same key for both encryption and decryption. Keys of EEA-PDP are generated from DNA Sequence and PI Sequence. The DNA sequence is obtained from the National Centre for Biotechnology Information (NICB)[18]. The EEA-PDP algorithm uses these two keys along with a prime number for encryption and decryption of data.

Data Hiding and the Prime Decomposition Technique

If we have k-bit cover image, only k bit-planes are available to embed secret data. Distortion increases exponentially with increasing bit-plane, it becomes impossible to embed data in higher bit-planes.

So, our primary target here is to increase the total number of available (and embeddable) bit planes without much distortion.



Qualitative Comparison

Compares the various hashing functions based on when the ideal balance is achieved, whether they satisfy the sequence invariance property, whether a simple hardware implementation exists, and whether they place restrictions on the replacement algorithm. The major disadvantage of the traditional hashing is that it achieves the ideal balance only when the stride amount s is odd.

After embedding the secret message bit, we convert the resultant sequence in prime number system back to its value in classical binary number system and get our stego-image. The extraction algorithm is exactly the reverse. From stego-image, we convert each pixel with embedded data bit to its corresponding prime decomposition and from pth bit-plane extract secret message bit. Combine all bits to get the secret message.

ASSESSMENT

The present and discuss five sets of evaluation results. We present the impact of using different single hashing functions on the various types of cache misses and on the cache miss distribution across the cache sets. The impact of using multiple hashing functions in conjunction with a skewed associative L2 cache on the various types of cache misses. Section 5.4 shows the performance gain achieved using the different hashing functions for the 23 applications. Finally, Section 5.5 shows the impact of our prime hashing functions on the execution time for various cache configurations.

Key Exchange

It is essential to pass the cipher text as well as the keys to the desired user after encryption. There is always a chance that a third party can spoof the network and steal the data. Though the data is in incomprehensible format the attacker can corrupt the data. So it is also essential to send the data over a secure channel. In order to tackle the secure transmission problem and prevent such kind of Man in the Middle Attacks (MIM Attack), Diffie-Hellman Key exchange algorithm can be used where it will generate the key on both sides rather than transmitting it. It is an Asymmetric algorithm and is irreversible.

Attending the primes, we create a map of k-bit (classical binary de-composition) to n-bit numbers (prime decomposition), $n > k$, marking all the valid representations (as discussed in previous section) in our prime number system. For an 8-bit image the set

of all possible pixel-values in the range has the corresponding prime decomposition. As one may notice, the size of the map to be stored has been increased in this case, indicating a slightly greater space complexity.

Next, for each pixel of the cover image, we choose a (virtual) bit plane, say path bit-plane and embed the secret data bit into that particular bit plane, by replacing the corresponding bit by the data bit, if and only if we ¯nd that after embedding the data bit, the resultant sequence is a valid representation in n-bit prime figure system, i.e., exists in the map if not throw away that meticulous pixel for data hiding.

After embedding the secret message bit, we convert the resultant sequence in prime number system back to its value (in classical 8-4-2-1 binary number system) and we get our stego-image.

Single Hashing Function Schemes

The normalized number of cache miss in each submission with no uniform cache accesses and uniform cache access, correspondingly. Each outline compares the number of cache misses with different hashing functions: a conventional hash purpose with 4-way associative L2 cache (Base), a traditional hashing function with an 8-way associative same-size L2 cache (8-way), the XOR hashing function (XOR), the prime modulo hashing function (pMod), and the odd-multiplier displacement hashing function (oDisp), as described in Section 3.3. The number of misses in each case is normalized to Base. All bars are divided into: cold misses (Cold), capacity misses (Capacity), and conflict misses (Conflict). The cold misses are computed as the sum of the first miss to each line that is accessed. The conflict misses are calculated as the number of misses eliminated if a fully associative L2 cache is used. The rest of the misses are categorized as capacity misses.

Prime Decomposition gives less distortion in higher bit-planes Here, we assume the secret message length (in bits) is same as image size, for evaluation of our test statistics. For message with die rent length, the same can similarly be derived in a straight-forward manner.

In case of our Prime Decomposition, WMSE for embedding secret message bit only in lath (virtual) biplane of each pixel (after expressing a pixel in our prime number system, using prime decomposition technique) = p2l , because change in lath bit plane of a pixel simply implies changing of the pixel value by at most lth prime number.

III.CONCLUSION

The level though using substitute hoard hash functions is a well-liked method to decrease divergence misses by achieve a more uniform hoard right of entry allocation crossways the sets in the hoard, no prior study has actually analyzed the pathological behaviour of such hash function that often consequence in performance squalor. we have used them to make the Key in our algorithm. This makes the Key hard to fracture it and ensure the safety of information. particularly our algorithm produce different secret message text for same plain text which confuse the cryptanalyst and makes it hard to break it. As shown in the consequences our algorithm is well-organized than the influential AES and is more complex than most of the customary techniques.

IV. REFERENCES

[1]. F. Battisti, M. Carli, A. Neri, K. Egiaziarian, A Generalized Fibonacci LSB Data Hiding Technique, 3rd International Conference on Computers and Devices for Communication (CODEC-06), Institute of Radio Physics and

Electronics, University of Calcutta, December 18-20, 2006.

[2]. C. Shao-Hui, Y. Tian-Hang, G. Hong-Xun, Wen, A variable depth LSB data hiding technique in images, International Conference on Machine Learning and Cybernetics, 2004,, Vol. 7, 26-29 pp.3990 – 3994, 2004.

[3]. J. M. Pollard, Theorems on Factorization and Primality Testing, Procedings of Cambridge Philosophy Society, 76 (1974), 521{528}.

[4]. P. Shor, Algorithms for Quantum Computation:Discrete Logarithms and Factoring , Proceedings of 35th Annual Symposium on Foundations of Computer Science, IEEE Computer Society Press, 1994,124-134.

[5]. R. D. Silverman, The Multiple Polynomial Quadratic Sieve , Mathematics of Computation, 48 (1987), 329-339.

[6]. Tutubalin Pavel Innokentievich, Mokshin Vladimir Vasilevich, "The Evaluation of the Cryptographic Strength of Asymmetric Encryption Algorithms", In the Proceeding of the 2017 Second Russia and Pacific Conference on Computer Technology and Applications (RPC), pp.180-183, 2017.

[7]. Partha Sarathi Goswami, Tamal Chakraborty, Harekrishna Chatterjee, "A Novel Encryption Technique Using DNA Encoding and Single Qubit Rotations", International Journal of Computer Sciences and Engineering, Vol.6, Issue.3, pp.364-369, 2018.

[8]. Tausif Anwar, Abhishek Kumar, Sanchita Paul, "DNA Cryptography Based on Symmetric Key Exchange", International Journal of Engineering and Technology (IJET), Vol.7, No.3, pp.938-950, Jun-July 2015.

[9]. M. Kharbutli, K. Irwin, Y. Solihin, and J. Lee, "Using Prime Numbers for Cache Indexing to Eliminate Conflict Misses," Proc. Int'l Symp. High Performance Computer Architecture, 2004.

[10]. V. Krishnan and J. Torrellas, "A Direct-Execution Framework for Fast and Accurate Simulation of Superscalar Processors," Proc. Int'l Conf. Parallel Architectures and Compilation Techniques, Oct. 1998.

[11]. D. H. Lawrie and C.R. Vora, "The Prime Memory System for Array Access," IEEE Trans. Computers, vol. 31, no. 5, May 1982.

Cite this article as :

K. Ravikumar, R. Raja, "A Study on Data Hiding Technique Using Prime Numbers", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 5 Issue 2, pp. 619-623, March-April 2019. Available at doi : https://doi.org/10.32628/CSEIT1952186
Journal URL : http://ijsrcseit.com/CSEIT1952186