Study on Cryptanalysis of the Tiny Encryption Algorithm

Rajashekarappa, K. M. Sunjiv Soyjaudah, Sumithra Devi K. A.

Abstract—In this paper we present the Study on a Tiny Encryption Algorithm. There is a requirement to specify cryptographic strength in an objective manner rather than describing it using subjective descriptors such as weak, strong, acceptable etc. Such metrics are essential for describing the characteristics of cryptographic products and technologies. Towards this objective, we use two metrics called the Strict Plaintext Avalanche Criterion (SPAC) and the Strict Key Avalanche Criterion (SKAC) mentioned in our study that the strength of popular ciphers such as DES and TEA. A related issue of significance in the context of cryptographic applications is the quality of random number generators which need to pass certain tests. In this Paper, we expose DES and TEA to some of the standard random number generator tests.

Keywords: Data Encryption Standard, Tiny Encryption Algorithm.

I. INTRODUCTION

There is a requirement to specify modern cryptography strength is heavily based on mathematical theory and computer science practice, cryptographic algorithms are designed around computational hardness assumptions, making such algorithms hard to break in practice by any adversary.

The Tiny Encryption Algorithm (TEA) is a block cipher encryption algorithm that is very simple to implement, has fast execution time, and takes minimal storage space [1].

The most widely used encryption scheme is based on the Data Encryption Standard (DES) adopted in 1977 by the National Bureau of Standards, now the National Institute of Standards and Technology (NIST), as Federal Information Processing Standard 46 (FIPS PUB 46). The algorithm itself is referred to as the Data Encryption Algorithm (DEA)[2]. For DES, data are encrypted in 64-bit blocks using a 56-bit key. The algorithm transforms 64-bit input in a series of steps into a 64-bit output. The same steps, with the same key, are used to reverse the encryption [3, 5].

In this Paper, we expose DES and TEA to some of the standard random number generator tests. We consider DES because of its general purpose use in a large number of applications. Our interest in TEA was driven by the need to develop a tiny algorithm for mobile applications. Mobile devices have limited computational power and battery power. So it is difficult for them to perform various compute-intensive operations. On the other hand, increase in applications involving mobile transactions requires that these devices should be capable of using compute-intensive encryption and decryption techniques. In general, higher compute-intensive algorithms provide more security. For example, more the number rounds, better is the security provided by a cryptographic algorithm. Thus there is a trade-off between the number of rounds to be used for encryption and decryption and security strength.

A. Cryptographic Metrics

Cryptographic metrics can be used to quantitatively measure the strength of a particular cryptographic algorithm. There are two kinds of metrics–qualitative and quantitative. In the literature, a lot of qualitative metrics are described. These metrics mainly depend on cryptanalysis of algorithms previously reported. However, using these metrics, it becomes difficult to measure the strength of an algorithm for different number of rounds. Thus, quantitative measures are required to do the same. Though there are not many standard quantitative metrics in the literature, [15] provides discussion on the related issues.

As computer systems become more pervasive and complex, security is increasingly important. Cryptographic algorithms and protocols constitute the central component of systems that protect network transmissions and store data. The security of such systems greatly depends on the methods used to manage, establish, and distribute the keys employed by the cryptographic techniques. Even if a cryptographic algorithm is ideal in both theory and implementation, the strength of the algorithm will be rendered useless if the relevant keys are poorly managed [6,7,8].

B. Quantitative metrics

There are two metrics mentioned in [15] which can be computed. These are Strict Plaintext Avalanche Criterion (SPAC) and Strict Key Avalanche Criterion (SKAC). Avalanche refers to a spreading effect. Dictionary meaning of Avalanche is “flood”. In terms of a cryptographic algorithm, it refers to a large change in the output due to a small change in the input. For a better cryptographic algorithm, avalanche effect should be higher. This is because, higher the avalanche effect, more difficult it is to trace the effect. of an individual bit in the input, making it harder for cryptanalysis.

Now we will discuss what SPAC and SKAC mean. SPAC refers to the number of bit-changes in the output corresponding to one bit change in the plaintext. Similarly, SKAC refers to the number of bits of the ciphertext that change after a bit in the key is changed.

For a fixed key to satisfy the SPAC, each bit of the ciphertext block should change with a probability of one half, whenever any bit of the plaintext block is complemented. For key changes, the algorithm satisfies the SKAC if, for a fixed plaintext block, each bit

Revised Manuscript Received on February 06, 2013.

Rajashekarappa, Dept. of Computer Science and Engineering, JSSATE, Mauritius. (Research Scholar, Jain University, Bangalore, Karnataka, India)

Dr. K M Sunjiv Soyjaudah, Dept. of Electrical and Electronic Engineering,, University of Mauritius, Reduit, Mauritius.

Dr. Sumithra Devi K A, Director, Department of Master of Computer Application, R. V. College of Engineering, Bangalore, India.

of the ciphertext block changes with a probability of one half when any bit of the key changes.

The rest of the paper is organized as follows: Section II presents the literature review. Section III gives a brief overview of Tiny Encryption Algorithm. Experimental results are discussed in Section IV. Section V concludes the paper and Future works.

II. RELATED WORK

Tiny Encryption Algorithm was designed David Wheeler and Roger Needham of the Cambridge University Computer Laboratory and presented in November 1994. It is designed to encrypt data through many iteration cycles rather than memory consuming arrays of data. TEA [1] is a short algorithm which will run on most machines and encipher safely.

In 1994, the cipher Tiny Encryption Algorithm is a 64-round Feistel cipher that operates on 64-bit blocks and uses a 128-bit key. Designed by Wheeler and Needham, it was presented at FSE 1994 [13]. Noted for this simple design, the cipher was subsequently well studied and came under a number of attacks.

In 1996, Kelsey et al. established that the effective key size of TEA was 126 bits [2]. This result led to an attack on Microsoft's Xbox gaming console where TEA was used as a hash function [12].

In 1997. Kelsey, Schneier and Wagner constructed a related-key attack on TEA with 2^{23} chosen plaintexts and 2^{32} time [4]. Following these results, TEA was redesigned by Needham and Wheeler to yield Block TEA and XTEA (eXtended TEA) [10]. While XTEA has the same block size, key size and number of rounds as TEA, Block TEA caters to variable block sizes for it applies the XTEA round function for several iterations. Both TEA and XTEA are implemented in the Linux kernel[11].

In 1998. To correct weaknesses in Block TEA, Needham and Wheeler designed Corrected Block TEA or XXTEA, and published it in a technical report [11]. This cipher uses an unbalanced Feistel network and operates on variable length messages. The number of rounds is determined by the block size, but it is at least six. An attack on the full Block TEA is presented in [11], where some weaknesses in XXTEA are also detailed.

In 2002–2010. A number of cryptanalysis results on the TEA family were reported in this period. Table 1 lists the attacks on XTEA and their complexities. In [14], it was shown that an ultra-low power implementation of XTEA might be better suited for low resource environments than AES. Note that XTEA's smaller block size also makes it advantageous if an application requires fewer than 128 bits of data to be encrypted at a time[9,10].

The Data Encryption Standard (DES) is a previously predominant algorithm for the encryption of electronic data. It was highly influential in the advancement of modern cryptography in the academic world. Developed in the early 1970s at IBM and based on an earlier design by Horst Feistel, the algorithm was submitted to the National Bureau of Standards (NBS) following the agency's invitation to propose a candidate for the protection of sensitive, unclassified electronic government data. In 1976, after consultation with the National Security Agency (NSA), the NBS eventually selected a slightly modified version, which was published as an official Federal Information Processing Standard (FIPS) for the United States in 1977. The publication of an NSA-approved encryption standard simultaneously resulted in its quick international adoption and widespread academic scrutiny. Controversies arose out of classified design elements, a relatively short key length of the symmetric-key block cipher design, and the involvement of the NSA, nourishing suspicions about a backdoor. The intense academic scrutiny the algorithm received over time led to the modern understanding of block ciphers and their cryptanalysis [3, 12, 16, and 17].

DES is now considered to be insecure for many applications. This is chiefly due to the 56-bit key size being too small; in January, 1999, distributed.net and the Electronic Frontier Foundation collaborated to publicly break a DES key in 22 hours and 15 minutes. There are also some analytical results which demonstrate theoretical weaknesses in the cipher, although they are infeasible to mount in practice. The algorithm is believed to be practically secure in the form of Triple DES, although there are theoretical attacks. In recent years, the cipher has been superseded by the Advanced Encryption Standard (AES). Furthermore, DES has been withdrawn as a standard by the National Institute of Standards and Technology (formerly the National Bureau of Standards).

In [21] we proposed the Heuristic Search Procedures for Cryptanalysis and Development of Enhanced Cryptographic Techniques" as mentioned the comparative result for Tabu search, genetic algorithm and simulated annealing.

III. TINY ENCRYPTION ALGORITHM (TEA)

Tiny Encryption Algorithm is a short algorithm which will run on most machines and encipher safely [1]. It uses a large number of iterations rather than a complicated program. It uses very little setup time and does a weak non -linear iteration enough rounds to make it secure. There are no preset tables or long setup times. The 128-bit key K is split into four 32-bit blocks.

Following is the program in C, for encoding with key k [0] – k [3]; data is assumed to be in v [0] and v[1].

```
void code (long* v, long* k)
 {
   unsigned long y=v[0],z=v[1], sum=0,   /* set up */
   delta=0x9e3779b9, n=32 ;      /* a key schedule constant */
   while (n-->0) {                  /* basic cycle start */
   sum += delta ;
    y += (z<<4) + k[0] ^ z+sum ^ (z>>5)+k[1] ;
    z += (y<<4) + k [2] ^ y+sum ^ (y>>5)+k[3] ; /* end cycle
*/
 }
v[0]=y ; v[1]=z ; }
```

Using the same routine, and also a similar routine for decryption, TEA was implemented. The implementation was made flexible in order to change the number of rounds used for encryption and decryption

IV. EXPERIMENTAL RESULTS

Strict Plaintext Avalanche Criterion (SPAC):

As described before, SPAC relates to bit changes in ciphertext corresponding to a single bit change in the plaintext. During experiments, a block of plaintext P was taken. For a key K, ciphertext

C was obtained by applying the TEA algorithm. Now, plaintext P' was obtained by complementing the first bit of the original plaintext P. This new plaintext block P' is now encrypted using the same key K to obtain the new ciphertext C'. Now the number of bits at which C and C' differ is calculated. This number is further divided by 64 (each block is of 64-bit length) to obtain the probability of bit-change. Let's call this probability as bit-change probability. The above experiment is repeated for different P'. There are 64 possible combinations of P', each obtained by flipping one of the 64 bits of the original plaintext block P. The experiment was performed for all the 64 possibilities of P'. The average of all the 64 bit-change probabilities was obtained. The expected value of this probability for a secure algorithm is 0.5.

These entire experiments were repeated for 1 to 32 rounds of TEA. The X-axis corresponds to the number of rounds used for encryption and decryption. The Y-axis shows the bit-change-probability deviation from the expected value of 0.5. As expected, the avalanche effect goes closer to the expected value of 0.5 as the number of rounds increases. SPAC seems to be a fair metric though it is difficult to say that it is the best metric. As can be seen from the graph, for rounds 0, 1, 2 and 3, TEA does not produce enough avalanche effect. After four rounds, the avalanche effect approaches the expected value of 0.5 and there is no significant change in the avalanche effect.

Experiments and Results for DES:

DES is a 64-bit block cipher with 56-bit key. Unlike TEA, we have two parameters to change in case of DES. We can change the number of rounds as we did in TEA. Also, we can remove SBOXes – arguably the most complicated part of the DES algorithm. SBOXes are also the ones which add non-linearity to the DES algorithm. It is expected that security of the algorithm should decrease drastically once SBOXes are removed[18]. With S-BOXes: As described before, SPAC relates to bit changes in the ciphertext corresponding to a single bit change in the plaintext. During experiments, a block of plaintext P was taken. For a key K, ciphertext C was obtained by applying the DES algorithm with SBOXes.

Now, plaintext P' was obtained by complementing the first bit of the original plaintext P. This new plaintext block P' is now encrypted using the same key K to obtain the new ciphertext C'. The number of bits at which C and C' differ is calculated. This number is further divided by 64 (each block is of 64-bit length) to obtain the probability of bit-change. The above experiment is repeated for different P'. There are 64 possible combinations of P', each obtained by flipping one of the 64 bits of the original plaintext block P. The experiment was performed for all the 64 possibilities of P'. The average of all the 64 bit-change probabilities was taken. The expected value of this probability for a secure algorithm is 0.5. These experiments were repeated for 1 to 16 rounds of DES with SBOXes[19,20].

V. CONCLUSION

Testing the random numbers is very important in all cryptographic applications. In the absence of generally accepted norms to be used to measure and specify cryptographic strength, it is desirable to carry out a number of tests on different ciphers to get an exposure to their strength and weakness. With this objective, we carried out a variety of randomness and security strength tests which are presented in this paper. The strength of TEA and DES emerges quite significantly. Also as a heuristic technique to be used in cryptanalysis, Tabu search performs better than genetic algorithm and simulated annealing, based on the randomness of the numbers generated from these algorithms [21].

REFERENCES

1. Wheeler D., and R. Needham. TEA, a Tiny Encryption Algorithm, Proceedings of the Second International Workshop on Fast Software Encryption, Springer-Verlag, 1995, pp. 97-110.
2. J. Kelsey, B. Schneier, D. Wagner, "Key-Schedule Cryptoanalysis of IDEA, G-DES, GOST, SAFER, and Triple-DES," CRYPTO 1996 (N. Koblitz, ed.), vol. 1109 of LNCS, pp. 237–251, Springer-Verlag, 1996.
3. William Stallings. *Cryptography and Network Security Principles and Practices*, Third Edition, Pearson Education Inc., 2003.
4. D. Wheeler and R. Needham, TEA Extensions, October 1997.
5. J. Kelsey, B. Schneier and D. Wagner,Related-Key Cryptanalysis of 3-WAY, Biham-DES, CAST, DES-X, NewDES, RC2, and TEA, In Information and Communica-tions Security-Proceedings of ICICS 1997, Lecture Notes in Computer Science 1334,Springer-Verlag, 1997.
6. Y. Ko, S. Hong, W. Lee, S. Lee, J.-S. Kang, "Related-Key Differential Attacks on 27 Rounds of XTEA and Full-Round GOST," FSE 2004 (B.K. Roy, W. Meier, eds.), vol. 3017 of LNCS, pp. 299–316, Springer-Verlag, 2004.
7. E. Lee, D. Hong, D. Chang, S. Hong, J. Lim, "A Weak Key Class of XTEA for a Related-Key Rectangle Attack," VIETCRYPT 2006 (P.Q. Nguyen, ed.), vol. 4341 of LNCS, pp. 286–297, Springer-Verlag, 2006.
8. J. Lu, "Related-key rectangle attack on 36 rounds of the XTEA block cipher," International Journal of Information Security, vol. 8(1), pp. 1–11, Springer-Verlag, 2009, also available at http://jiqiang.googlepages.com/IJIS8.pdf.
9. M. D. Moon, K. Hwang, W. Lee, S. Lee, J. Lim, "Impossible Differential Cryptanalysis of Reduced Round XTEA and TEA," FSE 2002 (J. Daemen, V. Rijmen, eds.), vol. 2365 of LNCS, pp. 49–60, Springer-Verlag, 2002.
10. R. M. Needham, D.J. Wheeler, "Correction to xtea," technical report, Computer Laboratory, University of Cambridge, October 1998, available at http://www.movable-type.co.uk/scripts/xxtea.pdf.
11. M.-J. Saarinen, "Cryptanalysis of Block TEA," unpublished manuscript, October 1998, available at http://groups.google.com/group/sci.crypt.research/msg/f52a533d1e2fa15e.
12. M. Steil, "17 Mistakes Microsoft Made in the Xbox Security System," Chaos Communication Congress 2005, available at http://events.ccc.de/congress/2005/fahrplan/events/559.en.html.
13. D.J. Wheeler, R.M. Needham, "TEA, a Tiny Encryption Algorithm," FSE 1994 (B. Preneel, ed.), vol. 1008 of LNCS, pp. 363–366, Springer-Verlag, 1994.
14. J.-P. Kaps, "Chai-Tea, Cryptographic Hardware Implementations of xTEA," INDOCRYPT 2008 (D.R. Chowdhury, V. Rijmen, A. Das, eds.), vol. 5365 of LNCS, pp. 363–375, Springer-Verlag, 2008.
15. Norman D., Jorstad., and Landgrave T. Smith, Jr. Cryptographic Algorithm Metrics, Technical Report, Institute for Defense Analyses, Science and Technology Division, Jan 1997.
16. Bruce Schneier. *Applied Cryptography, 2nd Edition*, John Wiley & Sons, 1996.
17. Rajashekarappa, Sunjiv Soyjaudah, K. M.,"Overview of Differential Cryptanalysis of Hash Functions using SMART Copyback for Data" published at International Journal of Computer Science and Technology(IJCST), Vol. 4, Issue 1, Jan-March 2013, pp 42-45.
18. Behrouz A. Forouzan, (2006)"Cryptography and Network Security", Firstedition, McGraw- Hill.
19. Atul Kahate, " Cryptography and Network Security", TMH, 2003.
20. A.Zugaj, K. Górski, Z. Kotulski, A. Paszkiewicz, J. Szczepański,(2000) "New constructions in linear cryptanalysis of block ciphers", ACS'2000, October.
21. Rajashekarappa and Dr. K M S Soyjaudah "Heuristic Search Procedures for Cryptanalysis and Development of Enhanced Cryptographic Techniques" Published at International Journal of Modern Engineering Research (IJMER), May 2012, Vol.2, Issue.3, pp-949-954.

AUTHORS PROFILE

Mr. Rajashekarappa working as a Lecturer since July 2010 in the Department of Computer Science and Engineering, JSS Academy of Technical Education, Avenue Droopnath Ramphul, Bonne Terre, Vacoas, Mauritius. He has one and half years of experienced as a Project Assistant at Indian Institute of Science (IISc), Bangalore, India. He worked as Project Internee in Indian Space Research Organization (ISRO), Bangalore, India. He has one and half years of experienced as a Project Trainee at LSI Technologies Pvt. Ltd, Bangalore, India.

Mr. Rajashekarappa obtained his Bachelor of Engineering in Computer Science and Engineering from Anjuman Engineering College, Bhatkal, India. He has qualified in Graduate Aptitude Test in Engineering (GATE), Computer Science and Engineering, 2006. He received his Master Degree in Computer Science and Engineering from R. V. College of Engineering, Bangalore, India. He is pursuing his Ph. D in Computer Science and Engineering at Jain University, Bangalore, India. Mr. Rajashekarappa area of interest and research include Cryptography, Data mining, Mobile Communication, Computer Networks and Cloud Computing. He has published several Research papers in international journal/conferences. He has guided many students of Bachelor degree in Computer Science and Engineering in their major projects. Mr. Rajashekarappa is a member of ISTE, IETE, IACSIT, IAEST, IAENG and AIRCC.

Professor K M Sunjiv Soyjaudah received his B. Sc (Hons) degree in Physics from Queen Mary College, University of London in 1982, his M.Sc. Degree in Digital Electronics from King's College, University of London in 1991, and his Ph. D. degree in Digital Communications from University of Mauritius in 1998. He is presently Professor of Communications Engineering in the Department of Electrical and Electronic Engineering of the University of Mauritius. His current interest includes source and channel coding modulation, cryptography, voice and Video through IP, as well as mobile communication. Professor K M Sunjiv Soyjaudah is a member of the IEEE, Director in the Multicarrier (Mauritius), Technical Expert in the Energy Efficiency Management Office, Mauritius. Registered Ph.D Guide in University of Mauritius, Reduit, Mauritius and Jain University, Bangalore, Karnatka, India.

Dr. Sumithra Devi K A, Professor and Director, in Master of Computer Applications at R V College of Engineering, Bangalore, India. She received B.E. from Malnad College of Engineering, Hassan. She received M.E and Ph D from UVCE, Bangalore and Avinashilingam University for Women, Coimbatore, India respectively. Reviewer for many International Journals/ Conferences like WEPAN, WICT, EDAS, IACSIT, ISCAS, JEMS. Published 14 journals and 65 International/ National Conferences. Professional Member in many IEEE, IETE, CSI, ISTE. Member in BoS and BoE, for Visvesvaraiah Technological University, Belgaum, Karnataka. Registered PhD Guide in Visvesvaraiah Technological University, Belgaum; Jain University, Bangalore; Prist University, Sathyabhama University, Tamilnadu. Authored a chapter "CAD algorithm for VLSI design" in the book "VLSI Design ", published by In-Tech Publications, ISBN 979-953-307-512-8, 2011, and authored book on Operating System, published by Shroff Publisher India.