Breaking des (data encryption systems)

Science, Computer Science

Data Encryption Standard (DES) is an algorithm for decrypting and encrypting unstipulated information in the United States administration standard. DES is derived from IBM’s Lucifer code and is depicted by the Federal Information Processing Standards (FIPS) 46, with its current modification being FIDS 46-3 (Conrad, 2007). DES is a mass code that takes a plaintext sequence as a key in and generates a code transcript wording of the same measurement lengthwise.

The mass of the DES obstruct is 64 bits which is also the same for the input dimension even though the 8 bits of the key are for the recognition of faults making the efficient DES input amount 56 bits. Because of the progressions in the authority of dispensation in workstations there are weaknesses in the 56-bit key extent presently (Conrad, 2007). In the company of proper hardware, there is a best chance assault on methodical efforts to all the 72 quadrillion hence, there is a possibility of dissimilar inputs.

Advanced Encryption Standard (AES) developed into an innovative FIPS-standard encryption average in 2001, 26th November to replace DES. Statistics Encryption Algorithm explains the definite algorithm as contested to the average. In such circumstances, TDEA is a short form for Triple DES. At the same time, there is a description of Triple Data Encryption Algorithm Modes of Operation ANSI X9. 52-1998 (Clayton & Bond, 2002). History of DES DES was proposed in 1975 and approved in 1977 as a federal information processing standard. It was criticized by the people who felt that it’s 56 key lengths to be insecure.

In spite of this, DES remained a strong encryption algorithm until mid 1990. In the year 1998 summer, the insecurity of DES was demonstrated when a $

250, 000 computer which was built by the electronic frontier foundation decrypted a DES-encoded message in 56 hours. This was improved in the 1999 to 2002 hours through a combination of 100, 000 networked personal computers and the EFF machine. DES remains a de facto standard unless a substitute is found (Landau, 2000, p. 341). A certified DES is obtained from the National Institute of Standards andTechnology(NIST).

This Advanced Encryption Standard (AES) works in three key lengths: 128, 192, and 256 bits. The publication of DES indicated a new era in cryptography. The development in the community of public cryptographers was enhanced by having an algorithm availability of study that the national security agent certified to be secure (Landau, 2000, p. 341). The (DES) Data Encryption Standard A system that encrypts quickly but is essentially what is impossible to break is all what cryptographers have always wanted. Public key systems have captured the imagination of mathematicians because of their reliance on elementary number theory.

Public key algorithms are used for establishing a key because they are too slow to be used for most data transmissions. Private key system does the encryption because they are typically faster than public key ones (Landau, 2000, p. 341). The data Encryption Standard (DES) workhorse uses private key algorithm besides relying on cryptographic design principles that predate public key. The RC4 in web browsers and the relatively insecure cable TV signal encryption are an exception to DES. DES is the most widely used public cryptosystem in the world. It is the cryptographic algorithm which is used by banks for electronic funds transfer.

It is also used for the protection of civilian satellite communications. Still, a variant of DES is used for UNIX password protection. There are three operation of the DES which involves XOR, substitution and permutation. The DES is an interrelated block cipher and a cryptosystem on a block of symbols that sequentially repeats an internal function which is called a round. It encrypts data by the use of a primitive that operates on a block of symptoms of moderate size. Self invert ability is also essential to enable one of the objects to encrypt and decrypt. When encrypting ordinary text, DES begins by grouping the text into 64 bit block.

A number of operations are performed by the DES on each block (Landau, 2000, p. 343). The transformation of how the block is to be carried out is determined by a single key of 56 bits. DES iterates sixteen identical rounds of mixing; each round of DES uses a 48-bit sub key. The DES begins with an initial permutation P and ends with its inverse. The permutations are of minor cryptographic implications but forms part of the official algorithm. The selection of sub keys starts by splitting the 56-bit key into two 28-bit halves and rotating each half one or two bits; either one bit in rounds 1, 2, 9, and 16 or two bits otherwise.

The two halves are put back together and then 48 particular bits are chosen and put in order (Landau, 2000, p. 343). Attacks of DES The selection of DES was followed by protests in which case some of the researchers appeared to object to the algorithm small key space. Investors in the key public cryptography claimed that a DES encoded message could be broken in about a day by a $ 20 million machine made up of a million specially designed VLSI capable of searching one key per microsecond while working in parallel.

The use of a meet in the middle attack to break a four round version of DES did not extend past seven rounds (Landau, 2000, p. 345). This is evidence that, for all these attacks none of them posed a serious threat to the DES. Other attacks on the DES were performed to poke harder to the innards of DES. This brought anomalies which led to the first attacks that were seen to be more theoretically better than exhaustive search. The attacks were against the block structure system and the need of all block-structured cryptosystems needed to be designed to be secure against differential and linear cryptanalysis.

There is a strong attack to DES which is differential cryptanalysis. This is apparently known to the algorithms designers. In order to design a secure cryptosystems, there is a need for a mixture of well known principles, some theorems and the presence of some magic. Attacks on a cryptosystem fall into two categories which are passive attacks and active attacks. The passive attacks are the ones which adversely monitors thecommunicationchannel. They are usually easier to mount although they yield less. The active attacks have the adversary transmitting messages to obtain information (Landau, 2000, p.

342). The aim of the attackers is to determine the plaintext from the cipher text which they capture. A more successful attack will determine the key and thus compromise a whole set of messages. By designing their algorithms, cryptographer’s help to resist attacks such as cipher text only attack whose adversary has access to the encrypted communications. The known plain text attack which has its adversary has some plain text and its corresponding cipher text. The third attack which can be avoided is the

chosen text attack and its adversary chooses the plain text for encryption or decryption.

The plain text chosen by the adversary depends on the cipher text received from the previous requests (Landau, 2000, p. 342). Observations about DES The simplicity found in the DES amounts to some fully desirable properties. To start with it is the complementation. To illustrate, allow X to denote the bitwise complement of X. If C is the DES encryption of the plaintext P with key K, then P is the DES encryption of P with key K. In some cases the complementation can simplify DES cryptanalysis by basically cutting the investigating space in half.

These properties do not cause serious weakness in the algorithm. The set generated by the DES permutations do not form a group. The group may have at least 102499 elements. There is strength in the DES when it lacks a group structure. It appears to be double encryption where this is twice by two different keys, EK2 (EK1 (P) and is not stronger than single encryption. The reason is that when meeting in the middle attacks for a given plaintext cipher text pair, an adversary will compute all 256 possible enciphering of the plaintext i. e.

EKi (P), and indexes the same. The adversary will then compute all possible deciphering of the cipher text (Landau, 2000, p. 345). Models of DES There are four forms of DES, which are accepted by FIPS 81. They include (ECB) Electronic Codebook form, code mass sequence form (CFB), productivity reaction form (OFB) and system response (CFB). The forms are used to with both DES and Triple DES. Within each form, there are main dissimilarities

which are based on the fault proliferation and obstruct vs. tributary codes (Conrad, 2007). Electronic Codebook (ECB) Mode

In this form of encryption, there is sovereign encryption into respective blocks of codes text. It is done by means of Feistel code which generates 16 sub-inputs derived from the symmetric input and also encrypts the plaintext using 16 surroundings of conversion. Similarly, the development is used in the conversion of code text reverse into simple text with the dissimilarity that, 16 sub inputs are contributed in overturn arrangement. The result of repeated blocks of identical plaintext is the repeated blocks of cipher text which is capable of assisting in the vault investigation of the code wording.

In Appendix 1 there is an illustration of the result (Conrad, 2007). The first picture of SANS symbol is the bitmap layout. The second picture is the encrypted logo of SANS bitmap via DES ECB form. The visibility of the model is due to the recurring of masses of the simple wording pixels in the bitmap which are encrypted into masses which are repeated and are of particular code pixels. In this form, faults do not proliferate due to the autonomous encryption of each obstruct. Cipher Block Chaining (CBC) Mode

The CBC form is an obstruct code which XORs every original obstruct of simple wording with the previous block of code wording. This indicates that repeated obstructs of simple wording do not give rise to repeated obstructs of code wording. CBC uses a vector of initialization which is an arbitrary original obstructs used to make sure that two simple wordings result in different code wordings. In figure 2 of the Appendix there is a clear illustration of the same SANS symbol bitmap data, encrypted with DES CBC

form. There is no visibility of any prototype which is true for all DES forms apart from ECB.

Therefore, in this mode, there is proliferation of faults as each prior step’s encrypted output is XORed with the original obstructing of simple wording (Conrad, 2007). Cipher Feedback (CFB) Mode The Cipher Feedback Mode is a tributary code that encrypts simple wording by breaking into X (1-64) bits. This permits encryption of the level of byte or bits. This mode uses an arbitrary vector of initialization. The preceding elements of code wording are XORed with consequent components of code wording. Therefore, in this mode of CBC there is proliferation of faults (Conrad, 2007).

Output Feedback (OFB) Mode Similar to CFB form, the productivity reaction form makes use of the vector of random initialization and also encrypts simple wording by shattering downward into a tributary by encrypting components of X (1-64) bits of simple wording. This form fluctuates from CFB form by generating a simulated-arbitrary tributary of productivity which is XORed with the plaintext during every step. Therefore, the productivity is fed back to the simple wording and because the output is XORed to the simple wording, faults there is no proliferation of mistakes (Conrad, 2007).

Counter (CTR) Mode The oppose form is a tributary code similar to OFB form. The main disparity is the accumulation of contradict obstructs. The offset can be supplementary to an arbitrary importance that is used only once and then increased for each component of simple wording that is encrypted. The initial counter obstructs acts as a vector of initialization. Therefore, in each surrounding there is XORing of the offset obstructs with simple wording.

Accumulation of offset obstructs permits disintegration of encryption into equivalent phases, improving presentation on a suitable hardware.

There is no proliferation of mistakes (Clayton & Bond, 2002). (Table 1 in the Appendix summarizes the Data Encryption Standard). Triple DES (T DES) In anticipation of 2030, TDES can be used as FIPS encryption algorithm which is permitted in order to allow conversion to AES. There are three surroundings of DES which are used by TDES which have an input extent of 168 bits (56 * 3). There is a possibility of reduced effective key length of TDES to roughly 12 bits though beast might assaults against TDES re not realistic at present (Conrad, 2007).

Architecture for Cryptanalysis All modern day practical ciphers both symmetrical and asymmetrical make use of security apparatus depending on their key length. In so doing, they provide a margin of security to cover from computational attacks with present computers. Depending on the level of security which is chosen for any software application, many ciphers are prone to attacks which unique machines having for instance a cost-performance ratio (Guneysu, 2006).

Reconfigurable computing has been recognized as way of reducing costs while also acting as an alternative to a variety of applications which need the power of a custom hardware and the flexibility of software based design such as the case of rapid prototyping (Diffie & Hellman, 1977, pp. 74-84). What this means is that cryptanalysis of today’s cryptographic algorithms need a lot of computation efforts. Such applications map by nature to hardware based design, which require repetitive mapping of the main block, and is easy to extend by putting in place additional chips as is needed.

However, it should be noted that the mere presence of resources for computation is not the main problem. The main problem is availability of affordable massive computational resources. The non-recurring engineering costs have enabled hardware meant for special purpose cryptanalysis in virtually all practicable situations unreachable. This has been unreachable to either commercial or research institutions, which has only been taken by government agencies as feasible (Diffie & Hellman, 1977, pp. 74-84).

The other alternative to distributed computing with loosely coupled processors finds its base on the idle circles of the large number of computers connected through the internet. This method has considerably been successful for some applications. However, the verified detection of extraterrestrial life is considerably still a problem more so for unviable problems with power of computing in a particular organization (Guneysu, 2006). In cryptanalysis some algorithms are very suitable for special-purpose hardware.

One main example for this is the search for the data encryption standard (DES) (FIPS, 1977). What this means is that a brute- force attack is more than twice the magnitude faster when put in place on FPGA’s as opposed to in software on computers meant for general purposes at relatively the same costs (FIPS, 1977). That notwithstanding, for many crypto algorithms the advantages due to cost-performance of hardware meant for special purposes over those meant for ordinary purposes is not really as dramatic as is usually the case of DES, more so for public-key algorithms (Guneysu, 2006).

Arising from the advent of low-cost FPGA families with much logic approaches recently, field programmable gate arrays offer a very interesting

way for the thorough computational effort which cryptanalysis needs (Lesnsta & Verheul, 2001, pp. 255-293). Many algorithms dealing with the most important problems in cryptanalysis is capable of being put in place on FPGAs. Code breaking though, requires more additional efforts as opposed to just programming a single FPGA with a certain algorithm (Electronic Frontier Foundation, 1998).

Owing to the enormous perspectives of cryptanalysis problems, many more resources as opposed to FPGA are needed. This implies that the main need is massively powerful parallel machinery suited to the requirements of targeted algorithms. Many problems are capable of being put in parallel and are perfectly suited for an architecture distributed. Conventional parallel architectures for computing can theoretically be used for applications of cryptanalysis (Guneysu, 2006). An optical Architecture to Break Ciphers The targeted DES brute force attack has several characteristics.

To begin with, expensive computational operations which are put in parallel. Next, there is no need of communication between single parallel instances. The next characteristic is the fact that the general expense for communication is not high owing to the fact that the stage of computation strongly outweighs the data input and output stages. According to Blaze et al, (1996), communication is almost entirely used for results reporting as well as initialization. A central control instance with regards to communication is capable of being accomplished by a conventional low cost personal computer, connected simply by an interface.

This would imply that there is no need for a high-speed communication interface. The fourth characteristic is the fact that a DES brute-force attack

and its following implementation require little memory. The final consequence of the above is the fact that the available memory on present day low cost FPGAs is sufficient (Guneysu, 2006). What this implies is that by making use of low-cost FPGAs, it is possible to develop a cost effective dynamic architecture which is capable of being reprogrammed which would be able to accommodate all the targeted architectures (Blaze et al, 1996).

Realization of COPACOBANA Drawing back, the Cost-Optimized Parallel Code Breaker (COPACOBANA) meeting the needs available comprise of several independent-low prized FPGAs, connected to a hosting PC by way of a standard interface such as a USB. Moreover, such a standard interface permits to extend a host-PC with more than one device of COPACOBANA. The initialization of FPGAs, the control as well as the process of results accumulation is carried out by the host. Critical computations are carried out by the FPGAs, which meet the actual cryptanalytical architecture (Schleiffer, 2006).

Developing a system of the above speculations with FPGA boards which are commercially available is certainly possible but at a cost. Therefore it is important to put into considerations the design and layout among others in coming up with the above kind of system (Schleiffer, 2006). This would therefore mean that our cost-performance design meant for cost optimization is only capable of being achieved if all functionalities are restricted to those required for code breaking. Arty the same time, many designs choices should be based on components and interfaces which are readily available (Guneysu, 2006).

Conclusion In conclusion, cryptanalysis of symmetric and asymmetric ciphers is extremely demanding in terms of computations. It would be fair to hold the belief that breaking codes with conventional PCs as well as super-computers is very much costly. Bit-sizes of keys should be chosen in a way that traditional methods of code breaking do not succeed (Rouvroy et al 2003, pp. 181-193). This would mean that the only way to go through ciphers is to develop special-purpose hardware purposely meant for suitable algorithms.

In the final analysis, traditional parallel architecture in the end equally appears to be too complicated and therefore not cost saving in finding solutions to cryptanalytical problems. As earlier observed, many of these problems can easily be put in parallel implying that the algorithms which correspond to them are equally capable of being parameterized to lower communication costs (Guneysu, 2006). A hardware architecture which is cost effective (COPACOBANA) is the end product of the algorithmic requirements of the intended problems of cryptanalysis.

This work represents not only the design but also the first prototype of an effective design which meets the demands of the request. In the final analysis, COPACOBANA would be able to accommodate as many as 120 FPGAs which are less costly. At the same time, it is possible to break data encryption standard (DES) within a period of nine days. This would require a hardware design comprising of reprogrammable logic which could be adopted to accommodate any task, even those not necessarily in line with code breaking (Rouvroy et al 2003, pp. 181-193). References Blaze, M.. , Diffie, W. , Rivest, R. L.

, Scheiner, B. , Shimomura, E. , and Weiner, M (1996). Minimal Key Lengths for Symmetry Ciphers to Provide Adequate Commercial Security. Ad Hoc Group of Cryptographers and Computer Scientists. Retrieved from December, 13, 2008 from http://www. counterpane. com/keylength. html. Clayton, R. and Bond, M. (2002). Experience Using a Low-Cost FPGA Design to Crack DES Keys. In B. S. Kaliski, C. K. Koc Cetin, and C. Paar, editors, Cryptographic Hardware and Embedded Systems - CHES 2002, 4th International Workshop, Redwood Shores, CA, USA, volume 2523 of series, pages 579 – 592. Springer-Verlag. Conrad, E. (2007).

Data Encryption Standard, The SANS Institute Diffie, W & Hellman, M. E. (1977). Exhaustive cryptanalysis of the NBS Data Encryption Standard. Computer, 10(6): 74-84 Electronic Frontier Foundation. (1998). Cracking DES: Secrets of Encryption Research, Wiretap Poolitics & Chip Design. O’Reilly & Associates Inc. Federal Information Processing Standard. (1977). Data Encryption Standard, U. S Department of Commerce. Guneysu, T. E. (2006). Efficient Hardware Architecture for Solving the Discrete Logarithm Problem on Elliptic Curves. AAmasters thesis, Horst Gortz Institute, Ruhr University of Bochum. Landau, S.

(2000). Standing the Test of Time: The Data Encryption Standard vol. 47, 3, pp. 341-349. Lenstra, A and Verheul, E. (2001). Selecting Cryptographic Key Sizes. Journal of Cryptology, 14(4): 255–293. Rouvroy, G. , Standaert, F. X. , Quisquater, J. , and Legat, D. (2003). Design Strategies and Modified Descriptions to Optimize Cipher FPGA Implementations: Fast and Compact Results for DES and Triple-DES. In Field-Programmable Logic and Applications- FPL, pp. 181-193 Schleiffer, C. (2006). Design of Host Interface

for COPACOBANA. Technical report, Studienarbeit, Host Gortz Institute, Ruhr University Bochum