IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 5, Issue 10, October 2016

A Study on Cryptography and its Algorithms

H.M. Kantharaju[1], Dr. M Srinivas[2]

Lecturer (Selection Guide) Dept of CSE, Govt. Women's Polytechnic Ramanagara, Karnataka, India[1]

Dept. of CSE, Head CSE, Head T&P, JKC Co-ordinator, TKR Engineering College, Hyderabad, India[2]

Abstract: The traditional model of cryptography examines the security of the cipher as a mathematical function. However, ciphers that are secure when specified as mathematical functions are not necessarily secure in real-world implementations. The physical implementations of ciphers can be extremely difficult to control and often leak so called side-channel information. Side-channel cryptanalysis attacks have shown to be especially effective as a practical means for attacking implementations of cryptographic algorithms on simple hardware platforms, such as smart-cards.

Keywords: Cryptography, Implementation, Security.

1. INTRODUCTION

Adversaries can obtain sensitive information from side-channels, such as the timing of operations, power consumption and electromagnetic emissions. Some of the attack techniques require surprisingly little side-channel information to break some of the best known ciphers. In constrained devices, such as smart-cards, straightforward implementations of cryptographic algorithms can be broken with minimal work. Preventing these attacks has become an active and a challenging area of research.

Hundreds of millions of cryptographic devices, the vast majority being smart-cards, are used today in a variety of applications. These cards execute cryptographic computations based on the secret key stored in their memories. The goal of an attacker is to extract the secret key from a tamper-resistant card in order to modify its content, create duplicate cards or perform an unauthorized transaction.

Kocher et al. described two types of attacks: simple power analysis (SPA) and differential power analysis (DPA). Basic to these attacks is the observation that the power consumed by the cryptographic device (in this case the smart-card) at any particular time during the cryptographic operation is related to the instruction being executed and to the data being processed.

One of the ideas to prevent the timing attack on the square-and-multiply algorithm was to pad the code with dummy computations, such as empty loops. Kocher et al. noticed that the power consumption of these dummy computations was different from the power consumption of meaningful ones.

By simply observing the power traces obtained from the RSA coprocessor, they were able to determine which operations were performed, what enabled them to disclose the secret exponent. This is the basis of simple power analysis.

Probably the most threatening and well studied side-channel attack is the DPA attack. The DPA attack exploits the characteristic behavior of transistor logic gates and software running on today's smart-cards and other cryptographic devices. The attack is performed by monitoring the electrical activity of a device, and then using advanced statistical methods secret information (such as secret keys and user PINs) stored in the device is determined.

Far from being a theoretical attack DPA has been successfully carried out on a wide range of existing cryptographic devices and, therefore, represents a real threat to the security of modern cryptographic systems. What makes the DPA attack especially dangerous is the fact that it is inexpensive to perform (using cheap and readily available equipment) and most implementations are vulnerable, unless specific countermeasures are in place.

The degree of security these countermeasures provide can be different, but any countermeasure is valuable because it increases the cost and the complexity of performing the attack. The complexity of power analysis attacks can be increased by introducing software (algorithmic) and hardware (physical) countermeasures.

Power analysis is a successful cryptanalytic technique that extracts secret information from cryptographic devices by analysing the power consumed during their operation. A particularly dangerous class of power analysis, differential power analysis (DPA), relies on the correlation of power consumption measurements. It has been proposed that adding non-determinism to the execution of the cryptographic device would reduce the danger of these attacks. It has also been demonstrated that asynchronous logic has advantages for security-sensitive applications.

Non-deterministic execution is achieved by exploiting concurrent execution of instructions both with and without data-dependencies; and by forwarding register values between instructions with data-dependencies using randomised routing over the network. The executions of cryptographic algorithms on different architectural configurations are simulated, and the obtained power traces are subjected to DPA attacks. The results show that the proposed architecture introduces a level of non-determinism in the execution that significantly raises the threshold for DPA attacks to succeed. In addition, the

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940
International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 5, Issue 10, October 2016

performance analysis shows that the improved security does not degrade performance.

Cryptography in its traditional setting examines the security of the cipher as a mathematical function. In addition, it assumes that the secret information can be physically protected in tamper-proof locations and manipulated in closed, reliable computing environments. However, cryptographic systems are implemented on real electronic devices that process, transmit and store data. While operating, these devices interact with and influence the environment and leak a certain amount of information into so-called side-channels. An attacker can potentially compromise the secret cryptographic key stored in these devices by monitoring information that is leaked into side-channels. This type of cryptanalysis is known as side-channel analysis.

Simple power analysis (SPA) is a cryptanalytic technique whereby information about the operation performed in the device, or the operands manipulated in the operation, can be directly interpreted from a single power trace. Often this single trace is replaced with the average of a number of traces in order to reduce the measurement noise. The success of this approach and the techniques used in the attack depends on the implementation of the cryptographic algorithm and the operations used in it.

Differential power analysis (DPA) is a class of side-channel attack that is more powerful than simple power analysis. Actually, DPA is believed to be the most threatening attack that resulted from Kocher's research. This is primarily because the attacker does not need to know as many details about the algorithm implementation in order to perform this attack. Moreover, this attack gains additional strength by using statistical analysis to help recover the secret information from the side-channel.

To carry out a DPA attack, an attacker must have a number of power consumption curves (PCC) collected from a device that has repeatedly executed a cryptographic operation with different inputs and the same key. It is crucial that PCCs contain information about the secret key that can be deduced using statistical methods. The algorithmic condition, the so-called fundamental hypothesis, states that for a DPA attack to be successful the following must be true.

2. REVIEW OF RELATED LITERATURE

Agarwal et al. (2012) described a general strategy to render side-channel attacks more difficult to apply is to balance and randomize major computations which involve the secret key. These attacks largely depend on the possibility to statistically correlate different runs of the same algorithm with the same key and different plaintexts. This means to correlate power consumption curves and the points on the curves that correspond to vulnerable operations (i.e. those that involve the secret key).

Moyart et al. (2010) described a number of countermeasures against the DPA attack and its variations have been proposed in recent years. However, the vast majority of these counter measures do not guarantee security against these attacks, but rather raise the threshold for such attacks to succeed or force the use of more complex and costly techniques.

Goubin et al. (2013) described a general observation concerning software countermeasures is that they are easy and inexpensive to implement (as they do not require the redesign of the existing hardware), but are not applicable to every cipher and are still susceptible to higher-order DPA attacks or signal processing analysis.

Anderson et al. (2011) described that hardware countermeasures, similarly to software countermeasures, focus on destroying the correlation between the power measurements and the values of the secret key. Another target of hardware countermeasures is the alignment of operations in power consumption curves, an important property used by DPA.

Kuhn et al. (2014) described that removing the correlation between features in the DPA profile and the algorithm source code makes retrieving useful information from the power traces significantly harder. Hardware countermeasures can generally provide a higher level of security but can also be costly in terms of performance, power efficiency and memory requirements.

Albert et al. (2012) described that the attack expresses the entire algorithm as multivariate quadratic polynomials, and uses an innovative technique to treat the terms of those polynomials as individual variables.

It relies on first analyzing the internals of a cipher and deriving a system of quadratic simultaneous equations. These systems of equations are very large, for example 8000 equations with 1600 variables for 128-bit AES. The variables represent not just the plaintext, cipher text and key bits, but also various intermediate values within the algorithm.

Richard et al. (2012) described that in the XSL attack a specialized algorithm, termed as extended Sparse Linearization (XSL), is applied to solve these equations and recover the key. In this attack, unlike other forms of cryptanalysis such as differential and linear cryptanalysis, only one or two known plaintexts are required.

3. RESEARCH WORK

For more than 40 years Data Encryption Standard (DES) has been the most widely used commercial encryption algorithm for protecting financial transactions and electronic communications worldwide. Developed by the US Government and IBM in the 1970s, DES was the government-approved symmetric algorithm for protecting sensitive information.

The DES algorithm uses a 56-bit encryption key, which means that there are 72,057,594,037,927,936 possible keys. Considering the computational power level of the 1970s, exhaustive search on the key space of this size was infeasible. However, with the increase in computational power this has become feasible.

A machine jointly built by Cryptography Research, Advanced Wireless Technologies, and Electronic Frontier

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940
International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 5, Issue 10, October 2016

Foundation can perform a fast key search on DES. This project developed purpose-built hardware and software to search 90 billion keys per second, and was able to determine the key after only 56 hours. This attack demonstrated that the exhaustive search on DES is possible and that the 56-bit key length is not sufficient. However, performing this attack is expensive.

The major concern for smart-card manufactures are the attacks which can be performed with relatively inexpensive equipment in a small amount of time, such as side-channel attacks. The DES algorithm uses 64-bit keys to encrypt and decrypt 64-bit blocks of data.

The 56 bits of the key are generated randomly and used directly by the algorithm. The remaining 8 bits are used for error detection and are set to make the parity of each 8-bit byte of the key odd. The operations of encrypting and decrypting in DES are performed using the same key.

The algorithm's overall structure is shown in Figure 1. The algorithm consists of the following: the initial permutation (IP), 16 identical stages of processing called rounds, and the final permutation (FP), which is the inverse of the initial permutation.

After the initial permutation, and before the main rounds, the resulting 64-bit block is divided into two 32-bit halves, left (L) and right (R), which are then processed alternately.

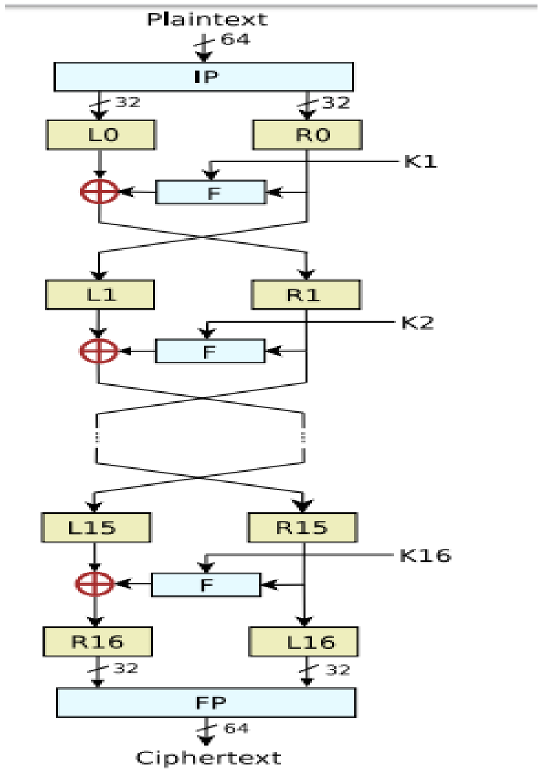

Figure 1: The Feistel structure of DES encryption algorithm.

4. SIGNIFICANCE OF THE STUDY

Cryptographic operations are physical processes in which data is represented by physical quantities in physical structures. These are then stored, sensed and combined by the elementary logic devices (gates). At any point in the evolution of technology, the smallest logic device must have a definite physical extent, require a certain amount of time to perform its function and dissipate switching energy when transiting from one state to another.

A corollary of the second law of thermodynamics states that in order to introduce direction into transition between states, energy must be lost irreversibly. A system that does not dissipate energy cannot make a transition and therefore cannot compute.

It has been shown that this energy can be correlated with the operations performed and the data that is being processed. While operating, electronic devices interact and influence the environment. Besides consuming and emitting power, these devices emit electromagnetic radiation and react to temperature changes.

This information leakage is intrinsic to the physical implementation of the device, and is characterized as the side-channel. If observed and recorded, information leaked into side-channels can be used to recover compromising information (secret keys for example) about the device in question.

This is particularly true for cryptographic devices for which the secrecy of the key is imperative. This type of analysis defines the branch of cryptanalysis known as side channel analysis.

REFERENCES

[1] Security Requirements for Cryptographic Modules. Federal Information Processing Standards Publication 140-2, National Institute of Standards and Technology, January 2014.

[2] D. Agarwal, B. Archambeault, J. Rao, and P. Rohatgi. The EM SideChannel(s): Attacks and Assessment Methodologies. Technical report, I.B.M. T.J. Watson Research Center, Yorktown Heights, NY.

[3] D. Agrawal, B. Archambeault, J. Rao, and P. Rohatgi. The EM Side-Channel(s). In B. S. Kaliski, C¸ . K. Koc¸, and C. Paar, editors, Revised Papers from the 4th International Workshop on Cryptographic Hardware and Embedded Systems (CHES 2012), volume 2523-LNCS, pages 29–45. Springer-Verlag, 2012.

[4] M.-L. Akkar, R. Bevan, P. Dischamp, and D. Moyart. Power Analysis, What Is Now Possible... In T. Okamoto, editor, The Proceedings of the 6th International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT 2010), volume 1976-LNCS, pages 489–502. Springer-Verlag, 2010.

[5] M.-L. Akkar and L. Goubin. A Generic Protection against High-Order Differential Power Analysis. In T. Johansson, editor, Revised Papers from the 10th International Workshop on Fast Software Encryption (FSE 2013), volume 2887- LNCS, pages 192–205. Springer-Verlag, 2013.

[6] R. Aleliunas. Randomised Parallel Communication. In The Proceedings of the ACM-SIGOPS Symposium on Principles of Distributed Computing, pages 60– 72, 2012.

[7] R. Anderson and M. Kuhn. Tamper Resistance - a Cautionary Note. In The Proceedings of the Second USENIX Workshop on Electronic Commerce, pages 1–11. USENIX Association, 2011.

[8] R. Anderson and M. Kuhn. Low Cost Attacks on Tamper Resistant Devices. In B. Christianson, B. Crispo, M. Lomas, and M. Roe, editors, The Proceedings of the 5th International Workshop on Security Protocols (IWSP'14), volume 1361- LNCS, pages 125–136. Springer-Verlag, 2014.