Energy Conscious Adaptive Security

Chryssanthi Taramonli

A thesis submitted for the degree of

Doctor of Philosophy

November 2014

School of Engineering

THE UNIVERSITY OF WARWICK

Table of Contents

List of Figures

List of Tables

Abbreviations

AES	Advanced Encryption Standard
ANOVA	Analysis of Variance
CBC	Cipher Block Chaining
CDF	Cumulative Density Function
CFB	Cipher text Feedback
CLT	Central Limit Theorem
CPU	Central Processing Unit
DES	Data Encryption Standard
DESL	Data Encryption Standard Light
ECB	Electronic Codebook
ECDF	Empirical Cumulative Density Function
I.I.D.	Independent and Identically Distributed
IP	Internet Protocol
IV	Initialization Vector
JCA	Java Cryptography Architecture
JCE	Java Cryptography Extension
MANET	Mobile Ad hoc Network
NIST	National Institute of Standards and Technology
OFB	Output Feedback
P-box	Permutation box
PIN	Personal Identification Number
PDF	Probability Density Function
RFID	Radio Frequency Identification
RAM	Random Access Memory
RC2	Rivest Cipher
R.V	Random Variable
S-box	Substitution box
SPN	Substitution Permutation Network
SPSS	Statistical Package for Social Sciences

SSL	Secure Sockets Layer
TDES	Triple Data Encryption Standard
VIF	Variance Inflation Factor
WSN	Wireless Sensor Network
XOR	Exclusive OR

Terms introduced for this work

Encryption parameters	Data size, Key size, Mode of operation, Padding scheme
Security mode/case	Combination of encryption parameters
Security requirement	Encryption parameters specified by the user
Time threshold	Maximum encryption time desired by the user
Energy threshold	Maximum energy consumption desired by the user
Lifetime	Probability that encryption will finish prior to a threshold
Reliability	Probability that encryption will continue after a threshold
Optimum threshold	Threshold that results in the desired reliability

Acknowledgements

First and foremost, I would like to express my sincere gratitude to my supervisors, Prof. Roger J. Green and Dr. Mark S. Leeson, for their help and guidance throughout this work. I am gratefully indebted to them for their support and valuable advice along the way.

I would also like to thank my parents for their continuous support and understanding throughout my studies.

Last but not least, I would like to thank my fiancé whose conversation with has enabled me to expand the perspectives of this work, but most importantly I would like to thank him for his constant encouragement, patience and for having faith in me.

Abstract

The rapid growth of information and communication systems in recent years has brought with it an increased need for security. Meanwhile, encryption, which constitutes the basis of the majority of security schemes, may imply a significant amount of energy consumption. Encryption algorithms, depending on their complexity, may consume a significant amount of computing resources, such as memory, battery power and processing time. Therefore, low energy encryption is crucial, especially for battery powered and passively powered devices. Thus, it is of great importance to achieve the desired security possible at the lowest cost of energy.

The approach advocated in this thesis is based on the lack of energy implication in security schemes. It investigates the optimum security mode selection in terms of the energy consumption taking into consideration the security requirements and suggests a model for energy-conscious adaptive security in communications. Stochastic and statistical methods are implemented – namely reliability, concentration inequalities, regression analysis and betweenness centrality – to evaluate the performance of the security modes and a novel adaptive system is proposed as a flexible decision making tool for selecting the most efficient security mode at the lowest cost of energy. Several symmetric algorithms are simulated and the variation of four encryption parameters is examined to conclude the selection of the most efficient algorithm in terms of energy consumption. The proposed security approach is twofold, as it has the ability to adjust dynamically the encryption parameters or the energy consumption, either according to the energy limitations or the severity of the requested service.

Chapter 1 - Introduction

The rapid evolution of communication and the subsequent rise of security threats in recent years, have led to the development and application of a plethora of security schemes. To facilitate secret communication, modern security systems rely mainly on encryption, the process of encoding information in a way that only the legitimate receiver can decode.

Communication over networks often requires propagation of sensitive information between parties. Online transactions, shopping, internet banking, Wireless Sensor Networks (WSN) for healthcare monitoring, are only few cases that require propagation of sensitive information, such as contact details, medical records, passwords and Personal Identification Numbers (PINs). Furthermore, with the evolution of Cloud Computing, that provides shared computing resources and data storage in a network, encryption is a vital concern regarding the security of the data. Therefore, the incredible growth of data communications and complexity of modern communication systems, as well as the resulting growth of security threats, has led to the development of complex and time consuming encryption algorithms. The latter, however, depending on the complexity of the algorithm, may consume a significant amount of computing resources, such as memory, processing time and battery power [1]. The key challenge in providing low energy encryption solutions is subject to the offset between minimum energy consumption and maximum encryption strength [2]. Hence, investigating and designing energy efficient encryption systems is necessary in order to minimize the energy consumption.

1.1. Motivation

Knowledge about the optimum selection of the most efficient encryption algorithm under specific security restrictions would help in designing systems that can adjust the security level, according to the desired level of strength while taking into consideration the energy implications. Consequently, the relationship between energy consumption and encryption parameters have to be investigated and modelled. In this work this issue is addressed and the performance of the encryption schemes for all available security options is investigated.

Traditional approaches mainly cope with maintaining a high level of confidentiality and along this line a great deal of effort is put in achieving high secrecy. However, the significant implication of energy consumption is not taken into consideration [2]. Therefore, modern approaches should bring together encryption strength and energy saving. Existing approaches that take into consideration the encryption energy cost are mainly based on experimental comparisons of encryption parameters in terms of effectiveness and provide results on their behaviour with respect to their impact on energy consumption. Although such efforts are very interesting, as demonstrated in [3], there is an inherent need to develop global metrics to be used in specifying the strength of encryption algorithms.

In spite of the fact that most authors use the individual performance of the encryption parameters as factors to compare and rank algorithms, it does not seem reasonable to consider the overall energy performance of the encryption system in complete isolation from security [1] and [2]. This further stresses the need for a *global* quality factor [3] and explains the importance of the development of a decision making framework that evaluates the overall impact of each security mode on energy consumption. The latter statement is based on the fact that system energy consumption depends on the *combination* of parameters not just on their individual impact [4].

The purpose of this thesis is to address all these issues and solve the problem associated with the lack of the energy implication in modern security techniques.

1.2. Research questions

The aim of this research is to answer the following research questions:

1. Can the adaptability of encryption systems be based on the energy consumption?

2. Does the combination of modelling techniques, improvement strategies and verification methods facilitate the development of energy conscious adaptive security?

3. Can energy conscious adaptive security be integrated into a formal decision making process?

1.3. Aims and objectives

The aims of this thesis concern the maximization of encryption performance in terms of energy consumption management, taking into consideration several inter-related factors. It is intended to develop a generic security framework for the decision making regarding the most efficient security mode selection. To this end, the following objectives have been set out:

1. To identify the limitations of existing security approaches

2. To deploy stochastic and statistical analysis using several mathematical methods

3. To build a model that represents the encryption components and can be utilised to estimate the energy consumption

4. To examine the impact of the encryption parameters on energy consumption

5. To investigate how a security framework can utilise this information to maximize encryption performance

6. To develop a decision making tool for energy consumption management in encryption systems

1.4. Scope

The approach advocated in this thesis is based on the lack of energy implication in security schemes. In order to deal with this energy implication with a consistent manner, this thesis adopts several stochastic and statistical methods. In this way, a novel adaptive system is proposed as a flexible decision making tool for selecting the most efficient security mode at the lowest cost of energy. This adaptive scheme permits the interaction of the desired security with the energy constraints, allowing the system to switch to the optimum security mode. The proposed security approach is twofold, as the security mode can be adjusted either according to the severity of the requested service, or according to a desired energy threshold [4].

The proposed framework is generic, as it is not tied to any specific encryption scheme or technology and could be therefore applied to any security system or device. Although it would be particularly applicable to battery powered devices and passively powered devices, where low energy encryption is crucial, it could also benefit traditional computing devices. The suggested scheme is intended to assist in the design, implementation, or management of encryption systems that need to adjust their operation based on computational resources and security constraints. In addition, the proposed energy conscious adaptive security scheme can be used in the evaluation phase of an encryption system's development life cycle, in order to assess the effectiveness and performance of the encryption algorithms and/or parameters. This could also prevent possible retrofitting after the final implementation of the system and therefore avoid any costly modifications. Finally, although in this research five symmetric algorithms are presented, the proposed scheme can be applied to other symmetric or asymmetric algorithms as well. Furthermore, the variation of the encryption parameters may include other or even more factors, depending on the application and the parameters of interest for the specific experiment and/or analysis.

The key contribution of this work is a novel approach that stochastically and statistically studies the overall influence of the configuration parameters on the total energy consumption. The distinguishing feature of the work presented in this thesis is the maximization of encryption system performance by energy consumption management, taking into consideration several inter-related factors.

1.5. Organization of this thesis

The rest of this thesis is organized as follows. In Chapter 2 background information of security, energy and mathematical methods is presented, while previous work in low energy encryption is reviewed. Chapter 3 describes the concept of how a global metric for performance evaluation allows for optimal security mode selection. In Chapter 4, the reliability model is presented, the limiting distribution of *n* encryption is analysed, and the results of the experiments are discussed. Chapter 5 presents an extension of the Reliability model, based on Chernoff bounds. Chebychev, Hoeffding, Bennett and Bernstein bounds are used to investigate further the impact of *n* encryptions on the energy consumption. Examples and results of this stochastic approach are included. In Chapter 6, a statistical solution is proposed, based on regression. The model and results obtained from the statistical analysis are also discussed. Chapter 7 deals with the centrality approach, where the encryption system is treated as a graph and the betweenness centrality that measures the significance of the encryption parameters is examined. Finally, Chapter 8 presents the conclusions of this work and discusses possible future work that could extend further the study of the proposed framework.

Chapter 2 - Background and literature review

The accelerated evolution of communication over networks as well as the consecutive rising sophistication of security threats in recent years, have led to the development of a wide variety of security approaches. With a view to achieving secret communication, modern security frameworks rely heavily on encryption, the process of transforming information into code, using mathematical formulas, in a way that only the authorised person is able to decode. Failure to encrypt propagated data, may allow an attacker who is sniffing the network's traffic, to eavesdrop the private communication between the legitimate parties. The importance of encryption becomes even more crucial for the propagation of sensitive data. Examples of sensitive information include, *inter alia*, contact details, passwords and Personal Identification Numbers (PINs) for various online transactions such as online shopping and internet banking, and medical records for Wireless Sensor Networks (WSN) in healthcare monitoring. Web traffic analysis could also reveal sensitive information about political opinions, religious beliefs, sexual preferences or criminal records. Sensitive information, if compromised may cause serious harm to the owner and therefore, sensitive data encryption is currently a privacy and financial regulation for many organizations [5].

In addition to the above online transactions, security and privacy must also be preserved for data that is stored, propagated and managed in modern cloud computing technologies and services that provide shared computing resources and data storage in a network. Encryption, which is the cornerstone of the security architecture, is particularly important when dealing with large and complex data sets. Managing big data from a security point of view may be a challenging task, due to the complexity in the structure of the data that are stored and also due to the difficulty in their processing when stored in shared repositories. Furthermore, when dealing with big data and cloud computing services, modern data retrieval is also a vital aspect that has to be taken into consideration when designing the security scheme of a system. This is especially true for shared resources that host large amounts of data where a lack of adequate security measures may result in security breaches.

Therefore, to ensure that an adequate level of security is provided for all the above processes and services, the development of complex and time consuming encryption algorithms is inevitable. However, depending on its complexity, an algorithm may consume a significant amount of computing resources, such as memory, processing time and battery power [1].

This chapter first provides an overview of communication security and protocols that have been employed in recent years in order to defend against security threats. Also, the notions of cryptography and cryptographic primitives are discussed. The chapter also introduces the adaptive security concept addressed in this thesis and presents the related work in the area of low energy encryption. Finally, the theory of the methods used in the development of the energy conscious adaptive security scheme namely, reliability theory, regression analysis, betweenness centrality, as well as the most popular concentration inequalities are introduced [6].

2.1. Secure communications

In recent years, communication system numbers have grown exponentially in terms of both devices and networks. Security, which is critical for ensuring protected communication among systems, is therefore an important aspect to be considered. Security concerns in communication systems range from user authentication to secure information storage and networking [6]. Therefore, with the growth of networks and communications, security has attracted considerable attention.

Security is the field of communications that consists of the provision and policies utilized to provide the communication system the protection required to deter any kind of threat [7]. It is widely recognised as a priority in the design and development of today's Information Technology systems, because threats such as malicious code and computer hacking are a concern that has been increasing dramatically over the last decade. There are various types of security attacks that a system may be exposed to, such as the denial of service, unauthorized access and confidentiality breach. Therefore, it is essential for a communication system to deploy several measures to defend against such threats. The most commonly accepted security approach involves three processes. These are Authentication, Authorization and Encryption.

2.1.1. Authentication

This is the first part of the security process, where users have to identify themselves and provide verification of their identity. The most common means of user authentication involves the use of a username and password. This is not the safest means of demonstrating identity, as there are various forms of attacks that a hacker may adopt to crack a password. For this reason, authentication technology is usually integrated using hardware mechanisms, such as smart cards, or even biometric solutions, e.g. fingerprint scans, and is also accompanied by other security processes [8].

2.1.2. Authorization

Authorization is the process which after Authentication examines whether the user is permitted to have access to the requested resource or not. Access to the specific resource may be granted or denied, based on a wide variety of criteria. Other than the use of a password, access control may depend on whether that user is a part of a particular group or not [8].

2.1.3. Cryptography

Cryptography concerns communication in the presence of an adversary [9]. Secure communications are commonly accomplished by utilizing security protocols which in turn invariably employ cryptographic algorithms. The latter may include encryption algorithms, which are used to provide authentication and privacy, as well as hash or message digest algorithms that are used to provide message integrity [6].

2.1.4. Encryption

Encryption is the process that is used to algorithmically transform the data into an unrecognizable format and can be achieved with the employment of encryption algorithms. It involves obscuring information through the use of ciphers and rendering it unreadable to the adversary without special knowledge [10]. The authorized communicating party has to decode the encrypted data using a decryption key in order to translate the encrypted data into recognizable information. Encryption complements the authorization and authentication processes and is very important because it can work independently to protect resources when authentication, authorization or both have failed or even in case that they are not at all considered in a security policy [11]. An encryption algorithm is a mathematical function that incorporates a keystream - a sequence of random bits generated from the algorithm's keyspace, which is the set of all the possible keys. Although the level of secrecy of the encrypted data depends on both the algorithm and the key, the encryption algorithm is known publicly. Therefore, the key plays an important role in determining the security of the encrypted data. The more possible keys that can be generated from the keyspace, the harder it becomes for an attacker to guess which key has been used for encryption. For example, an algorithm with a 128 bit key length offers 2^{128} possible keys. Encryption algorithms can be divided into symmetric (private) key algorithms and asymmetric (public) key algorithms. In the former, the sender and receiver must agree upon a key prior to encryption. This private key which is usually a string of characters is then used by the sender to encrypt a message and by the receiver to decrypt the result, known as the ciphertext. In the latter, each party has two keys; one that is displayed publicly and one that is kept private. The sender encrypts the message using the receiver's public key and only the intended recipient is able to decrypt the message [12].

2.1.5. Security primitives

Symmetric encryption or private key encryption is applied to achieve message confidentiality. A secret key is used to encrypt the data and change the content in a particular way. The key must be shared by both communicating parties. The sender applies the encryption function to the data using the key to produce the cipher text. The latter is sent instead of the original message to the receiver, which then applies the decryption function using the same shared key. Message integrity is implicitly provided, as altering the cipher text would result in an illegible decrypted message

[13]. Typical symmetric encryption algorithms include AES, DES, Triple DES, RC2 and Blowfish, which are introduced later in this chapter.

There are two fundamental types of symmetric algorithms: stream ciphers and block ciphers.

2.1.6. Stream ciphers

Stream ciphers operate on streams of plaintext and ciphertext one bit at a time. Bits are encrypted individually, by combining a bit from the key with a bit form the plaintext, using the exclusive or operation (XOR). In stream cipher encryption, the same bit will encrypt to a different bit every time it is encrypted, provided that a different initialization vector is used. Stream ciphers operate via keystream generators, where, a keystream is generated from a small initial key (seed) and is combined with the plaintext to produce the ciphertext, using simple encryption transformations. A typical encryption transformation is the XOR operation, which is applied to the keystream to transform it into a pseudorandom bit stream which is then XORed with a stream of plaintext bits to produce the stream of ciphertext bits. This encrypted stream can be decrypted using the same random bit stream [14].

2.1.7. Block ciphers

In the case of block ciphers, the message is partitioned into data blocks of fixed length and one block is encrypted at a time. If required, the blocks may be padded as well. The blocks are then joined together, using a fixed encryption transformation, to make the ciphertext. Block ciphers encrypt one block at a time with the same key and therefore, the encryption of a plaintext bit depends on every other plaintext bit within the same block. For older block ciphers the block size is 64 bits, while the block size for relatively new designs is usually 128 bits [14].

As a block cipher encrypts plaintext in fixed-size *n*-bit blocks, for arbitrary length messages that exceed the *n*-bit size that the cipher operates on, the simplest approach is to partition the message into *n*-bit blocks and encrypt each of these separately. This can be achieved by using different modes of operation that offer different properties. The modes of operation describe how blocks interconnect with each other. Four of the most common modes of operation namely ECB, CBC, CFB, and OFB, are discussed below [13].

2.1.8. Modes of operation

The Electronic Codebook Mode (ECB) is the simplest mode of operation. In ECB encryption, the plaintext is divided into blocks and each plaintext block is encrypted separately. The forward cipher function is applied directly and independently to each block of the plaintext. Therefore, any ciphertext block does not depend on any previous plaintext block. The resulting sequence of output is the ciphertext. Regarding the decryption, the inverse function is applied directly and independently to each block of the ciphertext and the resulting sequence of output blocks is the plaintext. The advantages of ECB are the simplicity of the operation, as well as the fact that multiple blocks can be operated simultaneously. The disadvantage of this mode of operation is that identical plaintext blocks are encrypted into identical ciphertext blocks and thus, data patterns can be identified. As a consequence, if a block of plaintext is repeated several times, the result of the encryption will contain several copies of the same ciphertext and therefore the encryption might be insecure [15].

The Cipher Block Chaining (CBC) mode of operation features the combining of the current plaintext blocks with the previous ciphertext blocks. Particularly, before being encrypted to generate the ciphertext block, each plaintext block is XORed (chained) with the previous ciphertext block. For the first block, an Initialization Vector (IV) is required as a starting point. When a plaintext block is encrypted, the resulting ciphertext is stored in a feedback register of the same size as the block size. Before the next plaintext block is encrypted, it is XORed with the feedback register to become the next input to the encrypting routine. This is repeated until the end of the message [14] meaning that the encryption of each block depends on all the previous blocks. In this mode, since the ciphertext is constantly changing, if a block of plaintext is repeated twice, the result of the encryption of the two identical blocks will produce two different ciphertext blocks. In decryption, the inverse cipher function is applied to the corresponding ciphertext block and the resulting block is XORed with the previous ciphertext block. In this mode, each ciphertext block depends on all the proceeding plaintext blocks. [15]. The advantage of this CBC is that the attacker cannot deduce the plaintext by looking at the encrypted blocks separately.

In the Cipher Feedback (CFB) mode of operation, the encryption of messages with fewer bits than the block size is allowed. It features the feedback of successive ciphertext blocks into the input blocks of the forward cipher to generate output blocks that are XORed with consecutive bits of plaintext in order to produce the ciphertext. Once the ciphertext is produced, it becomes the input block of the forward cipher to produce the output block and so on. The method also needs an IV for the first block to produce the first output block [15]. CFB mode links the plaintext segments together so that the ciphertext depends on all the preceding plaintext [14]. The advantages of CFB mode are that its operation is simple and that the input to the block cipher is randomized. The disadvantage of CFB mode is that encryption cannot operate in parallel.

The Output Feedback (OFB) mode of operation is a method of running a block cipher as a synchronous stream cipher [14]. OFB mode operates in a similar fashion to CFB, as it features the iteration of the forward cipher on the IV to generate an output block that is XORed with the plaintext to produce the ciphertext. The size of the plaintext is not necessarily an integer multiple of the block size, as in CFB mode. The difference between CFB and OFB is that in CFB, it is the ciphertext that is fed back to the register, whereas in OFB, it is the output block that is fed back to the register. In encryption, the successive output blocks are produced from applying the forward cipher function to the previous output blocks, and the output blocks are combined with the corresponding plaintext blocks by the means of the XOR method to produce the ciphertext blocks. Similarly, in decryption, the successive output blocks are produced from applying the forward cipher function to the previous output blocks, and the output blocks are XORed with the corresponding ciphertext blocks to recover the plaintext blocks [15]. OFB does not allow for parallelism, but it prevents error propagation, as an error in a ciphertext bit will only affect the corresponding plaintext bit.

2.1.9. Encryption algorithms

The Data Encryption Standard (DES) is one of the most well-known symmetric key block ciphers. It was developed in 1970 and was based on IBM’s 128-bit algorithm, called Lucifer. The US National Institute of Standards and Technology (NIST) accepted it as a standard encryption algorithm and it officially became a federal standard in 1976, with a 56-bit key [16]. DES processes plaintext blocks of 64 bits, producing 64-bit ciphertext blocks. The secret key consists of 64 bits but only 56 bits are effectively used. The remaining 8 bits are used for checking parity [11]. The algorithm involves carrying out combinations, substitutions and permutations between the plaintext secret key, while making sure the operations can be performed in both directions for encryption and decryption accordingly. Since it became a standard, many attacks and methods have been recorded that exploit weaknesses of DES, making it an insecure block cipher. In 1998, a computer system was designed that was able to break the DES encryption key in 3 days [17].

As an enhancement of DES, the Triple Data Encryption Standard (3DES) encryption standard applies the DES cipher algorithm three times to each data block. 3DES performs the DES encryption three times with different keys, making it more difficult for an attacker to crack the encryption code. It has a 64 bit block size and a key length of 56, 112 or 168 bits. In 3DES the encryption method is similar to the one in original DES but three 64-bit keys are used instead of one, for an overall length of 192 bits. Although 3DES is a more powerful version of DES and can be much more secure if used properly, due to its complex computation it is three times slower than DES and also slower than other block cipher methods [14].

The Advanced Encryption Standard (AES), which is one of the most popular symmetric key algorithms, was proposed in 1997 by Daemen and Rijmen and published in 2000 by the NIST [18]. It is a fast and flexible block cipher, as it can be implemented on various platforms. It is based on the Rijndael cipher and supersedes the DES. It is based on the substitution-permutation network design principle. A permutation (P-box) is a mathematical operation for rearranging the data, whilst the substitution (S-box) is the operation for the replacement of a data unit with another. There are several techniques for permutations and substitutions. AES operates on a 4 by 4 array of bytes, known as the state and has a fixed block size of 128 bits and a variable key length of 128, 192, or 256 bits with 10, 12 and 14 rounds accordingly [19].

Blowfish is a symmetric key block cipher designed by Schneier, in 1993, as a fast and free alternative to the existing encryption algorithms. It operates on a 64-bit block size of plaintext with a variable key length from 32 to 448 bits [14]. Blowfish is unpatented, license-free, and is available free for all uses [12]. It is one of the most common public domain encryption algorithms. As such, it has been subject to a significant amount of cryptanalysis. Regarding its security, it is susceptible to attacks on reflectively weak keys and therefore key selection is crucial. However, full encryption has not been broken.

Rivest Cipher (RC2) is a variable key-size block cipher, designed in 1989 by Ron Rivest for RSA Data Security, Inc. The cipher was initially intended as a drop-in replacement for DES [20]. According to the designer company, software implementations of RC2 are three times faster than DES. It is a 64- bit block cipher with a variable key length ranging from 8 to 128 in steps of 8 bits. In addition, the speed of the encryption is independent of the key size [14]. RC2 is based on the Feistel network design, which involves 16 mixing and 2 mashing rounds. A mixing round consists of interleaving an expanded key with the plaintext. A mashing round combines different pieces of the expanded key and the result of the mixing rounds [21].

2.1.10. Padding schemes

Block ciphers work on fixed sized encryption blocks. However, messages come in a variety of lengths, sometimes leading to a shorter final block, as the message cannot be divided into the required fixed size blocks [14]. Padding is the way to deal with this problem. When the plaintext to be encrypted is not an exact multiple of the block size, a padding string is added to the plaintext. For the decryption, the padding has to be removed and so the padding scheme has to be known to both communicating parties. Several padding schemes exist for the padding of the final block before encryption but the commonest are ISO10126 and PKCS5.

In ISO10126, the padding is done at the end of the last block with random bytes. The number of added bytes that are required in order to fill the block size is specified by the last byte which is assigned the value of this number so that the receiver knows how many bytes have been padded [14].

In the PKCS5 padding scheme, the number of bytes to be padded is equal to 8 - (number of bytes of plaintext) mod 8. This results in 1 to 8 bytes and depends on the plaintext length. The number of bytes remaining to fill the required block size is added at the end of the last block before encryption, all are assigned the value of the number of the remaining bytes so that the receiver knows the number of padded bytes [11].

2.2. Energy

Encryption algorithms are computationally intensive, consuming a significant amount of energy and computational resources that are crucial especially for battery powered devices. Algorithms may result in a different level of energy efficiency under different circumstances, as some algorithms may provide the same level of security while consuming less energy [22]. This can be shown if one considers the following. Encryption strength is related to the difficulty of discovering the key, which in turn depends on both the cipher used and the key size [23]. In general, longer keys provide stronger security. Different ciphers may require different key lengths to achieve the same security strength. Although a large key length could provide stronger security, it could increase computation and thus energy consumption. Therefore, variations of the algorithmic parameters, i.e. key size, mode of operation, data size and padding scheme and so forth, may result in different levels of energy consumption.

2.2.1. Energy efficiency

Energy efficiency e is defined as the energy dissipation that is essentially needed to perform a certain function, divided by the actual total energy dissipation [24].

$$e = \frac{\text{essential energy dissipation for a cerain function}}{\text{total energy dissipation}} \tag{2.1}$$

The energy efficiency of a certain function is independent of the actual implementation and thus is independent of the issue whether an implementation is low power. Low power is generally closely related to the hardware, whereas energy-efficiency relates to the algorithm using the hardware [24].

Research in the area of energy efficiency was initially focussed on the physical layer, as particularly for wireless devices, power consumption depends on the system hardware [25]. The primary problem regarding energy in mobile devices is that battery capacity is limited. Thus, the main objective of battery technology research is to increase power capacity. However, this area has not experienced significant advances in order to conform to the increasing energy demands [25]. Therefore, the solution lies in the design of energy efficient schemes.

2.2.2. Energy consumption computation

To compute the energy consumption, the technique described by Naik and Wei [26] is employed. Energy consumption can be represented by the product of the total number of clock cycles taken by the encryption and the average current drawn by each CPU clock cycle to deliver the basic encryption cost in units of ampere-cycles. To calculate the total energy cost, this basic ampere-cycle encryption cost is divided by the processor clock frequency in cycles per second to obtain the energy cost of encryption in ampere-seconds. Multiplying this by the processor's operating voltage produces the energy cost (ε) in Joules.

$$\varepsilon = \frac{\text{clock cycles}\times*\text{average current by CPU clock cycle}\times\text{processor}'\text{s operating voltage}}{\text{clock frequency}} \qquad (2.2)$$

2.3. Adaptive security

The rapid development and extensive application of computer networks have brought new challenges in the area of information security [27]. In addition, the number and complexity of security attacks in recent years has increased considerably. This raises particular concerns on the ability of software development methods to deal with this challenge. The traditional static security model and a single security policy cannot solve this problem [28]. Most traditional security techniques were developed without taking into account the significance of dynamic elements implication. Initial efforts to develop security systems employed static methods with fixed security measures. Although this has led to the reduction of system vulnerabilities, with the advance of the attacks in terms of complexity, the security measures had to be strengthened. However, this advance in the security system brought with it an increased rate of processing overhead and resource consumption of the system infrastructure, leading to higher development and maintenance costs [27]. The solution to this obstacle is therefore adaptive security. Adaptive systems dynamically change their behaviour in order to respond to specific changes. There are several advantages in a system that is able to adapt its security mechanisms compared to a static security system. For example a system could respond to intrusions by strengthening its security policies. In addition, different users need different level of security strength, for example the department of defence network would require higher security than a personal webpage [29]. Finally, different users have different access rights and therefore the adaptive system could apply the appropriate restrictions by dynamically adapting users' access.

What is more, mobile devices have experienced a period of rapid evolution in recent years, bringing unprecedented changes in mobile applications. At the same time, security risks have risen with the sophistication of mobile devices leading to the development of several security schemes for mobile devices [30]. However, encryption, which is the cornerstone of security, comes at a significant energy cost [31]. Coupled with the aforementioned lack of progress in battery technology this has led to a considerable decrease in battery life. According to [32] there is a widening *battery gap* between trends in processor power consumption and improvements in battery capacity. Thus, referring to low energy encryption for mobile devices, there is an intrinsic need to provide a sufficient level of security at the lowest energy cost [2].

The key challenge when designing adaptive security schemes is to find a common point between the static behaviour of traditional security systems and the dynamic security provision of adaptive systems.

The following section provides a background in adaptive security through discussion of related literature.

2.4. Relevant work

This section provides an overview of previous work in the area of adaptive security with regards to low energy consumption. This work has been influenced by several research areas. The main and most relevant one is the comparison based approaches, where researchers compare and rank encryption algorithms and encryption parameters based on their impact on the energy consumption. The comparison is made either between algorithms, or based on the variation of the encryption parameters of the same algorithm. Furthermore, the concept of reusing existing ideas and principles that is described in Green and lightweight cryptography, as well as in minimalism in cryptography, has been adopted and is considered the main intention of this work. First three resource conservation oriented areas of cryptography are introduced and then general approaches in the area of low energy encryption, including the comparison based approaches, are discussed.

2.4.1. Green Cryptography

Green cryptography suggests using ideas that have proven their merits. Encryption algorithm and protocol design should recycle existing components and primitives, while developing encryption should be based on the selection of existing algorithms, according to the needs of the individual service. Green cryptography is aimed at sustainable security within scalable implementations. It is about maximizing confidence in cryptographic primitives while minimizing complexity in their implementation [33].

In Troutman and Vincent’s work [33], a green approach to the design process is suggested. They illustrate the concept of green cryptography using the pedigree of AES and how essential elements of AES have been recycled in the design of its successors. The aim of their work is to optimize the efforts of designers that have already been spent on designing primitives and algorithms. To further support their concept, they used Rijndael’s round transformation and compared to Twofish’s and Serpent’s round transformation. They concluded that for different number of rounds, the resulting numbers of full diffusion steps is varied, and therefore the ranking of the compared ciphers is changed. This way, they showed that it is not always easy to compare algorithms based on only one metric; instead, sometimes combinations of metrics should be taken into consideration.

Several authors proposed security methods based on the concept of green cryptography. A method for the construction of a compression function that could be extended to a hash function, based on a fixed key block cipher was proposed in [34]. The authors analysed several schemes in terms of their security strength, by performing attacks and they provided bounds on the security of each scheme.

In [35] the authors introduced the idea of using block ciphers to construct hash functions and they proposed a method for constructing hash functions based on block ciphers, where the hash code size is equal to the block size of the cipher and approximately equal to the key size. Their model can be used to identify and compare secure schemes. The aim of their work was to minimize the design and implementation effort. However, the first attempt to construct hash functions from block ciphers was intended for the use with the DES [35].

Although these approaches mainly deal with specific ciphers and they present ways to minimize the design and implementation effort, the concept of reusing existing schemes is similar to the concept of this thesis. Specifically, although the proposed scheme is not tight to any specific algorithms, the analysis in terms of their efficiency can be used to identify the most energy efficient algorithm, for the requested security service. In this way, it is not necessary to implement a new low energy encryption algorithm; instead, based on its performance, the most efficient scheme can be identified.

2.4.2. Lightweight Cryptography

In this area of cryptography the aim is to provide cryptographic algorithms and primitives intended for use in devices with limited resources [36]. The main concern of lightweight cryptography is extremely low resource requirements. Therefore, the main idea is to find a compromise between low resource requirements, performance and strength of cryptographic algorithms and primitives [36].

DESL, a new lightweight DES variant was proposed in [37], which is based on the classical DES design, but uses a single S-box repeated eight times, instead of eight S-boxes of the original DES. The reduction of the memory requirements for the S-box storage made the light version of the original DES suitable for devices with constrained resources. Although the proposed algorithm was proved to be resistant against certain types of attacks, this is not true for all types of attacks. DESL does not provide high level of security, compared to its predecessor, the classical DES cipher.

In [38] the authors propose PRESENT, which is a Substitution-Permutation Network based block cipher, suitable for small cyber-physical systems and is notable for its compact size – 2.5 times smaller than AES. The cipher has been designed considering security and power constraints as well. Although the main goal when designing PRESENT was simplicity and hardware optimization, it provides adequate security for applications with low security requirements. However, PRESENT is targeted to specific applications in constrained environments, such as Radio Frequency Identification (RFID) tags and sensor networks and therefore can only be applied to applications that require moderate security levels.

The KLEIN family of lightweight block ciphers proposed by Gong et al. [39] is an SPN cipher and was also designed for resource constrained devices. KLEIN offers can have various key sizes and therefore can provide a moderate security level for several applications, specifically in environments such as RFID tags and sensor networks. Although it is resistant against specific cryptanalytic attacks, it has been proved that it has a conservative security margin against various cryptanalysis.

The authors in [40] proposed KATAN and KTANTAN a new family of efficient hardware oriented block ciphers that offer a solution for low-end devices where encryption is necessary. In [41] the authors suggest the use of the KATAN lightweight block cipher as a base for various cryptographic functions, including block ciphers, stream ciphers and hash functions, therefore incorporating ideas form Green cryptography. In [42] the same authors propose the use of a lightweight block cipher as a cryptographic kernel to mount various types of cryptographic algorithms that do not require significant resources. The authors also suggest a way to extend the set of cryptographic algorithms of the IPSec protocol and include lightweight algorithms.

Efforts in the area of lightweight cryptography are mainly aiming to environments with limited resources. This however is usually at the cost of security strength, as these schemes can only provide moderate security. Although the design of lightweight ciphers is very important for applications with low or moderate security requirements, the mechanisms that are adopted in order to achieve higher efficiency in the implementation may result in great expense in the security level that can be achieved. Therefore, it is very important to investigate how this issue can be balanced. In this work, the aim is to provide the most efficient security mode, considering possible resource limitations.

2.4.3. Minimalism in Cryptography

Over the past decades, the analysis of minimal constructions has played an important role in the area of cryptography. A great deal of effort was put in achieving the minimal cryptographic assumptions that are sufficient for the construction of cryptographic primitives and algorithms. Research has been carried out for example on the analysis of the smallest number of rounds that is needed to make Feistel structures with truly random functions secure, as well as on the simplest way to transform one primitive into another by using the appropriate mode of operation [43].

In 1984 Ron Rivest proposed DES-X, an extension of DES, intending to increase the DES key size without altering the cipher's internal structure, in order to increase the strength of DES against exhaustive key search attack [44]. The idea was to augment the original 56-bit key DES by XORing an extra 64-bit key to the input before applying DES and then XORing another 64-bit key to the output of DES-after the encryption. Although this version of DES was proved to improve the cipher's resistance against differential and linear cryptanalysis compared to the original DES, for brute force attacks there is no significant improvement regarding the security strength compared to DES.

Influenced by the DES-X design, in 1991 the Even-Mansour block cipher was proposed [45]. This scheme used similar keys but eliminated the keyed block cipher between the two XOR operations, replacing it with a fixed random permutation. Therefore, in order to encrypt a plaintext, the latter has to be XORed with one key before applying the random permutation, and the outcome is then XORed with a second key. Furthermore, as only one permutation is required, there is no need to generate and store many permutations. The designers showed that when the permutation is random, the cipher is secure. However, it was later proved that the scheme can be compromised.

As described in this subsection, the idea of adjusting specific parameters of an algorithm or a primitive can improve the security level. This concept has been adopted in this thesis, as the security modes can be alternated by adjusting the encryption parameters of the algorithms. In this way, although the required security level is achieved, the selected security mode does not provide stronger security than is actually needed and therefore no unnecessary energy is consumed.

2.4.4. Other efforts

In the absence of generally accepted metrics that could be used to analyse and quantify cryptographic strength, Jorstad and Smith [3] tried to explore the possibility of developing an approach to cryptographic metrics that could be used to describe the attributes of encryption algorithms and develop a framework for specifying the strength of cryptographic technologies. Although an objective metric was not identified, a subjective scale was suggested for rating the overall strength of an algorithm. The concept of generalising the way that the available security modes can be compared is very interesting. Although this is a rather challenging task, the concept has been adopted in this thesis. Specifically, a metric has been proposed, that can be used to compare algorithms based on their probability of finishing the encryption process prior to a threshold – based either on time or energy.

Existing efforts to investigate the energy consumption characteristics of encryption algorithms mainly deal with comparison based approaches. These studies are based on experiments performed either for different encryption algorithms where the impact of the encryption parameters on the energy consumption is observed, or for one algorithm where the impact of its encryption parameters' variation on energy consumption is observed. The latter is usually based on a specific encryption parameter i.e key size variation, and the impact of this variation is then analysed. Although this information can be useful for further analysis of the algorithms and their behaviour with respect to the energy consumption, the desired security level is not taken into consideration. In this work, the suggested scheme examines and compares the energy consumption of different algorithms based on the encryption parameters' variation, while the security restrictions for the requested service are also taken into account. Furthermore, in the proposed scheme, several algorithms can be compared, as the scheme is not tight to any specific ciphers. The comparison is based on the encryption parameters' variation and therefore the impact of each one of them can also be investigated.

Lamprecht et al. [46] conduct a comparative performance evaluation based on the implementation of different encryption algorithms. The authors examined and compared the average encryption time of a fixed size file using AES, DES, Blowfish and DESede (DES variant). Following, they measured the encryption time for the same algorithms using two modes of operation, CBC and ECB. Finally, they analysed the performance of DES for different data sizes. Their study provides interesting knowledge concerning the cryptographic methods, but does not generalise a methodology for performance evaluation. Furthermore, it does not provide any information about the relationship between security and performance and they do not take energy consumption into consideration.

In [1], [47]-[48], the authors describe the effects of individual adjustments in encryption parameters on security with respect to energy consumption. In their work, a performance comparison between common encryption algorithms is presented. In [1], the authors compared six algorithms based on their performance for different types of files – text, audio and video. They used the throughput as the metric for the performance comparison of the algorithm. Guo et al [47] compared AES, DES, 3DES and Blowfish in terms of the energy they consume. In addition, they performed comparisons of the algorithms for different key sizes and modes of operations. However, the encryption parameters were varied one at a time. Therefore, only one parameter can be examined following their approach. In [48] the authors examined AES, DES, 3DES and Blowfish and compared them for a specific data size. In their approach, they varied the key size and they performed the experiment for two different modes of operation – ECB and CBC. They used the throughput as a performance indicator.

Although such efforts are very interesting, there is a demonstrated inherent need [3] to develop global metrics for use in specifying the strength of encryption algorithms. Potlapally et al. [6] studied the energy consumption requirements of security protocols, using a parametric approach and focusing on battery-powered devices and the application of the SSL. In their work, the authors compare several algorithms, by measuring their encryption time and the corresponding energy consumption. In their approach, they also consider the levels of cryptanalytic difficulty. However, the encryption parameters remain fixed in their comparative approach. The same authors also presented a framework for analysing the energy consumption of encryption algorithms and compared the performance of common ciphers.

In [49]-[50] the authors highlight general problems and methods of their solutions concerning the adaptive security concept in complex information systems. Their theoretical approach concerning adaptive security implies the use of control theory and dynamical systems theory. According to the authors, information gathering required for the adaptation of a complex secure system can be achieved by registering external influences and/or internal states. The optimal solution would be achieved by the combination of these, but at the cost of the resources. The proposed method is based on the optimal control of the system whose internal states depend on the external influences.

In [51] a security framework for distributed system control is presented with a focus on device level system control. The security problems of collaborative distributed systems are addressed and a security framework is proposed based on three logical domains, namely the client (stores user credentials), task repository (stores task components, including functions and policies) and low level control device (security control gateways incorporate requests and control actions and guarantee the secure task and control action execution).

An adaptive security scheme for denial of service threat has been proposed in [52] based on a fuzzy feedback controller that behaves similarly to human immune system when a virus is detected. The system monitors specific parameters and how fast they change to identify threats. In addition, it allows the user to select the security level according to need.

In [53] the authors describe a resource aware adaptive security framework for mobile ad hoc networks (MANETs), at the protocol level. Their scheme selects the optimal set of protocols, one from each layer, with the maximum security and network performance services. They introduce two indices; a security index and a performance index that are computed and then used for the optimal protocol set selection. Security is evaluated using high, medium and low security levels, while analysis of variance is used for the performance evaluation. The concept of categorizing algorithms based on their security strength has been adopted by several authors. Son et al. [54] propose a security manager that dynamically adapts to real-time performance conditions. Their method provides four degrees of protection that depend on a four-level security classification. Zou et al. [55] present the architecture of an intelligent firewall. In their method, packet characteristics (IP address and port number) are “fuzzified” to produce fuzzy inputs. Using an adaptive fuzzy security algorithm, the fuzzy inputs, as well as the security policy rules, the appropriate security level is figured out and adjusted accordingly. It could be characterised as an attempt to combine packet filtering and application level firewalls.

Although some of the approaches described in this section are not directly linked to this work, several concepts and methods based on these approaches have been incorporated for the implementation of the security scheme presented in this thesis. Since there is no established scheme that investigates the energy consumption of encryption algorithms based on the probability of finishing the encryption prior to a threshold, whilst the security restrictions are also taken into consideration, here a generic model is developed that can be used to explore how energy consumption and encryption requirements can compromise.

2.5. Methods used

The rest of this chapter focuses on the theory of the techniques that were used for the development of this energy conscious adaptive security scheme.

2.5.1. Reliability function

Reliability theory considers the performance of a system over time. Let T be the lifetime of a system or component with Probability Density Function (PDF) *f(t)* and Cumulative Density Function (CDF) *F(t)* as shown in (2.3).

$$F(t) = P(T \leq t) = \int_0^t f(s)ds \tag{2.3}$$

In reliability engineering the concern is with the probability that the system will survive for a stated interval of time i.e. there is no failure in the interval (0, t). This is known as the survival function and is given by *R(t)*.

$$R(t) = 1 - F(t) = P(T > t) \tag{2.4}$$

A reliability function represents the probability that for a given time *t*, the system will survive [56]. A system S that consists of four subsystems connected in series is considered. The reliability for system *S* will be [57]:

$$\mathrm{R(t)} = \mathrm{P(T} > t) = \prod_{\mathrm{i=1}}^{n=4} \mathrm{P}(T_i > t) \qquad (2.5)$$

where *T* represents the total lifetime of the system, while T_i stands for the lifetime of subsystem S_i [4].

The reliability function is the complement of the CDF. If modelling the time to fail, the CDF represents the probability of failure and the reliability function represents the probability of survival. Thus, the CDF increases from zero to one as the value of *t* increases, and the reliability function decreases from one to zero as the value of *t* increases [57]. The CDF is thus:

$$\mathrm{F(t)} = 1 - \mathrm{R(t)} = \prod_{\mathrm{i=1}}^{n=4} \mathrm{P}(T_i \leq \mathrm{t}) \qquad (2.6)$$

The Empirical Cumulative Density Function (ECDF) F(t) is a step function with jumps *i/n* at observation values, where i is the number of tied observations at that value and *n* is the number of observations. For observations $x = (x_1, x_2, \dots, x_n)$, F is the fraction of observations less than or equal to *t*:

$$ECDF(t) = F(t) = \frac{1}{n} \sum_{i=1}^{n} I(x_i \leq t) \qquad (2.7)$$

where I is the indicator function [56].

2.5.2. Concentration inequalities

Concentration inequalities provide probability bounds on the deviations of functions of random variables from their expectation. A random variable with good concentration is a variable which is close to its mean with high probability. A concentration inequality, also known as tail bound, is a theorem providing that a random variable has good concentration [58].

For any nonnegative random variable X,

$$E[X] = \int_0^\infty P(X \geq t)dt \tag{2.8}$$

2.5.2.1. Markov's inequality

For any nonnegative random variable X and for any $t > 0$,

$$P(X \geq t) \leq \frac{E[X]}{t} \tag{2.9}$$

Although this is the simplest concentration inequality, the drawback is that it gives weak bounds [58].

2.5.2.2. Chebyshev's inequality

From (2.9), if f is a monotonically increasing nonnegative-valued function, then for any random variable X and real number *t* [58],

$$P(X \geq t) = P\big(f(X) \geq f(t)\big) \leq \frac{E[f(X)]}{f(t)} \tag{2.10}$$

For $f(X) = X^2$,

$$P(|X - E(X)| \geq t) = P\left(\big(X - E(X)\big)^2 \geq t^2\right) \leq \frac{E\left(\big(X-E(X)\big)^2\right)}{t^2} \tag{2.11}$$

In addition,

$$Var(X) = E\left[\left(X - E(X)\right)^2\right] \tag{2.12}$$

Therefore, from (2.11), (2.12)

$$(|X - E(X)| \geq t) \leq \frac{Var(X)}{t^2} \tag{2.13}$$

2.5.2.3. Chernoff bounds

The tail estimates given by Markov and Chebyshev inequalities, work for random variables in general. When the random variable (r.v.) X can be expressed as a sum of n independent random variables, one can obtain tighter bounds on the tail estimates. Chernoff bounds are used to bound the tail of the distribution for a sum of independent r.v. [59].

Let X be a r.v defined as $X = X_1 + X_2 + \cdots + X_n = \sum_{i=1}^{n} X_i$.

Also let X_i be independent and identically distributed (i.i.d.) such that $X_i \in \{0,1\}$, $\forall\, i \leq n$.

Let $\mu = E[X] = E[\sum_{i=1}^{n} X_i]$ and $P[X_i = 1] = p_i,\ 1 \leq i \leq n$

Then for any $\delta > 0$,

$$P[X \geq \mu(1+\delta)] \leq \left(\frac{e^{\delta}}{(1+\delta)^{1+\delta}}\right)^{\mu} \tag{2.14}$$

2.5.2.4. Hoeffding’s inequality

Chernoff studied the problem of finding a tight bound for binary random variables. Later Hoeffding derived a more general result for arbitrary bounded random variables.

Hoeffding’s Lemma: Let X be a random variable with $E[X] = 0,\ a \leq X \leq b$, then for $s > 0$,

$$E[e^{sX}] \leq e^{s^2(b-a)^2/8} \tag{2.15}$$

For bounded random variables $X_i \in [a_i, b_i]$ where $X_i, \ldots X_n$ are independent [60], then for $S_n = X_1 + X_2 + \cdots + X_n$

$$P(S_n - ES_n \geq t) \leq exp\left(\frac{-2t^2}{\sum_{i=1}^{n}(b_i - a_i)^2}\right) \tag{2.16}$$

and

$$P(ES_n - S_n \geq t) \leq exp\left(\frac{-2t^2}{\sum_{i=1}^{n}(b_i - a_i)^2}\right) \tag{2.17}$$

This inequality is similar to the concept of Markov’s inequality but it is a sharper one.

Probabilities can be estimated from a set of examples using the sample average. The latter approaches the expected average as the number of samples approaches infinity according to the strong law of large numbers. Hoeffding’s inequality provides an estimate of the error of an unconditional probability given n samples.

Hoeffding’s Theorem: Let $X_1, X_2, \ldots X_n$ be i.i.d. observations such that $E[X_i] = \mu$ and $a \leq X_i \leq b$. Then for any $\epsilon > 0$, [58]

$$P(|\overline{X_n} - \mu| \geq \epsilon) \leq 2e^{\frac{-2n\epsilon^2}{(b-a)^2}} \tag{2.18}$$

where n is the sample size. This can be used to determine how many samples are required to guarantee a probably approximately correct estimate of the probability.

2.5.2.5. Bennett's inequality

Hoeffding's inequality does not use any knowledge about the distribution or variance of the variables. Bennett's inequality is a stronger concentration inequality [61], since it uses the variance of the distribution to provide a tighter bound.

Bennett's Theorem: Let $X_1, X_2, \dots X_n$ be independent observations with $E[X_i] = 0$ and $|X_i| < c$ with probability 1.

Let $\sigma_i{}^2 = Var(X_i)$ and $\sigma^2 = \frac{1}{n}\sum_{i=1}^{n} \sigma_i{}^2$

Then,

$$P\left(\frac{1}{n}\sum_{i=1}^{n} X_i \geq \varepsilon\right) \leq exp\left\{-\frac{n\sigma^2}{c^2} H\left(\frac{ct}{n\sigma^2}\right)\right\} \tag{2.19}$$

Where $H(u) = (1+u)ln(1+u) - u$ for $u \geq 0$

2.5.2.6. Bernstein's inequality

Bernstein's inequality is also a stronger concentration inequality [62] compared to Hoeffding's, since it uses the variance of the distribution to provide a tighter bound.

Bernstein's Theorem: Let $X_1, X_2, \dots X_n$ be independent observations with $E[X_i] = 0$ and $|X_i| < c$ with probability 1.

Let $\sigma_i{}^2 = Var(X_i)$ and $\sigma^2 = \frac{1}{n}\sum_{i=1}^{n} \sigma_i{}^2$

Then,

$$P\left(\frac{1}{n}\sum_{i=1}^{n} X_i \geq \varepsilon\right) \leq exp\left(-\frac{n\varepsilon^2}{2\sigma^2+2c\varepsilon/3}\right) \qquad (2.20)$$

2.5.3. Statistical analysis

In this section statistical analysis techniques that identify relationships among variables as well as their impact on the response variable are presented.

2.5.3.1. Correlation

In order to state with certainty if and how predictor variables affect the response variable, the null hypothesis testing *H₀* is used. Its practice is related to the decision making about the statistical significance. In null hypothesis testing, the idea is to state a null hypothesis by assuming that there is no effect on the output variable and then assess whether the evidence obtained from the test does or does not support this hypothesis and rejects or accepts the test accordingly [63]. The null hypothesis is tested by gathering data and then measuring how probable is the occurrence of data, under the assumption that the null hypothesis is true. If data do not contradict the null hypothesis, then H_0 is true and the predictor variables do not affect the response variable. In this case the null hypothesis test is accepted. If data is very improbable – usually defined as observed less than 5% of the time – it is expected that some predictor variables with influence on the response variable will be found and the null hypothesis is rejected.

A *P-value* is a measure of how much evidence there is against the null hypothesis. The smaller the *P-value*, the more evidence one has against H_0. It is also a measure of how likely it is to get a certain sample result or a more extreme result, assuming H_0 is true. The *P-value* is used to obtain the most statistically significant variables influencing the output variable. If the *P-value* between one predictor variable and the response variable is less than 0.05, then the null hypothesis is rejected and the predictor and output variables are highly correlated [64].

2.5.3.2. Regression analysis

Regression analysis is a statistical method for investigating functional relationships among variables. The relationship is expressed in the form of an equation or a model connecting a response variable with one or more predictor variables. Let y denote a response variable and $x_1, x_2, \dots, x_n$ denote predictor variables, a multiple regression equation between y and $x_1, x_2, \dots, x_n$ can be written as

$$y = a_0 + a_1x_1 + a_2x_2 + \cdots + a_nx_n$$

The correlation coefficient R describes the degree to which two or more predictors – independent or X variables – are related to the dependent variable Y. The *R-square* value is an indicator of how well the model fits the data [65].

2.5.4. Betweenness Centrality

In graph theory and network analysis, centrality of a vertex measures its relative importance within a graph [66].

Betweenness Centrality has been established as an important quantity to characterize how influential a node is in communications between each pair of nodes. It is a measure that computes the relative importance of a vertex in a graph and it is widely used in network analysis [67]. The betweenness centrality of a vertex in a graph is a measure for the participation of the vertex in the shortest paths in the graph. It is in some sense a measure of the influence that a node has over the flow of information through the network. Conceptually, high betweenness nodes lie on a large number of non-redundant shortest paths between other nodes [68].

Let $G = (V, E)$ with V vertices and E edges be a graph and let $s,\ t$ be a fixed pair of graph nodes. Let σ_{st} be the number of shortest paths between s and t, and let $\sigma_{st}(v)$ be the number of those shortest paths that pass through v. According to [69] the betweenness centrality of the vertex v is then expressed as:

$$C(v) = \sum_{s \neq v \neq t \in V} \frac{\sigma_{st}(v)}{\sigma_{st}} \qquad (2.21)$$

2.6. Closing remarks

In this chapter, an overview of cryptography and adaptive security as well as related work in the area has been presented. The concept of the energy conscious adaptive security has been introduced and the methods used in the development of the scheme have been briefly discussed.

Chapter 3 is intended to demonstrate the conceptual energy conscious adaptive security scheme. Throughout the following chapters of this thesis, results and experimental procedures performed on the methods introduced in this chapter are presented.

Chapter 3 - Concept and implementation

This chapter introduces the concept of how a global metric for performance evaluation allows for optimal security mode selection, with respect to energy consumption. The first section is intended to demonstrate the conceptual energy conscious adaptive security scheme that involves the minimum energy needed to achieve a desired security. In the second section, an implementation overview is presented, related to the algorithm selection, as well as the simulation and data analysis software used for the implementation, including methods, packages and classes. Finally, the contribution of this work in the area of low energy encryption is addressed.

This chapter is based on the "*Energy Conscious Adaptive security Scheme for Optical Wireless*", published in the proceedings of the 14th IEEE International Conference on Transparent Optical Networks (ICTON), 2012.

3.1. Concept

Mobile devices have experienced a period of rapid evolution in recent years, bringing unprecedented changes in mobile applications. At the same time, security risks have risen with the sophistication of mobile devices leading to the development of several security schemes for mobile devices [30]. However, encryption, which is the cornerstone of security, comes at a significant energy cost [31] and also battery technology has not been able to conform to the increasing energy demands, leading to a considerable decrease in battery life. There is thus an intrinsic need to provide a sufficient level of security at the lowest energy cost [2]. In this work this issue is addressed and the performance of the encryption schemes for all available security options is investigated. One possible way to achieve this is by adjusting all encryption parameters, i.e. key size, data size, mode of operation, padding and so forth.

Traditional approaches generally deal with ensuring the security and accuracy of the propagated data [2]. Although modern approaches take into account the encryption energy cost, existing efforts to examine the energy cost characteristics of encryption mainly comprise experimentally based comparative approaches which assess the behavioural and energy impacts of the encryption parameters [1, 47, 48].

An important further aspect is that energy consumption does not depend only on isolated factors – i.e. key size, padding scheme, mode of operation – but rather there is a correlation between factors and their global effect on energy. To achieve low energy encryption, the offset between minimum energy consumption and maximum encryption strength has to be investigated, meaning that it is essential to explore the relationship between energy consumption and functional encryption parameters. This will facilitate an adaptive security scheme with efficient adjustment of encryption parameters to deliver energy efficient encryption algorithms and protocols.

In order to minimize the energy consumption in an encryption system, many inter-related factors must be considered. The latter are often internally related in a complex system, resulting in high complexity for encryption optimization. An encryption system can be seen as a parametric system with several configuration parameters. Clearly, assigning proper values for these parameters can increase the performance and reduce the overall energy consumption.

In symmetric algorithms, the security level can be altered by adjusting functional parameters [6], such as key size, mode of operation and number of rounds. Although examining the performance of an algorithm, based on the individual performance of each encryption parameter, provides interesting results with regards to their impact on energy consumption, it does not seem reasonable to consider encryption performance in complete isolation from security. One evident example is the key size parameter: a large key length makes the algorithm slower but should provide greater security. Although an obvious way to investigate each parameter's impact on the overall security and energy consumption would be to split the overall security into individual security units, this would only make sense when comparing parameters within the same algorithm. When examining several algorithms, encryption parameters cannot be used as global performance evaluation indicators. Although some authors use those encryption parameters as factors to compare and rank algorithms, this is not generally accepted, as the criteria used are not universal – i.e. there is no reasonable way to judge if a 256-bit algorithm is better or worse than a 128-bit algorithm.

The importance of the dependencies exploration is based on the fact that the system's energy consumption will depend on the combination of the parameters, and not just on individual parameters. In the case of the AES algorithm, for example, the selection of a 128-bit key will operate in 10 rounds [70], which makes the dependence between the key size and the number of rounds evident. Since each parameter combination will result in an energy cost, the aim is to identify the security mode that consumes the minimum energy needed to achieve the desired overall security level. Therefore, it is of great importance to take into consideration the dependencies between those parameters, so that they can be viewed as a function rather than as isolated impact factors. This would allow for a global performance metric that could be further used for the investigation of the balancing between encryption strength and energy consumption.

Adaptive security is based on the observation that the security requirements of a system or service heavily depend on the severity of the operation requested, and should therefore be dynamically adjusted to operate in the most effective way. This scheme is concerned with adapting the choice of encryption algorithms and primitives with respect to energy consumption. In a security system, where several security levels are provided, each associated with its individual energy consumption characteristics, the security scheme offers the option to adapt the level of security depending on the security requirements or on any possible energy consumption restrictions. In the first case, less critical information would be encrypted with lower security, thus resulting in lower energy consumption, whilst more critical information would be encrypted with higher security, consuming more energy. The second case is best exemplified in the case of battery powered devices, where security could be adapted with regards to the state of the battery in order to extend its life.

Therefore, the subsequent security approach is twofold. Firstly, encryption strength is adjusted according to the severity of the requested service. This helps save energy, while preserving the encryption strength. Secondly, for battery-powered devices, data can be encrypted according to a specified threshold, or even based on the battery level itself. The individual method of each security approach, namely reliability theory, Chernoff bounds, multiple regression and betweenness centrality, serves as a quality factor that describes all encryption parameters and their impact on energy consumption, and therefore as a global indicator of the energy consumption.

The proposed adaptive security scheme for low energy encryption is based on the principles of green cryptography, and suggests reusing ideas that have proven their merits, in the specific scheme with respect to security and energy.

3.2. Implementation overview

Simulation tests were carried out on an Intel Core i3 3GHz CPU computer with 3GB of RAM and the 32-bit Windows 7 Home Premium OS. For testing purposes, several performance data streams were collected, including the encryption time and the CPU process time.

3.2.1. Simulation

Simulation represents an efficient way to generate test data rapidly using the appropriate tools. In this work, the Sun Netbeans IDE Platform for Java Application Development was used as the platform of the implementation [71]. Java Cryptography Architecture (JCA) and Java Cryptography Extension (JCE) set of classes were used for the implementation of the desired cryptographic functions [72]. Being a universal, powerful and object-oriented language, Java was considered as the appropriate tool for data gathering. Java class *javax.crypto.Cipher* is the engine class for encryption and decryption services. It provides the functionality of a cryptographic cipher used for encryption and decryption [73].

Cipher objects are obtained by invoking the static method *getInstance()* and requires a transformation string that describes the operation to be performed on the given input. The transformation makes use of the algorithm parameters that include the name of the encryption algorithm, followed by the mode of operation and padding scheme. The transform is of the form *algorithm/mode/padding*. For example, the following is a valid transformation: *“DES/ECB/PKCS5Padding”*. Following, method *update()* is called to pass byte arrays for encryption or decryption [74]. *Encrypt* and *Decrypt* are used for the encryption and the decryption process to produce the *Ciphertext* and the

Plaintext respectively. Finally, the *doFinal()* method must be invoked to complete the cipher operation and reset the Cipher object so that it will be ready for the next encryption process. The implementation process of the class for symmetric encryption and decryption is illustrated in Figure 3.1.

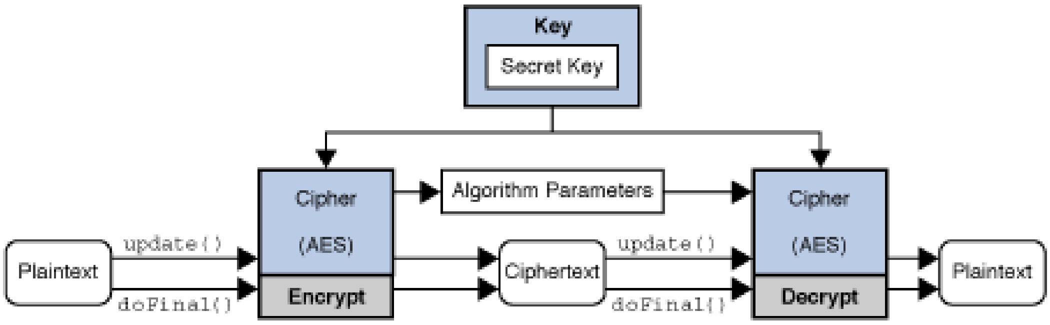

Figure 3.1: Cipher class, retrieved from [73].

The encryption procedure was simulated 100 times for all 576 combinations of four encryption parameters for the five algorithms. The encryption time ranged between 74 µs and 2.7 ms, whereas the energy consumption ranged between 26 nJB^{-1} and 17.6 μJB^{-1}. The energy consumption has been calculated based on the encryption times that resulted from the simulation, using Equation (2.2).

For simulation purposes, five encryption algorithms, namely: AES, DES, 3DES, RC2 and Blowfish, were considered but the method investigated is generic and so would work with any encryption algorithms and functional parameters.

Here, some parameter choices were the same for all algorithms, namely: ECB, CBC, OFB and CFB modes; data block sizes of 16, 1024, 2048 and 4096; padding scheme with NoPadding, ISO101126 and PKCS5. The key sizes used were different for each algorithm, and are shown in Table 3.1.

Table 3.1: Key size variation

Algorithm	**Key size**
AES	128,192, 256
DES	56
3DES	112, 168
Blowfish	56, 112, 256
RC2	40, 64, 128

3.2.2. Data analysis

The R language [75] has been used for data manipulation, calculation and graphical display of the probabilistic approaches. The code is designed to read the encryption times as resulting from the simulation. The graphical displays of the results of the data analysis that are used in the following chapters have been delivered using the R plotting commands and attributes.

In the first approach, based on reliability, stochastic data analysis using the R language was applied to the simulation outputs to provide the reliability metric, facilitating the evaluation of the overall impact of the interrelated encryption parameters on energy consumption and delivering a global performance metric for energy consumption. Specifically,

- In this implementation, the CDF and reliability are calculated for a given time threshold that is set by the user.

- The code is also designed to return the cases that meet the criteria specified by the user, i.e. return the security scenarios that will finish the encryption prior to the set threshold and by a desired probability. An example criterion could be the following: for a time threshold t = 300 µs, identify the cases that will finish the encryption prior to the threshold with probability P ≥ 0.99.

- For the case where the user requests specific security parameters with a specific reliability level, then a threshold has to be set. This optimum threshold selection algorithm is also coded in R.

Regarding the bound approach, stochastic data analysis using the R language was performed on the simulation outputs in order to provide an upper bound on the probability that the mean time of *n* encryptions will exceed the expected time by a desired threshold.

- First the expected and theoretical means are calculated. A Bootstrap method has been used for the generation of the sample.

- The bound is then calculated for the probability that the true and expected means will not deviate more than a desired threshold.

- Based on a point-to-point comparison, the difference between the true and expected mean is calculated.

- The relationship among several attributes, as well as the impact of their variations, is also coded in R.

In the centrality-based approach, data analysis using the R language was also applied to the simulation outputs to measure the significance of the encryption parameters in the algorithms as well as their impact on energy consumption.

- The *betweenness* function is used in order to measure the centrality of the encryption parameters based on the number of shortest paths - in terms of energy consumption - going through each parameter. The shorter the path, the lower energy will be consumed.

- The *igraph* package is used to produce the graphs as an illustration of the betweenness of each encryption parameter.

Finally, in the statistical approach, data analysis was performed using the SPSS (Statistical Package for the Social Sciences) Statistics package [76].

- Multiple linear regression is conducted in order to determine whether energy consumption can be predicted by the encryption parameters.
- Correlations among the variables are taken into account in the estimation of the coefficients.
- The dependencies between encryption parameters are examined.
- Residual plots are used to evaluate the goodness of fit.

3.3. Contribution

Although most approaches compare algorithms based on individual encryption parameters, the performance of the encryption system should be evaluated with respect to the security restrictions. Therefore, the development of a decision-making framework that evaluates the overall impact of each security mode on energy consumption based on a global quality factor is suggested. The distinguishing feature of the work presented in this thesis is the maximization of encryption system performance by energy consumption management, taking into consideration several inter-related factors.

The contribution made here is to study the overall influence of the configuration parameters on the energy consumption regarding either the security requirements or the available resources in terms of energy. Stochastic and statistical considerations in terms of evaluation have been developed in order to conclude the overall effect of the encryption parameters on energy.

To the best of the author’s knowledge, there has been no attempt to study the overall influence of these configuration parameters on the energy consumption of encryption systems. This work is an attempt to analyse and study the overall energy consumption and encryption parameters’ variation and obtain the most effective configuration of the encryption system, for specified security requirements.

Hereafter, it is intended to demonstrate the four novel approaches concerning the energy conscious adaptive security implementation and to quantify the effectiveness of the scheme over the traditional methods.

Chapter 4 - Reliability approach

In this chapter a security framework is presented, based on the reliability function, with the ability to adjust dynamically the security mode with respect to energy consumption. The proposed security approach is twofold, as the security mode can be adjusted either according to the severity of the requested service, or according to a specified energy threshold. The rest of the chapter proceeds as follows. In the first section reliability and its implication in the security scheme will be covered. The chapter then goes on to providing a brief description of the methodology that is adopted in the development of the proposed security scheme. This is followed by the implementation of the framework and the results obtained from the analysis of the security mode performance based on reliability. Finally, for the case of the security adjustment based on the user reliability level requirements, the second option of the twofold approach, the optimum threshold concept and the related algorithm, are illustrated.

This chapter is based on the two following papers:

- "*Energy Conscious Adaptive Security Scheme for Optical Wireless*", published in the proceedings of the 14th IEEE International Conference on Transparent Optical Networks (ICTON), 2012

- "*Energy Conscious Adaptive Security Scheme: A Reliability-based Stochastic Approach*", submitted to the Journal of Performance Evaluation.

4.1. Reliability

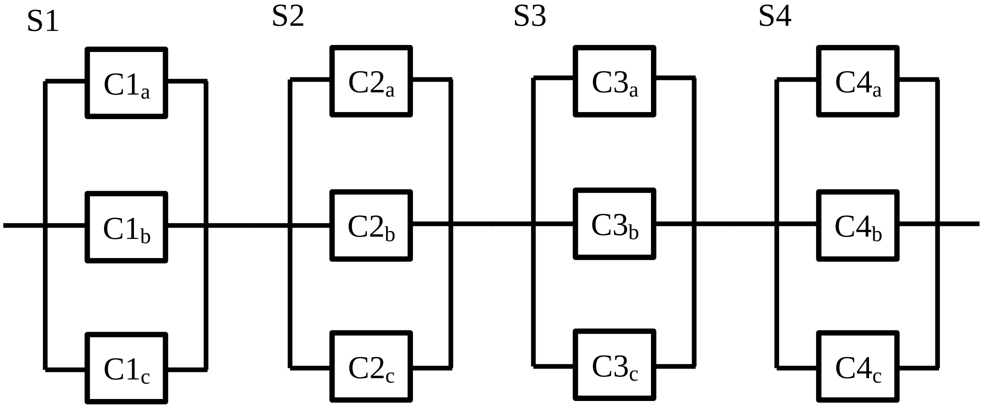

Figure 4.1: Encryption system S.

A reliability function represents the probability that for a given time, the system will survive [56]. For illustration purposes, system *S* is considered that consists of individual subsystems S_1, S_2, S_3 and S_4 connected in series, as shown in Figure 4.1. Those subsystems have independent individual lifetimes, not necessarily coming from the same probability distribution. Each subsystem consists of several components, C_a, C_b, C_c, connected in parallel with each other.

For the above system, the reliability function as shown in Equation (4.1) is:

$$R(t) = P(T > t) = P(T_1 > t\ , T_2 > t\ , T_3 > t\ ) \tag{4.1}$$

where *T* represents the total lifetime of the system, while T_1, T_2 and T_3 stand for the lifetime of subsystems S_1, S_2 and S_3 respectively.

The above reliability function (4.1) can also be expressed as a function of energy, as shown in Equation (4.2):

$$R(e) = P(E > e) = P(E_1 > e\ , E_2 > e\ , E_3 > e\ ) \tag{4.2}$$

where *E* represents the total energy consumption of the system, whereas E_1, E_2 and E_3 stand for the energy consumed by subsystems S_1, S_2 and S_3 respectively, for a given energy e.

An equivalent formulation in terms of encryption can be made by replacing every occurrence of death with the completion of the encryption procedure. Empirical distribution of lifetimes of each encryption mode can be easily measured with several simulations running for all possible combinations. A subsystem's lifetime refers to its execution time when used in the encryption procedure.

By considering *e* as a given energy threshold, the above function could also be applied in an adaptive encryption scheme, as it could be used to derive upper bounds on the minimum operation of the encryption parameters, in order to achieve the required security. Those upper bounds are expected to be monotonically increasing in reliability, suggesting that it is better from an energy efficiency perspective to use relatively less secure primitives. Another way to express the reliability of a system is to use the lifetime distribution function (4.3), which is the complementary probability that derives from the reliability function (4.2). Specifically:

$$L(e) = 1 - R(e) = 1 - P(E > e) = P(E_1 \leq e\ , E_2 \leq e\ , E_3 \leq e\ ) \quad (4.3)$$

Equation (4.3) demonstrates the usefulness of the energy threshold. The use of the lifetime distribution function could be easily adopted and serve as a global indicator of performance.

4.2. Methodology

As mentioned in the previous sections, Reliability, and therefore the CDF function can be used to calculate the probability of survival or failure for a given system respectively. The CDF, can be used to describe the probability that the system will finish its operation prior to a given threshold and therefore indicate the impact of its subsystems' variation on the total system lifetime. In security analysis, the CDF function can easily be adopted and treated as a quality factor that describes all encryption parameters and their impact on energy consumption. It serves as an indicator of the performance of the encryption parameters with respect to the energy consumption of the overall security system. This forms the basis for the proposed adaptive security scheme that extends the fitting of the model for each security mode accordingly, by properly adjusting functional parameters and always taking into consideration the energy cost. In this way, a metric that indicates the impact of all encryption parameters is developed, and thus a global indicator is derived. The proposed model can be thus considered global, as it is not based on distinct parameters, but, instead, arises from the impact of all the individual encryption parameters on energy consumption [4]. According to Equation (2.5), to calculate system reliability one should isolate all encryption parameters and calculate their individual probabilities $P(T_1 > t)$, $P(T_2 > t)$, $P(T_3 > t)$, $P(T_4 > t)$ accordingly. However, it is not easy to derive their individual encryption parameter distributions since these cannot be isolated. Therefore, in this work, several simulation runs have been performed, as described in section 3.2.1, and the results provided the means to determine the empirical CDF (ECDF) of $P(T > t)$. The encryption times of the 576 cases considered in this work have been measured and the ECDF for each security mode has been concluded. Based on the ECDF, the security modes can be compared and evaluated. Depending on the requirements, the selection of the most efficient mode can be made either based on the security restrictions or the energy/time threshold.

```
Variables:  list Cases                 # 576 cases data set
            boolean S                  # security requirements
            boolean E                  # energy requirements
            list $S_r$                 # security restrictions
            float $E_r$                # energy restriction
            float R                    # reliability
            float $R_{min}$ = 1        # lowest reliability
            int efficient = 0          # most efficient case

if S = TRUE then
    for case in Cases
        if case not in $S_r$ then
            delete case
        else
            calculate R
            if $R < R_{min}$
                $R = R_{min}$
                efficient = case
            end if
        end if
    end for
    return efficient
else
    for case in Cases
        if case not in $E_r$ then
            delete case
        else
            calculate R
            if $R \neq 0$
                delete case
            end if
        end if
    end for
    return Cases
end if
```

Figure 4.2: Adaptive security scheme algorithm.

In the first case, the security modes that do not meet the security requirements are excluded. For the rest of the security modes, first the ECDF is determined and then the selection follows based on the ECDF metric. In the second case, the security modes that do not meet the energy/time requirements are excluded. For the rest of the security modes, first the ECDF is determined and then the selection follows based on the ECDF metric. The proposed adaptive security scheme and its operation are described in the adaptive security algorithm, shown in Figure 4.2.

As mentioned earlier, the suggested adaptive security scheme provides two options for achieving the desired encryption strength at the lowest energy cost:

- For given security requirements for the requested service, the reliability function is used to return the most efficient option with respect to energy, for the specific security mode. This can be done by excluding the modes that do not meet the security requirements, and by ranking the modes after the elimination based on the reliability or the ECDF. The higher the ECDF, the highest the probability of finishing the encryption on time.

- In the case of battery powered devices, for a given energy threshold that derives from the battery state, the reliability function is used to return the most efficient option with respect to security, for the specific energy threshold. The modes that do not meet the time/energy requirements are excluded and the rest of the cases are ranked based on the ECDF/reliability and the selection is made based on this ranking.

Overall, the proposed adaptive security scheme consists of several security modes, each providing a different level of security, depending on the severity of the service requested. Each security mode operates using the appropriate security algorithms and/or primitives. As the energy cost depends on the encryption parameters, each policy will induce a different level of energy consumption.

Using the empirical CDF, a probability metric is calculated for a specified energy threshold. In this way, one can either accept or reject the combinations according to the desired level of the probability, and, depending on whether they satisfy the requirements or not, a decision will be made, which implies that the combinations that do not meet the given constraints will be eliminated.

A general rule applied to most of the cases is that the highest probability of completing the encryption procedure prior to the time threshold will be selected, meaning that, for the specified threshold, the system will accomplish complete encryption in the most secure mode possible as well as at the lowest energy cost. Depending on the desired reliability, the most secure option will be selected.

In the case of a battery-level threshold, the security modes that do not meet the energy constraints are excluded and the rest of them are ranked according to their reliability/ECDF. As mentioned earlier, reliability is the probability that the system will continue the encryption process even after the given energy threshold. Therefore, the lower the reliability is, the higher the probability that the system will have finished the encryption procedure before the battery dies. Thus, the probability will be used to return the most efficient option with respect to security, for the specific energy threshold. In the case of specific security requirements, the probability will be used to return the most efficient option with respect to energy, for the specific security. Thus, the system will select the lowest reliability metric from the available options that meet the requirements of the desired security.

4.3. Implementation and results

In what follows, examples of how the reliability function can be used in an adaptive security scheme are presented. By setting a time threshold t for the encryption procedure, one can exclude the cases that do not meet the time or energy constraints and, therefore, the energy limitations that derive either from the available resources or the energy saving requirements. Since the probability that the system will have finished the encryption procedure on time is given by the ECDF: the higher the ECDF, the greater the probability of success and hence of energy consumption less than or equal to the level desired. Given that T represents the total lifetime - or the total encryption time of the encryption system, the highest probability that the encryption time *T* is less than or equal to *t* is desired:

$$ECDF(t) = F(t) = P(T \leq t) \rightarrow 1 \tag{4.4}$$

As a consequence, the higher the reliability, the higher is the probability that the system will continue the encryption after the specified time threshold *t*. Thus, a reliability equal to 0 would be the optimal probability, since it is desired that the system will operate for as little time as possible, and therefore consume the lowest energy possible:

$$R(t) = P(T > t) \rightarrow 0 \tag{4.5}$$

4.3.1. Example 1: Upper quartile time threshold

To illustrate the implementation concept, a time threshold *t* will firstly be assigned. In the specific example the threshold is set to be in the upper quartile of the data. This means that the ordered data is divided into four equal-sized subsets, and the quartile is the point taken at four intervals from the CDF of the variable, marking the boundaries between the consecutive subsets. In other words, the threshold is set equal to the value that represents the boundary of the quartile that lasts the longest time during encryption. It has to be noted, however, that one could also set a specific threshold, based on either the desired time or energy. For illustration purposes, Table 4.1 shows a sample of 6 of the 576-case (security modes) data set that resulted from the simulation, along with the variations of the parameters and the mean encryption time and energy consumption for the 100 iterations executed for each algorithm.

The ECDF and reliability probabilities are shown, for the specified threshold that resulted in a time threshold of t = 864 μs, or an energy threshold of e = 233 μJ from Equation (2.2), corresponding to 56 nJB^{-1} for the encryption of a 4096 byte data set. As shown in the table, Reliability is very low for all of the cases of the random sample. This was expected, as the time threshold is well above the overall mean encryption time.

Table 4.1: Simulated data sample

Security mode	Algorithm	Mode	Key	Data	Padding	ECDF	Reliability	Mean time	Mean energy
9	AES	CBC	256	4096	No	0.99	0.01	534	35
16	AES	CFB	128	4096	No	1	0	493	32
135	DES	CBC	56	4096	ISO	0.96	0.04	772	50
215	3DES	OFB	112	4096	ISO	0	1	1596	105
279	BF	ECB	256	4096	PKCS5	0.98	0.02	608	40
342	RC2	CFB	128	4096	No	0.98	0.02	652	42

By examining the results in Table 4.1, it may be seen that case 215 will be eliminated, as it exceeds the specified threshold, which also appears as an ECDF equal to zero, meaning that almost surely the encryption will not be finished before the requested threshold. The other cases offer some of the available options that satisfy the threshold requirements. Depending either on the energy consumption or the desired level of probability, the security option that best satisfies the requirements can be selected.

4.3.2. Example 2: 500 μs time threshold

When a set time threshold is desired, this can be set by the user and then the calculation of the reliability metric for all the available security modes can proceed. For the specified time threshold *t* = 500 μs, some of the cases that will complete the encryption procedure prior to t with a desired lifetime ≥0.97, are shown in Table 4.2. The ECDF of the five cases that satisfy the requirements for the specified time threshold are presented.

Table 4.2: Simulated data sample for a set threshold of 500 μs

Case	**Algorithm**	**Mode**	**Key**	**Data**	**Padding**	**ECDF**	**Reliability**	**Mean time (μs)**	**Mean Energy (nJB^{-1})**
4	AES	CBC	128	2048	NoPadding	1	0	293	38
6	AES	CBC	256	2048	NoPadding	0.99	0.01	324	42
119	DES	OFB	56	2048	NoPadding	0.97	0.03	444	58
312	BF	ECB	256	2048	ISO	0.97	0.03	427	56
374	RC2	CFB	64	2048	PKCS5	0.97	0.03	379	50

Figure 4.3 depicts the behaviour of the estimated ECDF function which depends on

the encryption time as taken from the simulation for the five cases mentioned above. The area on the left side of the vertical line - which is the specified time threshold t - represents the probability that for the given time threshold the encryption procedure will be completed. Specifically, analysing the ECDF probability as illustrated in Figure 4.3, the following results may be extracted: Case 4, P(T ≤ 500) = 1; Case 6, P(T ≤ 500) = 0.99; cases 119, 312 and 374, P(T ≤ 500) = 0.97.

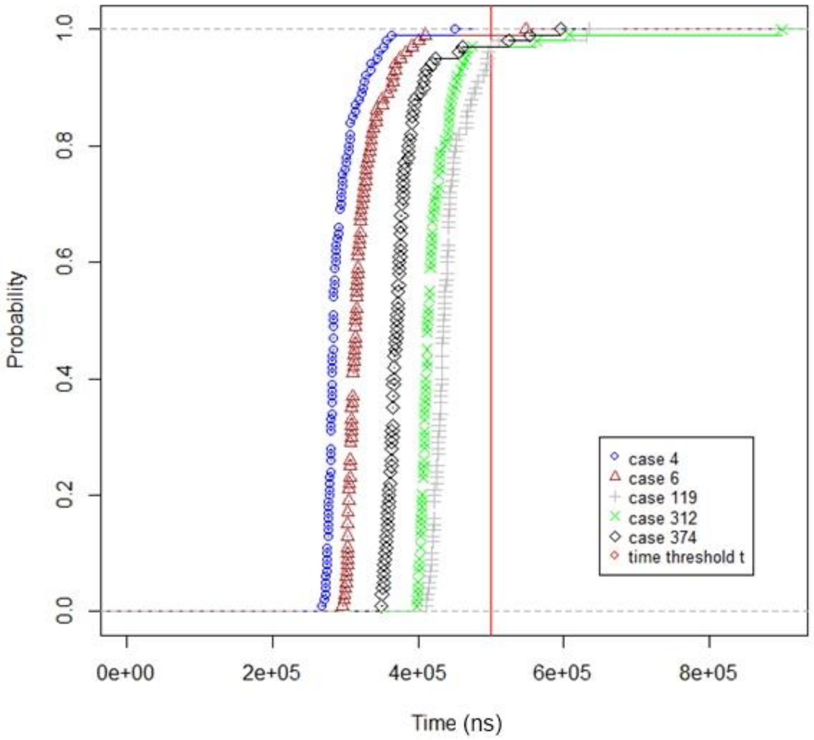

Figure 4.3: ECDF for sample cases 4, 6, 119, 321, 374.

Figure 4.4 shows the performance of Case 4 in terms of encryption time. The resulting ECDF and reliability are presented, illustrating their reciprocal relationship. For the specified time threshold *t* = 500 μs, the ECDF tends to 1, while the reliability function tends to 0. This can be easily explained by comparing Figure 4.3 and Figure 4.4, where one can observe that regarding the ECDF, which describes the probability of the encryption time, the maximum observed time is 450 μs. For this reason, the ECDF probability tends to 1, while reliability tends to 0.

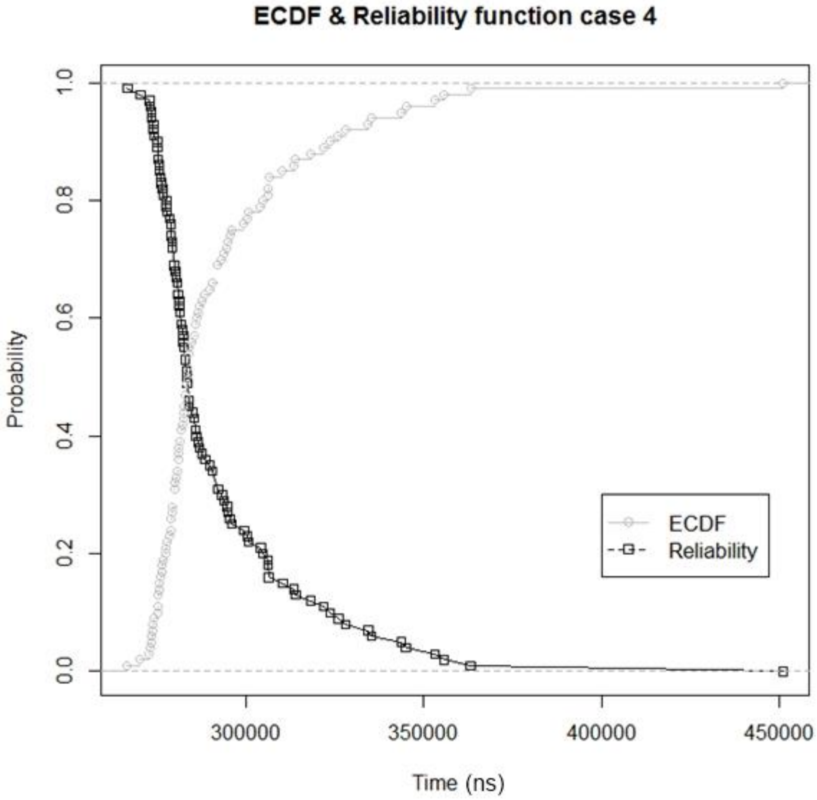

Figure 4.4: ECDF and Reliability – Case 4.

Following, an ideal encryption performance scenario, where the ECDF is 1 (and reliability is 0) for a time threshold *t*, is considered. A time threshold may be set, such that P(T ≤ t) = 1, which will here be 550 µs. For example, considering case 6 from the previous scenario, the probability of finishing before 500 µs is 0.99; increasing the time threshold by 50 µs makes the probability that Case 6 will finish before 550 µs, equal to 1. Translating the 500 µs encryption time to energy, this is equal to 135 µJ. Similarly, translating the 550 µs encryption time to energy, results in 150 µJ. Therefore, the 10% increase in the time threshold will incur an energy cost of 100-135*100/150=10% increase, which is 150-135=15 µJ. This, once again, reinforces the idea of the energy consumption investigation under certain security requirements, which offers the user the option for the optimal security mode selection at the lowest energy cost.

The proposed adaptive security scheme evaluates the performance of several encryption algorithms and functional variation of their parameters with respect to energy consumption. Its methodology includes all possible combinations of the encryption functional parameters ranked with regard to the quality of the security, whilst also allowing for sorting security modes with respect to the level of energy consumption. At its most fundamental, this scheme determines a probability for each combination of functional parameters based on their impact on energy consumption. In Table 4.2 for example, one can not only see the ranking of a sample of the 576 cases, but also a quantitative comparison of the latter, i.e. Case 4 is 3% more likely to finish the encryption prior to the time threshold *t*, than case 119, and so on.

4.4. Optimum threshold

As described in the previous sections, one can adjust the level of security in the proposed scheme, based either on the desired threshold for energy restricted cases, or on the security requirements. In the second case, the reliability metric calculation reduces the adjustment of the security mode to the setting of an appropriate threshold for efficient probability estimation to meet user reliability level requirements. Hence, it is of great importance to develop a model that returns the optimum threshold, and the following subsection addresses this issue.

When the security mode has to be adjusted according to the severity of the requested service, the system needs to deliver a specific likelihood that the encryption procedure will have finished before a time threshold. Although threshold selection could be random or set at the beginning of the operation, here the optimum threshold is computed for each security mode independently. This offers optimized performance, since the respective reliability metric is concluded for each security mode, based on the individual optimum threshold, allowing for a more realistic, precise and rational evaluation of the reliability metric.

4.4.1. Proposed algorithm

The proposed algorithm calculates the optimum threshold for each security mode as follows. Using binary search, the time that results in the desired reliability level is found. Therefore, the list that the encryption times are stored has to be sorted in ascending order. Staring from the middle element of the list, the algorithm calculates the reliability and if it is equal to the requested one, the encryption time of this element is used as the optimum threshold for the next part of the algorithm. If reliability is greater than the requested one, the search is repeated for the first half of the list. If reliability is less than the requested one, the search is repeated for the second half of the list. This process is repeated until the level of reliability is equal to the requested one. There is also an option to adjust the decimal precision of the threshold. In this case, after the above procedure is completed, the algorithm subtracts one decimal point of the minimum time unit the reliability requirements are met. The overall optimum threshold selection procedure proposed in this section is summarized in the above algorithm as shown in Figure 4.5. It requires the choice of a reliability level together with the exact decimal precision and starts from the minimum observed encryption time.

```
Variables:   list Times          # observed encryption times
             float $R$           # reliability
             float $R_{req}$     # requested reliability
             int $d$             # number of decimal precision
             float $j = 1$       # initialized at 1

Search ( list, k, low, high )
    mid = ( low + high ) / 2
    Calculate $R$                # for key(mid)
    if $k = R$ then
        $t$ = key ( mid + 1 )
    else if $k < R$ then
        Search ( list, k, low, mid – 1 )
    else
        Search ( list, k, mid + 1, high )
    end if
    return $t$
end Search

Search ( Times, $R_{req}$, Times[1], Times[n] )

for $i = 1$ to $i = d$
    $j = j / 10$
    Calculate $R$                # for t
    while $R > R_{req}$
        $t = t – j$
        calculate $R$
    end while
end for

return $t$
```

Figure 4.5: Optimum threshold selection algorithm.

4.4.1.1. Example 1: R ≤ 5% - Case 4

Figure 4.6 gives an example where a user has requested the optimum threshold for security mode 4 that offers a probability of ≤5% that the encryption will not be completed prior to that threshold. This means that the ECDF lifetime metric has to be >0.95. Using the algorithm described in the previous section, the optimum threshold for the specific example was found equal to 345.014 μs. As shown in Figure 4.6, the encryption procedure will be finished prior to the threshold by 0.95.

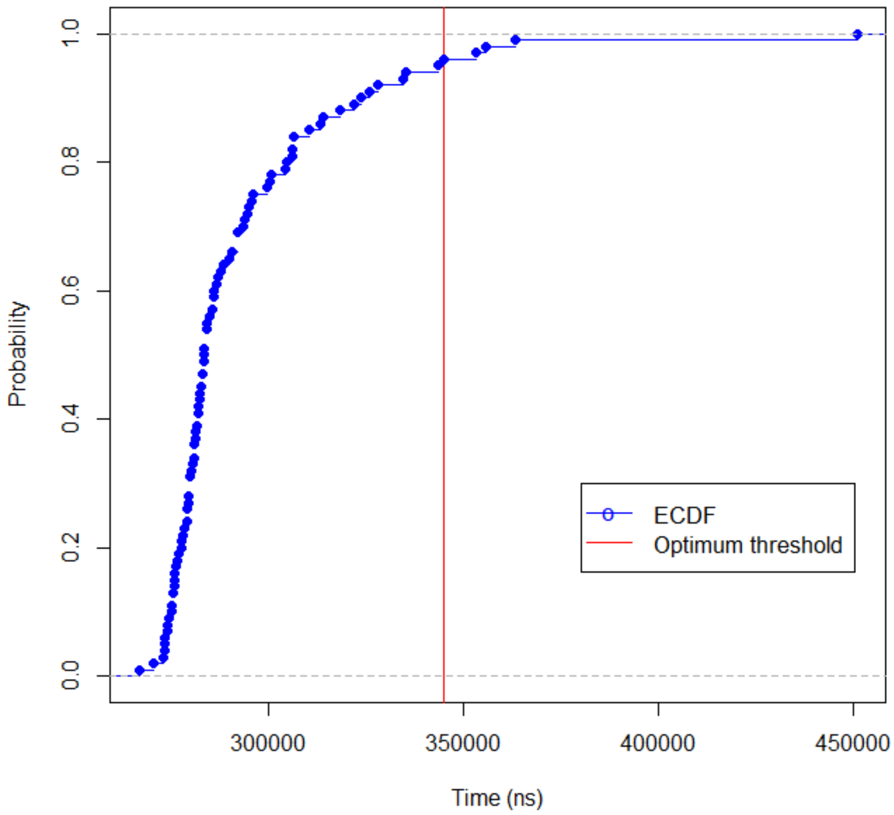

Figure 4.6: ECDF and Optimum threshold for R ≤ 0.05 – Case 4.

4.4.1.2. Example 2: R ≤ 5% - Case 6

Another example is illustrated in Figure 4.7, where the user has requested the optimum threshold for security mode 6. The desired reliability is ≤5%. Using the algorithm for the optimum threshold selection, it was found that t_{opt} = 383.566 μs. As shown in Figure 4.7, the encryption procedure will be finished prior to the threshold with probability 0.95.

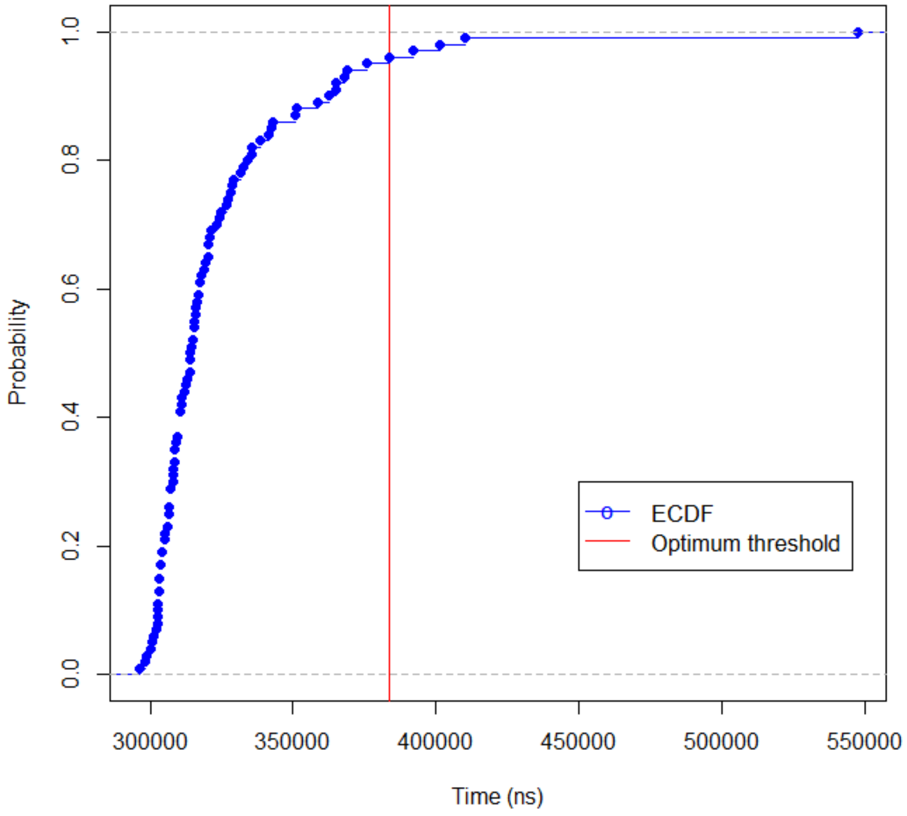

Figure 4.7: ECDF and Optimum threshold for R ≤ 0.05 – Case 6.

4.4.1.3. Example 3: R ≤ 3% - Case 6

Finally, for security mode 6 and for a desired reliability level ≤0.03 according to the proposed optimum threshold selection algorithm, t_{opt} was found to be equal to 401.166 μs. Figure 4.8 illustrates the optimum threshold for this case. Compared to the previous example, where for a desired reliability level ≤0.05 t_{opt} = 383.566 μs, it can be observed that, by increasing the desired probability of not exceeding the time threshold by 2%, the optimum threshold is increased by 4.7%.

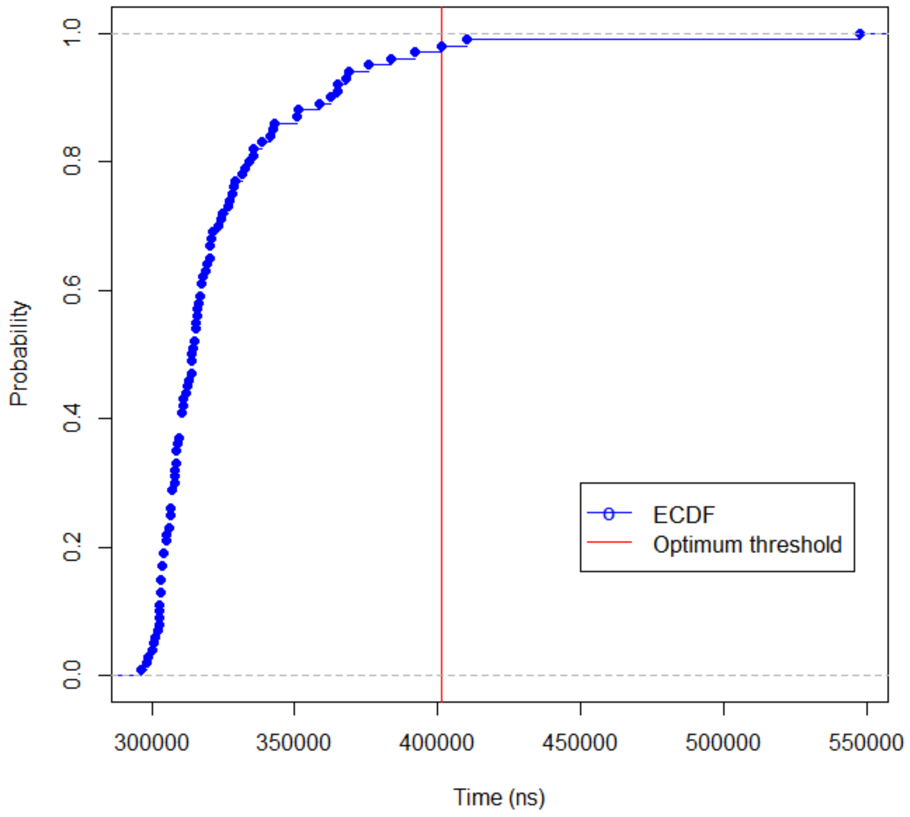

Figure 4.8: ECDF and Optimum threshold for R ≤ 0.03 – Case 6.

4.5. Results

When a system has not been configured to differentiate between the security hierarchy of the requested services, all the propagated data will be encrypted using the same encryption scheme. Thus, the mode that meets the requirements of the most crucial service is selected, so that an adequate level of security is guaranteed. However, it is not always necessary to encrypt data with a higher level of security strength than is actually needed, as this might result in unnecessary time and energy consumption.

In the second example presented in the previous section, for the encryption of a 2 kB of data, the user aims to encrypt prior to the 500 μs threshold, with probability ≥ 0.97. Consider that apart from the time/energy requirements, the user also desires a high level of security. Assuming that AES is adequate for the user security requirements, according to Table 4.2, the available options for this encryption process are Cases 4 and 6, which differ only in the key size. However, given that both options provide an adequate level of security, Case 4 runs at a saving of 4nJB^{-1}. Although this might be negligible for one encryption, using the appropriate parameters could save a significant amount of energy over a large number of encryptions, and this section now presents the concept and examines the results.

Let n be the number of encryptions and X be a r.v with mean μ and variance σ^2 that represents the encryption time of a security mode.

Let $S = \sum_{i=1}^{n} X_i$ be the overall encryption time of the n encryptions. Since $X_i, \dots, X_n$ are i.i.d., from Central Limit Theorem (CLT), S approaches a normal distribution. Hence, the ECDF of $\hat{S}$ converges in distribution to

$$S \sim N(n\mu,\ n\sigma^2) \text{ as } n \to \infty \qquad (4.6)$$

From Equation (4.6), it can be derived that for Case 4 in Table 4.2, $\widehat{S_4}$ converges to $S_4 {\sim} N(n\mu_4,\ n\sigma_4^2)$ as $n \to \infty$, where $n = 1000$, $\mu_4 = 2.93 \times 10^5$ns, $\sigma_4^2 = 6.61 \times 10^8$ ns, $\mu_{S_4} = n\mu_4 = 2.9 \times 10^8$ ns and $\sigma_{S_4}^2 = n\sigma_4^2 = 6.61 \times 10^{11}$ ns.

Figure 4.9 illustrates a point to point comparison of the theoretical distribution of S_4 and the approximate of $\widehat{S_4}$ as generated from $\widehat{S_4} = \sum_{i=1}^{n} X_{i_j}$, where $j \in \{1, \dots, m = 10000\}$ and $X_i {\sim} ECDF_4$ with mean μ_i and variance $\sigma_i^2,\ \forall i \in \{1, \dots, n\}$. As shown in the histogram, $\widehat{S_4}$ is distributed evenly around the mean, with most of the frequencies gathered in the centre, indicating that $\widehat{S_4}$ follows the Normal distribution. Hence, the approximation of $\widehat{S_4}$ is good, since the theoretical density maps the histogram. The

Q-Q plot indicates that $\widehat{S_4}$ follows the normal curve as well since the data points lie close to the diagonal line.

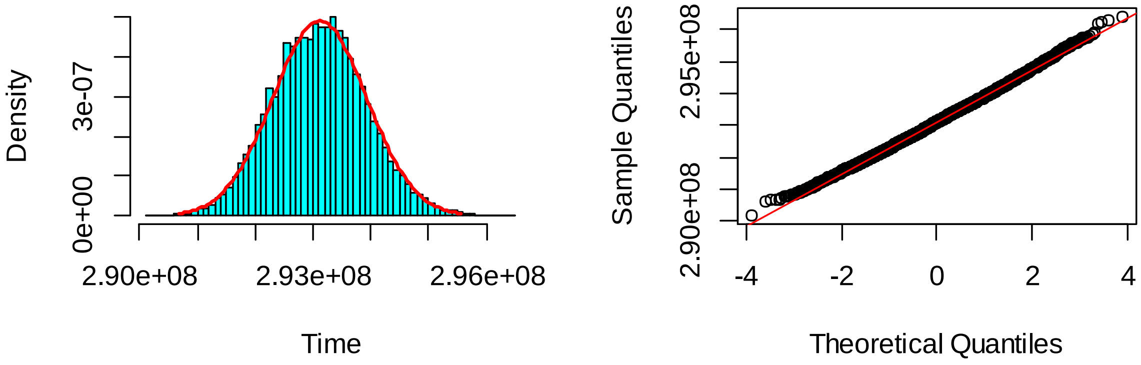

Figure 4.9: Theoretical S_4 vs estimated $\widehat{S_4}$ distribution.

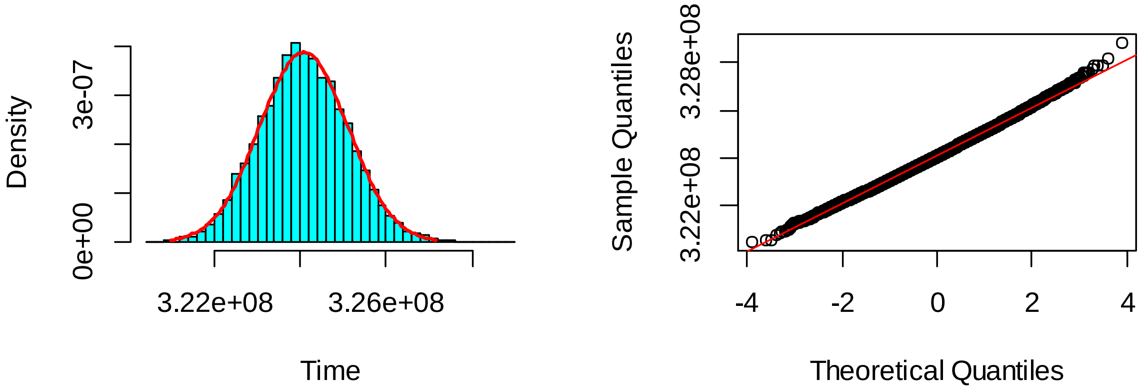

Figure 4.10: Theoretical S_6 vs estimated $\widehat{S_6}$ distribution.

Similarly, for Case 6, $\widehat{S_6}$ converges to $S_6 \sim N(n\mu_6,\ n\sigma_6^2)$ as $n \to \infty$, where $n = 1000$, $\mu_6 = 3.24 \times 10^5$ ns, $\sigma_6^2 = 1.05 \times 10^9$ ns, $\mu_{S_6} = n\mu_6 = 3.24 \times 10^8$ ns and $\sigma_{S_6}^2 = n\sigma_6^2 = 1.05 \times 10^{12}$ ns.

Figure 4.10 illustrates a comparison of the theoretical distribution of S_6 and the approximate version $\widehat{S_6}$ as generated from $\widehat{S_6} = \sum_{i=1}^{n} Y_{i_j}$, where $j \in \{1, \ldots, m = 10000\}$ and $Y_i \sim ECDF_6$ with mean μ_i and variance σ_i^2, $\forall i \in \{1, \ldots, n\}$.

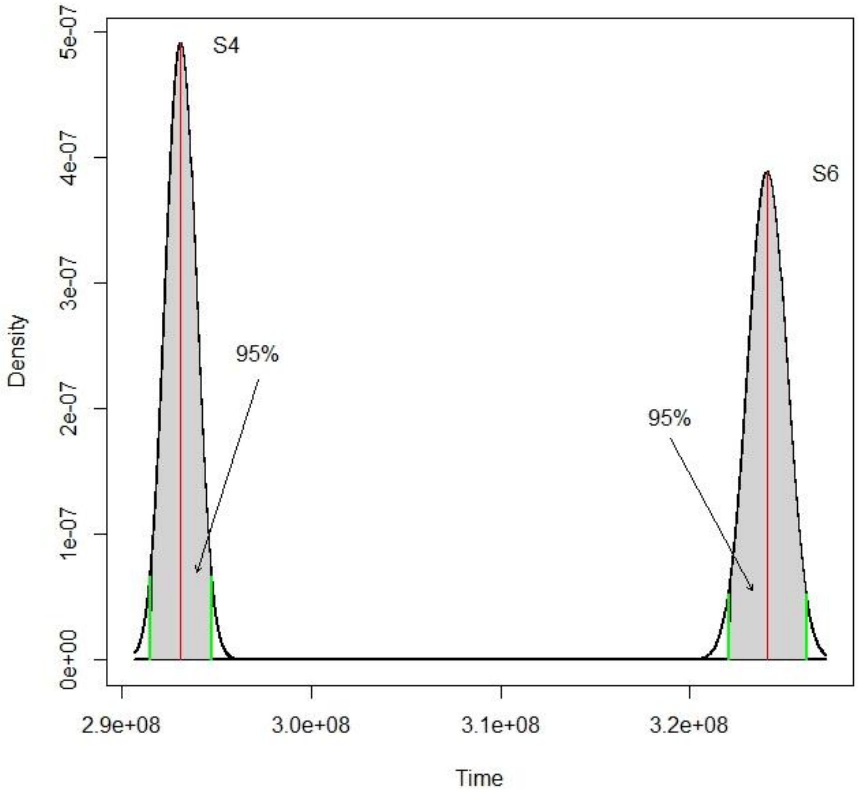

Figure 4.11: S_4 vs S_6 density plot.

Again, the figure indicates that the distribution of the sum approaches a normal distribution. The Q-Q plot indicates that $\widehat{S_6}$ is a good fit as well, as the data points do not deviate from the diagonal line.

Therefore, by fixing all encryption parameters that meet the user requirements and by distinguishing the key size, Cases 4 and 6 are compared. Figure 4.11 illustrates the contrast of the two encryption modes.

As expected, S_4 has a smaller mean compared to S_6, $\mu_{S_4} = 2.93 \times 10^8 s < \mu_{S_6} = 3.2410^8 s$, as well as smaller variance, $\sigma_{S_4}^2 = 6.61 \times 10^{11} s^2 < \sigma_{S_6}^2 = 1.05 \times 10^{12} s^2$.

In terms of n encryptions, this difference could be translated to 10% more time for encryption with mode 6 than with mode 4. In addition, for an observation that follows the distribution of S_4, the probability that the overall encryption time of n services will take values from the following ranges is:

$$P\left(S_4 \in \left(\mu_{S_4} - 3\sigma_{S_4},\ \mu_{S_4} + 3\sigma_{S_4}\right)\right) \approx\ 0.99 \tag{4.7}$$

$$P\left(S_4 \in \left(\mu_{S_4} - 2\sigma_{S_4},\ \mu_{S_4} + 2\sigma_{S_4}\right)\right) \approx\ 0.95 \tag{4.8}$$

$$P\left(S_4 \in \left(\mu_{S_4} - \sigma_{S_4},\ \mu_{S_4} + \sigma_{S_4}\right)\right) \approx\ 0.68 \tag{4.9}$$

As it has been shown, by encrypting n times under Case 6 parameterization, it is expected that the overall encryption time will be 10% higher than the Case 4 parameterization. Knowledge of the distributions of S_4 and S_6 provides further understanding regarding the deviation of the encryption time from the mean by computing the confidence intervals (4.7-4.9). There follows an analysis that will enable the user not only to rank security cases, but also quantify and mathematically evaluate the selection among the available options. This will allow a user to predict the encryption time/energy saving he could achieve and make inference on how likely his predictions are to be true. Therefore, the distribution of the difference between the time of n encryptions form Case 4 and 6 will be investigated.

Let $W_1 = S_6 - S_4$ be a random variable that represents the difference of two independent random variables, where $S_6 \sim N(n\mu_6,\ n\sigma_6^2)$ and $S_4 \sim N(n\mu_4,\ n\sigma_4^2)$.

The characteristic function of a r.v. X is defined by

$$\varphi_x(t) = E\left(e^{itX}\right) \tag{4.10}$$

and has the property that uniquely characterizes the probability function of X [78]. Hence, from Equation (4.10) the characteristic function of a normal r.v. with expected value μ and variance σ^2 is given by [78].

$$\varphi_x(t) = \exp(it\mu - \tfrac{\sigma^2 t^2}{2}) \qquad (4.11)$$

Thus, from Equation (4.11),

$$\varphi_{W_1}(t) = \varphi_{S_6 - S_4}(t)$$

$$= \varphi_{S_6}(t)\ \varphi_{S_4}(t) \qquad \text{(independence)(4.12)}$$

Also, by symmetry $-S_4 \sim N(-n\mu_4,\ n\sigma_4^2)$ [79] and Equation (4.12), results in

$$\varphi_{W_1}(t) = \exp\left\{itn\mu_6 - n\sigma_6^2 \frac{t^2}{2}\right\} \cdot \exp\left\{itn\mu_4 - n\sigma_4^2 \frac{t^2}{2}\right\}$$

$$= \exp\left\{itn\mu_6 - n\sigma_6^2 \frac{t^2}{2} + itn\mu_4 - n\sigma_4^2 \frac{t^2}{2}\right\}$$

$$= \exp\{itn(\mu_6 - \mu_4) - t^2 n(\sigma_6^2 + \sigma_4^2)\} \qquad (4.13)$$

Hence, from (4.13), W_1 follows the normal distribution

$$W_1 \sim N(n(\mu_6 - \mu_4),\ n(\sigma_6^2 + \sigma_4^2)) \qquad (4.14)$$

The distribution of the difference between the time of n encryptions from Cases 4 and 6 is illustrated in Figure 4.12. It is shown that 95% of the shaded area is inside the range $\mu_{W_1} \pm 2\sigma_{W_1} = (2.8 \times 10^7 s,\ 3.3 \times 10^7 s)$ as stated in Equation (4.9), whilst from Equation (4.8), 99% of the area under the curve lies within $\mu_{W_1} \pm 3\sigma_{W_1} = (2.7 \times 10^7 s,\ 3.4 \times 10^7 s)$. This reveals that the likelihood of n

services that are encrypted using security mode 4 finish prior to n services encrypted using security mode 6, is a rather rare event, since $P(W_1 < 0) \rightarrow 0$.

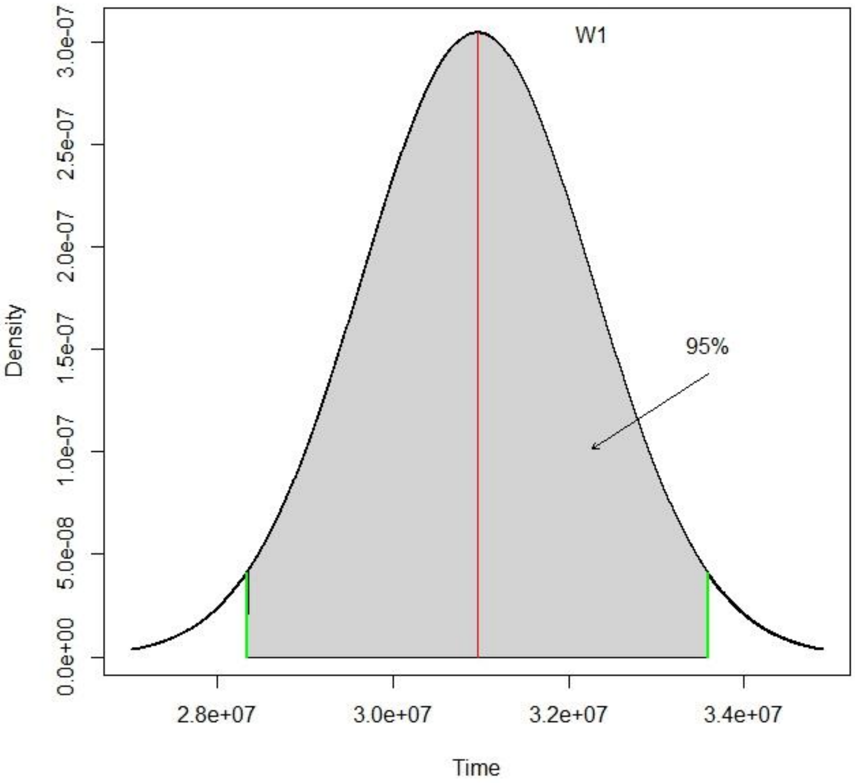

Figure 4.12: W_1 density plot.

As expected, the results show that between S_4 and S_6, S_4 should be selected for services whose security requirements are satisfied, since it proves greater efficiency than S_6. There now follows an examination of an adaptive scenario to illustrate the adaptability of the proposed scheme. In the scenario, the user has requested k services to be encrypted using mode 4 and $(n-k)$ using mode 6.

Let $Q_1 = Z_6 + Z_4 = \sum_{i=1}^{k} X_i + \sum_{j=1}^{n-k} Y_j$

where $Z_6 = \sum_{i=1}^{k} X_i \sim N(k\mu_{6,}\, k\sigma_6^2),\ Z_4 = \sum_{j=1}^{n-k} Y_j \sim N((n-k)\mu_4, (n-k)\sigma_4^2),$

$X_i \sim ECDF_6,\ i \in \{1, \dots, k\}, with\ \mu_6\,, \sigma_6^2 < \infty$

$Y_j \sim ECDF_4,\ j \in \{1, \dots, n-k\}, with\ \mu_4, \sigma_4^2 < \infty$

In addition, X_i are i.i.d. , Y_j are i.i.d. and X_i, Y_j independent $\forall i, j$. Similar to Equation (4.12) and by independence and because

$$\varphi_{Q_1}(t) = \varphi_{Z_6}(t) \cdot \varphi_{Z_4}(t)$$

$$= exp\left\{itk\mu_6 - k{\sigma_6}^2\frac{t^2}{2}\right\} \cdot exp\left\{it(n-k)\mu_4 - (n-k){\sigma_4}^2\frac{t^2}{2}\right\}$$

$$= exp\left\{itk\mu_6 - k{\sigma_6}^2\frac{t^2}{2} + it(n-k)\mu_4 - (n-k){\sigma_4}^2\frac{t^2}{2}\right\}$$

$$= exp\left\{it(k\mu_6 + (n-k)\mu_4) - \frac{t^2}{2}(k{\sigma_6}^2 + (n-k){\sigma_4}^2)\right\}$$

the overall encryption time $Q_1$of the compound scenario, is distributed according to

$$Q_1 \sim N(k\mu_6 + (n-k)\mu_4,\ k\sigma_6^2 + (n-k)\sigma_4^2) \tag{4.15}$$

The density of Q$_1$ for k = 200 encryptions under mode 4 and $n-k$ = 800 encryptions under mode 6 is illustrated in Figure 4.13. For the compound mode, it is expected that the overall encryption time will be 8% higher than mode 4 and 2% less than mode 6. Further, from Equation (4.8), the time interval that assures the user's overall encryption time will lie within $(3.16 \times 10^8 s,\ \ 3.2 \times 10^8 s)$ with probability 0.95. This provides statistical confidence that with high probability the right 2.5% tail of Q_1 will not overlap with the left 2.5% tail of S_6, since

$$P\left(Q_1 > \mu_{S_6} - 2\sigma_{S_6}\right) \approx 0.6 \times 10^{-10} \to 0 \tag{4.16}$$

and

$$P\left(S_6 < \mu_{Q_1} + 2\sigma_{Q_1}\right) \approx 0.7 \times 10^{-20} \to 0 \tag{4.17}$$

Hence, with a 95% probabilistic level of confidence, time predictions belonging to the set of the 2.5% best case scenarios for S_6 do not to overlap with those lying in the 2.5% worst case scenarios for the compound mode.

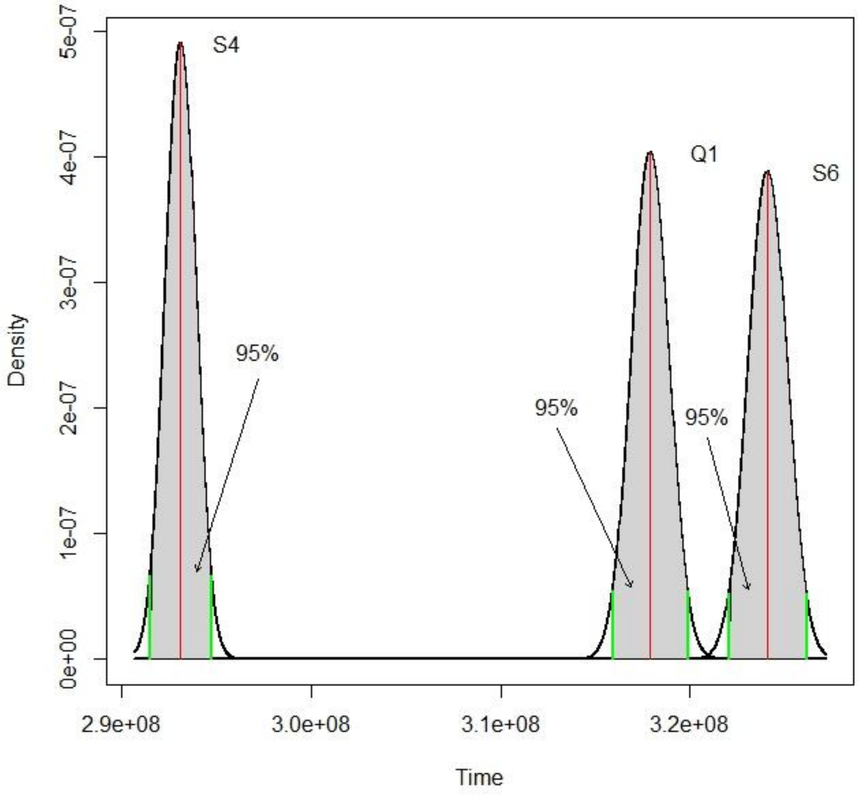

Figure 4.13: Q_1, S_4, S_6 density plot.

Therefore, S_4 and S_6 can be considered as benchmarks for the user customization options and decisions regarding the mode selection, since the distributions of S_4 and S_6 provide an upper and lower bound on the customization of security. The user can make inferences and predict the expected times for the different encryptions according to the severity of each service. Depending on the allocation of the k and $n-k$ services to different encryption modes, the user can therefore customize security according to need.

4.6. Discussion

In this chapter an adaptive security scheme that suggests a new approach in the area of low energy encryption has been proposed. The method relies on the use of the CDF as a global performance indicator. The performance of five encryption algorithms has been evaluated on the basis of the encryption time, energy consumption and the encryption parameter variation, taking into consideration the overall impact of the encryption parameters on energy consumption. CDF has been used as a global indicator for the optimal security mode selection among algorithms and encryption parameters. An adaptive security scheme has been suggested that results in the most efficient security mode, at the lowest energy cost. Furthermore, in this work an optimum threshold selection algorithm has been introduced and developed. This is based on the reliability metric and provides the solution to the problem associated with the selection of the threshold in the case where the security mode has to be adjusted according to the severity of the requested service, and, therefore, the system is required to operate for a specific reliability level.

The asymptotic distribution of n encryptions of the two cases that were assumed to meet user's security requirements has been investigated. The approximate distribution of the overall execution time of n encryptions as calculating by applying CLT, is a Normal distribution $N(\mu, \sigma^2)$ with parameters μ equal to n times the mean execution time of a single encryption and σ^2equal n times the variance of the execution time of a single encryption. It has to be noted that the general form of the Normal distribution as shown in Equation (4.6) is applicable for any case by properly adjusting the parameters μ and σ^2. In addition, this can be applied to the compound scenario, as calculated in Equation (4.15). Furthermore, the distribution of the difference between the time of n encryptions from Case 4 and Case 6 has been calculated. From CLT, the calculated distribution is Normal with parameters as shown in Equation (4.14). This allows a user not only to rank security modes, but also to quantify and mathematically evaluate the selection among the available options. Finally, by using the asymptotic distribution of Case 4 and Case 6 as benchmarks, the derived distribution of a compound policy has been analyzed, with respect to time/energy saving.

This was the first part of the stochastic approach introduced earlier in this thesis concerning the energy conscious adaptive security implementation. Chapter 5 extends the reliability approach with the use of probability inequalities, so that predictions for a finite number of encryptions can be achieved.

Chapter 5 - Probabilistic bound approach

The objective of this chapter is to extend the reliability framework proposed in Chapter 4, where reliability was adopted to deliver a global quality factor for optimal security mode selection, with respect to energy consumption. Here, the focus is on the aspects of bounding the probability that the mean time of *n* encryptions will exceed the expected time by a desired threshold. Similarly, the probability that the overall encryption time will exceed the expected time is also bounded. The advantage of such a model is that predictions outside the range of the most frequently observed values can be made.

A probabilistic upper bound-based approach is introduced and evaluated on the same experiment as in Chapter 4 [4, 80] in order to develop a framework for energy conscious adaptive security. The bound approach relies on stochastic modelling and probabilistic decision making to bound the tail distribution of *n* encryptions. The key contribution of this work is a new bound-based approach that stochastically studies the overall influence of the configuration parameters on the total energy consumption. Chernoff-type bounds are applied to the probability that the encryption time will exceed a given threshold, so as to return the most effective combination regarding either the desired security or the available energy resources for *n* encryptions.

5.1. Bounds on the tail distribution

In addition to the reliability model described in the previous chapter, the ideal model should also have the ability to provide a bound on the tail distribution of *n* samples. In what follows, Chernoff-type bounds are applied in order to examine the impact of the encryption parameters on the overall energy consumption for n encryptions.

The advantage of such a model is that predictions outside the range of those most frequently observed can be made. Several models may fit well the most frequent values of the observed data, but vary considerably in the tails of the distributions of the variable of interest. Here, this is the longest encryption time, naturally leading to the consideration of the extreme values. Thus, to evaluate the effectiveness of the model proposed, the probability that the mean time for n encryptions will exceed the expected time by a desired threshold, as well as the probability that the overall time for n encryptions will exceed n times the expected time by a desired threshold, should be bounded.

Chernoff's bounds estimate the tail probability of random variables and give exponentially decreasing bounds on tail probabilities [82] that

$$P(T > t) \leq e^{-h(t)} \tag{5.1}$$

where $h(t)$ is a function of t such that for $\theta \geq 0$, the supremum of function h is given by $h(t) = sup_{\theta}\{n\theta t - nlogM(\theta)\}$, where $M(\theta) = E(e^{\theta T})$ is the moment generating function of T.

Let $T_i, \ldots., T_n$ be i.i.d. r.v. so that

$$T = \sum_{i=1}^{n} T_i \tag{5.2}$$

Then, from the Chernoff-Cramer inequality [81]

$$P(T \geq nt) \leq e^{nh(t)} \tag{5.3}$$

In this work, as there is no knowledge concerning the theoretical distribution, the moment generating function cannot be computed and thus the original form of the Chernoff bound cannot be applied. Instead, Chebyshev's and Hoeffding's inequality [82] is going to be used as a starting point to provide a bound for the tail distribution. In contrast to other popular concentration inequality methods, Hoeffding's inequality can be applied to arbitrary distributions. Furthermore, Bennett's and Bernstein's inequalities have been used to extend the initial bound. By utilizing knowledge about the variance of the distribution, tighter bounds can be derived.

In this section, an upper bound on the probability that the encryption time of n trials exceeds the time threshold is presented, which is considered as the objective function to be minimized. The aim is to find an upper bound for the probability that the mean time of n encryption procedures will exceed the expected time by a given threshold. Due to symmetry, a two-sided version of those bounds has also been considered and investigated.

To establish the bound and compute the probability that *T* deviates significantly from *t*, $f(T) = e^{sT}$ is used, where f is a function of *T* that, after applying Markov's inequality, allows for expressing the bound as a function of the moment generating function. This methodology is due to Chernoff [82] and is based on finding the value of s that minimizes the upper bound for $s > 0$,

$$P(T > t) = P(e^{sT} > e^{st}) \tag{5.4}$$

According to Markov's inequality, for a nonnegative random variable T and $s > 0$

$$P(T > t) \leq \frac{E[e^{sT}]}{e^{st}} \tag{5.5}$$

In the following subsections, the bound calculation for $n>1$ is presented. The calculated bound is then applied to the data to obtain the bounded tail probability.

5.2. Bound calculation – Chebyshev's inequality

Let $T_1, \dots, T_n$ be i.i.d. random variables so that:

$$\hat{\mu} = \tfrac{1}{n}\textstyle\sum_{i=1}^{n} T_i \tag{5.6}$$

and

$$\mu = E[\hat{\mu}] = \tfrac{1}{n}\textstyle\sum_{i=1}^{n} E[T_i] \tag{5.7}$$

$$P\left(\left|\sum_{i=1}^{n} T_i - E\sum_{i=1}^{n} T_i\right| \geq \varepsilon\right) = P\left[\left(\sum_{i=1}^{n} T_i - E\sum_{i=1}^{n} T_i\right)^2 \geq \varepsilon^2\right]$$

$$\leq \frac{\boldsymbol{E(\sum_{i=1}^{n} T_i - E\sum_{i=1}^{n} T_i)^2}}{\boldsymbol{\varepsilon^2}} \qquad \text{by Markov}$$

$$= \frac{Var(\sum_{i=1}^{n} T_i)}{\varepsilon^2} \qquad \text{by definition of variance}$$

$$= \frac{n\sigma^2}{\varepsilon^2} \tag{5.8}$$

since $\sigma^2 = \frac{1}{n}\sum_{i=1}^{n} Var(T_i)$

and by independence $Var(\sum_{i=1}^{n} T_i) = \sum_{i=1}^{n} Var(T_i)$

The absolute difference between mean and sample mean can be bounded by

$$P(|\hat{\mu} - \mu| \geq \varepsilon) = P\left(\left|\frac{1}{n}\sum_{i=1}^{n} T_i - \frac{1}{n}E\sum_{i=1}^{n} T_i\right| \geq \varepsilon\right)$$

$$= P\left(\left|\sum_{i=1}^{n} T_i - E\sum_{i=1}^{n} T_i\right| \geq n\varepsilon\right)$$

$$\leq \frac{n\sigma^2}{n^2\varepsilon^2}$$

$$= \frac{\sigma^2}{n\varepsilon^2} \tag{5.9}$$

Hence, from Equation (5.9), to achieve a desired level of probability, γ, the required sample size is given by

$$n \geq \frac{\sigma^2}{\gamma\varepsilon^2} \tag{5.10}$$

The accuracy ε derived from Equation (5.9) is given by

$$\varepsilon \geq \frac{\sigma}{\sqrt{n\gamma}} \tag{5.11}$$

Then, with probability at least $(1-\gamma)$, the sample mean lies within the interval

$$\mu - \frac{\sigma}{\sqrt{n\gamma}} \leq \hat{\mu} \leq \mu + \frac{\sigma}{\sqrt{n\gamma}} \tag{5.12}$$

In order to bound the right tail of $\sum_{i=1}^{n} T_i$, the application of the Cauchy-Schwarz inequality to the classical Chebyshev inequality will result in Equation (5.13)

$$P(X - EX \geq \varepsilon) \leq \frac{Var(X)}{Var(X)+\varepsilon^2} \tag{5.13}$$

which is due to Cantelli [83].

By letting $X = \sum_{i=1}^{n} T_i$, Equation (5.13) becomes

$$P\left(\sum_{i=1}^{n} T_i - E\sum_{i=1}^{n} T_i \geq \varepsilon\right) \leq \frac{Var\left(\sum_{i=1}^{n} T_i\right)}{Var\left(\sum_{i=1}^{n} T_i\right) + \varepsilon^2}$$

$$= \frac{n\sigma^2}{n\sigma^2+\varepsilon^2} \qquad (5.14)$$

Where (5.14) follows by independence and the definition of σ^2.

From Equation (5.14), the required sample size to achieve the desired level of probability is

$$n \geq \frac{\varepsilon^2\gamma}{\sigma^2(\gamma-1)} \qquad (5.15)$$

The accuracy ε derived from Equation (5.14) is

$$\varepsilon \geq \sigma\sqrt{\frac{n(1-\gamma)}{\gamma}} \qquad (5.16)$$

Equation (5.14) provides an upper bound for the sum of n r.vs, while for $\varepsilon = nt$, the sample mean can be bounded by

$$P(\hat{\mu} - \mu \geq \varepsilon) = P\left(\sum_{i=1}^{n} T_i - E\sum_{i=1}^{n} T_i \geq n\varepsilon\right)$$

$$\leq \frac{n\sigma^2}{n\sigma^2 + n^2\varepsilon^2}$$

$$= \frac{\sigma^2}{\sigma^2+n\varepsilon^2} \qquad (5.17)$$

From Equation (5.17), the required sample size to achieve the desired level of probability γ is

$$n \geq \frac{\sigma^2(1-\gamma)}{\varepsilon^2\gamma} \tag{5.18}$$

The accuracy ε derived from Equation (5.17) is

$$\varepsilon \geq \sigma\sqrt{\frac{(1-\gamma)}{\gamma n}} \tag{5.19}$$

5.3. Bound calculation – Hoeffding's inequality

The goal is to find an upper bound for the probability that the difference between the true and estimated mean is equal to or greater than the desired threshold ε

$$P(\hat{\mu} - \mu \geq \varepsilon) \tag{5.20}$$

For $\varepsilon > 0$, $$P(\hat{\mu} - \mu \geq \varepsilon) = P\left(e^{s(\hat{\mu}-\mu)} \geq e^{s\varepsilon}\right), \qquad s > 0$$

$$\leq \frac{E\left[e^{s(\hat{\mu}-\mu)}\right]}{e^{s\varepsilon}}$$

$$= \frac{E\left[e^{s\frac{1}{n}\sum_{i=1}^{n}(T_i-E[T_i])}\right]}{e^{s\varepsilon}}$$

$$= \frac{\prod_{i=1}^{n} E\left[e^{s\frac{1}{n}(T_i-E[T_i])}\right]}{e^{s\varepsilon}} \tag{5.21}$$

In order to provide a bound for the numerator, Hoeffding's lemma (5.22) will be applied.

Let T be random variable so that $T \in [a, b\,]$ almost surely. Then $\forall\, s > 0$ [82] and from (2.15),

$$E[e^{sT}] \leq e^{\frac{s^2(b-a)^2}{8}} \tag{5.22}$$

The proof lies upon the convexity of the exponential (Jensen's inequality) and Taylor's theorem.

By letting $Z = T_i - E[T_i]$

and considering that $T_i \in [a_i, b_i]$

then $a \leq T_i \leq b$

$$\frac{a-E[T_i]}{n} \leq \frac{T_i-E[T_i]}{n} \leq \frac{b-E[T_i]}{n} \tag{5.23}$$

Let $R_1 = \frac{a}{n} - \frac{ET_i}{n}$

and $R_2 = \frac{b}{n} - \frac{ET_i}{n}$

Substituting Equation (5.23) in Equation (5.21), and recognizing that the argument above applies $\forall i$

$$P(\hat{\mu} - \mu \geq \varepsilon) \leq e^{-s\varepsilon} \prod_{i=1}^{n} e^{s^2(R_2-R_1)^2/8}$$

$$= exp\left\{\frac{s^2(b-a)^2}{8n} - s\varepsilon\right\} \tag{5.24}$$

The bound is minimized for

$$s = \frac{4n\varepsilon}{(b-a)^2} \tag{5.25}$$

and hence:

$$P(\hat{\mu} - \mu \geq \varepsilon) \leq exp\left\{\frac{-2n\varepsilon^2}{(b-a)^2}\right\} \qquad (5.26)$$

The target is to minimize the probability in Equation (5.26).

Allowing the number of samples to approach infinity results in the probability approaching zero since

$$\lim_{n\to\infty} P(\hat{\mu} - \mu \geq \varepsilon) = 0$$

Moreover, to achieve a desired level of probability γ, the number of samples needed may be found form Equation (5.26) thus:

$$n \geq -\frac{(b-a)^2 \ln \gamma}{2\varepsilon^2} \qquad (5.27)$$

While for $\varepsilon > 0$ the accuracy ε for a desired level of confidence γ and sample size *n* is

$$\varepsilon \geq (b-a)\sqrt{-\frac{\ln \gamma}{2n}} \qquad (5.28)$$

Similarly to the upper bound Equation (5.26), by applying the appropriate arguments, the lower bound can be derived:

$$P(\mu - \hat{\mu} \geq \varepsilon) \leq exp\left\{\frac{-2n\varepsilon^2}{(b-a)^2}\right\} \qquad (5.29)$$

From Equations (5.26) and (5.29), the 2-sided inequality can be obtained as shown in Equation (5.30).

$$P(|\hat{\mu} - \mu| \geq \varepsilon) \leq 2exp\left\{\frac{-2n\varepsilon^2}{(b-a)^2}\right\} \qquad (5.30)$$

Hence, for the 2-sided inequality, to achieve a desired level of probability γ, the required sample size can be found from Equation (5.29), thus:

$$n \geq -\frac{(b-a)^2}{2\varepsilon^2}\ln\left(\frac{2}{\gamma}\right) \tag{5.31}$$

The accuracy ε derived from Equation (5.30) is

$$\varepsilon \geq \sqrt{\frac{(b-a)^2}{2n^2}\ln\left(\frac{2}{\gamma}\right)} \tag{5.32}$$

From Equation (5.32) the confidence interval, which is the range $[\mu-\varepsilon,\ \mu+\varepsilon]$ is given by

$$\mu - \sqrt{\frac{(b-a)^2}{2n}\ln\left(\frac{2}{\gamma}\right)} \leq \hat{\mu} \leq \mu + \sqrt{\frac{(b-a)^2}{2n}\ln\left(\frac{2}{\gamma}\right)} \tag{5.33}$$

with probability at least $(1-\gamma)$.

In order to bound the right tail of $\sum_{i=1}^{n} T_i$, let $\varepsilon = \frac{t}{n}$. From Equation (5.26) the following can be derived:

$$P(\sum_{i=1}^{n} T_i - E\sum_{i=1}^{n} T_i \geq t) \leq exp\left\{-\frac{2t^2}{n(b-a)^2}\right\} \tag{5.34}$$

From Equation (5.34) and by letting $t=\varepsilon$ for consistency in notation, to achieve a desired level of probability γ, the required sample size is

$$n \geq \frac{2\varepsilon^2}{2(b-a)\ln(1/\gamma)} \tag{5.35}$$

The accuracy ε derived from Equation (5.34) is

$$\varepsilon \geq \sqrt{\frac{n(b-a)^2\ln\left(\frac{1}{\gamma}\right)}{2}} \tag{5.36}$$

By symmetry, the lower bound follows from Equation (5.29), resulting in the following 2-sided bound for the sum of n r.vs.

$$P(|\sum_{i=1}^{n} T_i - E\sum_{i=1}^{n} T_i| \geq \varepsilon) \leq 2exp\left\{-\frac{2\varepsilon^2}{n(b-a)^2}\right\} \tag{5.37}$$

From Equation (5.37), the desired level of probability γ is achieved for sample size given by

$$n \geq \frac{2\varepsilon^2}{(b-a)^2 \ln(2/\gamma)} \tag{5.38}$$

The accuracy ε derived from Equation (5.37) is

$$\varepsilon \geq \sqrt{\frac{n(b-a)^2 \ln\left(\frac{2}{\gamma}\right)}{2}} \tag{5.39}$$

From Equation (5.39), the confidence interval for $\sum_{i=1}^{n} T_i - E\sum_{i=1}^{n} T_i$ is given by

$$E\sum_{i=1}^{n} T_i - \sqrt{\frac{n(b-a)^2 \ln\left(\frac{2}{\gamma}\right)}{2}} \leq \sum_{i=1}^{n} T_i \leq E\sum_{i=1}^{n} T_i + \sqrt{\frac{n(b-a)^2 \ln\left(\frac{2}{\gamma}\right)}{2}}$$

or

$$n\mu - \sqrt{\frac{n(b-a)^2 \ln\left(\frac{2}{\gamma}\right)}{2}} \leq \sum_{i=1}^{n} T_i \leq n\mu + \sqrt{\frac{n(b-a)^2 \ln\left(\frac{2}{\gamma}\right)}{2}} \tag{5.40}$$

5.4. Bound calculation – Bennett’s inequality

Let $\mathrm{T}_1, \dots, \mathrm{T}_\mathrm{n}$ be i.i.d. r.v., such that $T_i \in [a, b]$, $\forall\, i \in \{1, \dots, n\}$ and $0 < a < b < \infty$.

Also let $\sigma^2 = \frac{1}{n}\sum_{i=1}^{n} {\sigma_i}^2$, where ${\sigma_i}^2 = Var(T_i)$.

Without loss of generality, r.v. T_i will be centered by $a \leq T_i \leq b$. Then,

$$a - ET_i \leq T_i - ET_i \leq b - ET_i \tag{5.41}$$

Note, that since $0 < a < b$ implies that $b - ET_i > a - ET_i$, the centred r.v.

$Z_i = T_i - ET_i$ can be symmetrized by letting $C = b - ET_i$, resulting in $|Z_i| < C$ and $E(Z_i) = 0$ as required.

Similarly to Hoeffding's inequality, the moment generating function needs to be bounded, but in this case using knowledge of the variance. Therefore, a tighter bound than Hoeffding's will be derived.

For $E > 0$, $s > 0$

$$P\left(\sum_{i=1}^{n} Z_i \geq \varepsilon\right) = P\left(e^{s\sum_{i=1}^{n} Z_i} \geq e^{s\varepsilon}\right)$$

$$\leq \frac{E\left[e^{s\sum_{i=1}^{n} Z_i}\right]}{e^{s\varepsilon}} \qquad \text{(from Markov)}$$

$$= \frac{\prod_{i=1}^{n} E\left(e^{sZ_i}\right)}{e^{s\varepsilon}} \qquad \text{(by independence) (5.42)}$$

From Taylor series for e^{sZ_i} and taking the expectation

$$Ee^{sZ_i} = 1 + sEZ_i + \sum_{r=2}^{\infty} \frac{s^r E({Z_i}^r)}{r!}$$

and since $EZ_i = 0$

$$Ee^{sZ_i} = 1 + \sum_{r=2}^{\infty} \frac{s^r E({Z_i}^r)}{r!}$$

$$= 1 + s^2 {\sigma_i}^2 \sum_{r=2}^{\infty} \frac{s^{r-2} E({Z_i}^r)}{r!\, {\sigma_i}^2}$$

$$\leq exp\{F_i s^2 {\sigma_i}^2\}$$

where $F_i = \sum_{r=2}^{\infty} \frac{s^{r-2} E({Z_i}^r)}{r! {\sigma_i}^2}$

Using Schwarz's inequality and since the expectation of a function is the Lebesgue integral [84] with respect to the probability measure P,

$$E{Z_i}^r = \int_P {Z_i}^{r-1} Z_i \; dP$$

$$\leq \left(\int_P \left|{Z_i}^{r-1}\right|^2 \; dP \right)^{1/2} \left(\int_P \; |Z_i|^2 \; dP \right)^{1/2}$$

$$= \sigma_i \left(\int_P \; \left|{Z_i}^{r-1}\right|^2 \right)^{1/2}$$

Applying Schwarz's inequality recursively n times,

$$EZ_i{}^r \le \sigma_i{}^{1+\frac{1}{2}+\cdots+\frac{1}{2^{n-1}}}\left(\int_P \left|Z_i{}^{2^n r-2^{n+1}-1}\right| dP\right)^{\frac{1}{2^n}}$$

$$= \sigma_i{}^{2\left(1-\frac{1}{2^n}\right)}\left(\int_P \left|Z_i{}^{(2^n r-2^{n+1}-1)}\right| dP\right)^{\frac{1}{2^n}} \qquad (5.43)$$

Since $|Z_i| \le C$,

$$\left(\int_P \left|Z_i{}^{2^n r-2^{n+1}-1}\right| dP\right)^{\frac{1}{2^n}} \le \left(C^{2^n r-2^{n+1}-1}\right)^{\frac{1}{2^n}} \qquad (5.44)$$

From equations (5.43) and (5.44) the r^{th} moment is bounded by

$$EZ_i{}^r \le \sigma_i{}^{2\left(1-\frac{1}{2^n}\right)} C^{\left(r-2-\frac{1}{2^n}\right)} \qquad (5.45)$$

And since $\lim_{n\to\infty} \sigma_i{}^{2\left(1-\frac{1}{2^n}\right)} C^{\left(r-2-\frac{1}{2^n}\right)} = \sigma_i{}^2 C^{r-2}$,

$$EZ_i{}^r \le \sigma_i{}^2 C^{r-2} \qquad (5.46)$$

Applying Equation (5.46) to F_i,

$$F_i = \sum_{r=2}^{\infty} \frac{s^{r-2} E(Z_i{}^r)}{r!\,\sigma_i{}^2}$$

$$\le \sum_{r=2}^{\infty} \frac{s^{r-2} \sigma_i{}^2 C^{r-2}}{r!\,\sigma_i{}^2}$$

$$= \frac{1}{s^2 C^2} \sum_{r=2}^{\infty} \frac{s^r C^r}{r!}$$

$$= \frac{1}{s^2 C^2}\left(e^{sC} - 1 - sC\right) \qquad \text{(by Taylor's theorem) (5.47)}$$

From Equation (5.47),

$$Ee^{sZ_i} \leq e^{F_i s^2 \sigma_i{}^2}$$

$$= exp\left\{s^2 \sigma_i{}^2 \frac{(e^{sC} - 1 - sC)}{s^2 C^2}\right\} \tag{5.48}$$

Combining Equations (5.42) and (5.48) and because $\sigma^2 = \frac{\sum_{i=1}^{n} \sigma_i{}^2}{n}$,

$$P(\textstyle\sum_{i=1}^{n} Z_i \geq \varepsilon) \leq exp\left\{s^2 n\sigma^2 \frac{(e^{sC} - 1 - sC)}{s^2 C^2} - s\varepsilon\right\} \tag{5.49}$$

The right hand side of Equation (5.49) is minimised for s as shown in Equation (5.50)

$$s = \frac{1}{C} \ln\left(\frac{\varepsilon C}{n\sigma^2} + 1\right) \tag{5.50}$$

Substituting Equation (5.50) in (5.49),

$$P(\textstyle\sum_{i=1}^{n} Z_i \geq \varepsilon) \leq exp\left\{\frac{n\sigma^2}{C^2}\left[\frac{\varepsilon C}{n\sigma^2} - \ln\left(\frac{\varepsilon C}{n\sigma^2} + 1\right) - \frac{\varepsilon C}{n\sigma^2} \ln\left(\frac{\varepsilon C}{n\sigma^2} + 1\right)\right]\right\} \tag{5.51}$$

Let $H(x) = (1 + x)\ln(1 + x) - x$, to derive Bennett's inequality:

$$P(\textstyle\sum_{i=1}^{n} Z_i \geq \varepsilon) \leq exp\left\{-\frac{n\sigma^2}{C^2} H\left(\frac{\varepsilon C}{n\sigma^2}\right)\right\} \tag{5.52}$$

Equation (5.52) bounds the sum of n r.v.s, while for $\varepsilon = nt$, the sample mean can be bounded by

$$P\left(\frac{1}{n}\textstyle\sum_{i=1}^{n} Z_i \geq t\right) \leq exp\left\{-\frac{n\sigma^2}{C^2} H\left(\frac{tC}{\sigma^2}\right)\right\} \tag{5.53}$$

5.5. Bound calculation – Bernstein's inequality

By applying the elementary inequality

$$H(x) \geq G(x) = \frac{3}{2}\frac{x^2}{x+3}, \forall x \geq 0 \tag{5.54}$$

to Equation (5.52) to bound Bennett's inequality further,

$$P\left(\sum_{i=1}^{n} Z_i \geq \varepsilon\right) \leq exp\left\{-\frac{n\sigma^2}{C^2}\, G\left(\frac{\varepsilon C}{n\sigma^2}\right)\right\}$$

$$= exp\left\{-\frac{\varepsilon^2}{2\left(n\sigma^2 + \frac{\varepsilon C}{3}\right)}\right\} \tag{5.55}$$

From Equation (5.55), to achieve the desired level of probability γ, the required sample size is

$$n \geq -\frac{\varepsilon}{\sigma^2}\left(\frac{\varepsilon}{2\ln\gamma} + \frac{C}{3}\right) \tag{5.56}$$

By symmetry, and by applying Equation (5.55) to $-Z_i$, the following 2-sided bound for the sum of n r.v.s can be derived, as shown in the following

$$P(|\textstyle\sum_{i=1}^{n} Z\, i| \geq \varepsilon) \leq 2exp\left\{-\frac{\varepsilon^2}{2\left(n\sigma^2 + \frac{\varepsilon C}{3}\right)}\right\} \tag{5.57}$$

Equation (5.55) bounds the sum of n r.v.s, while for $\varepsilon = nt$ the sample mean can be bounded by

$$P\left(\frac{1}{n}\textstyle\sum_{i=1}^{n} Z_i \geq t\right) \leq\ exp\left\{-\frac{nt^2}{2\left(\sigma^2 + \frac{Ct}{3}\right)}\right\} \tag{5.58}$$

By symmetry, applying Equation (5.58) to $-Z_i$ and recalling that $Z_i = T_i - \mu$, the following 2-sided bound due to Bernstein can be derived, as shown in Equation (5.59)

$$P\left(\left|\frac{1}{n}\sum_{i=1}^{n} T_i - \mu\right| \geq \varepsilon\right) \leq 2exp\left\{-\frac{n\varepsilon^2}{2\left(\sigma^2+\frac{\varepsilon C}{3}\right)}\right\} \tag{5.59}$$

From Equation (5.58), to achieve the desired level of probability γ, the required sample size is

$$n \geq -2\left(\sigma^2 + \frac{C\varepsilon}{3}\right)\frac{\ln(\gamma)}{\varepsilon^2} \tag{5.60}$$

To calculate the accuracy ε from Equation (5.58),

let
$$y = -\frac{\varepsilon^2}{2\left(n\sigma^2+\frac{\varepsilon C}{3}\right)} \tag{5.61}$$

Then Equation (5.58) is of the form

$$P\left(\sum_{i=1}^{n} Z_i \geq \varepsilon\right) \leq e^{-y}$$

Solving Equation (5.61) for ε, results in

$$\varepsilon = \frac{yC}{3} + \sqrt{\frac{y^2C^2}{9} + 2n\sigma^2 y} \tag{5.62}$$

Note, that $\varepsilon = \frac{yC}{3} - \sqrt{\frac{y^2C^2}{9} + 2n\sigma^2 y}$ is rejected, since $\varepsilon \geq 0$ and

$\frac{yC}{3} < \sqrt{\frac{y^2C^2}{9} + 2n\sigma^2 y}$

Using the inequality $\sqrt{a+b} \leq \sqrt{a} + \sqrt{b}$ in Equation (5.62),

$$\sum_{i=1}^{n} Z_i \geq \frac{yC}{3} + \frac{yC}{3} + \sqrt{2n\sigma^2 y} = \frac{2yC}{3} + \sqrt{2n\sigma^2 y}$$

Normalizing by n,

$$\frac{1}{n}\sum_{i=1}^{n} Z_i \geq \frac{2yC}{3n} + \sqrt{\frac{2\sigma^2 y}{n}}$$

Finally, since

$$P\left(\frac{1}{n}\sum_{i=1}^{n} Z_i \geq \frac{2yC}{3n} + \sqrt{\frac{2\sigma^2 y}{n}}\right) \leq e^{-y} \leq \gamma$$

or

$$P\left(\frac{1}{n}\sum_{i=1}^{n} Z_i \leq \frac{2yC}{3n} + \sqrt{\frac{2\sigma^2 y}{n}}\right) \geq 1 - \gamma$$

the confidence interval of $\frac{1}{n}\sum_{i=1}^{n} T_i - \mu$ is given by

$$-\left(\frac{2yC}{3n} + \sqrt{\frac{2\sigma^2 y}{n}}\right) \leq \frac{1}{n}\sum_{i=1}^{n} T_i - \mu \leq \frac{2yC}{3n} + \sqrt{\frac{2\sigma^2 y}{n}}$$

with probability at least $(1-\gamma)$.

5.6. Results

Testing of the bounding approach was conducted based on the reliability concept introduced previously, using the initial dataset that resulted from the simulation in Chapter 4 and generated based on the variation of the four attributes – data size, key size, padding scheme and mode of operation.

For each collection of attributes (i.e. for each case), bootstrap sampling (i.e. sampling with replacement from the ECDF) was used to generate 10,000 samples, each of size n. Those samples generated from the reliability function, were utilized to derive the ECDF of events

$$A = \left\{\frac{1}{n}\sum_{i=1}^{n} Z_i = \hat{\mu} - \mu \geq \varepsilon\right\},$$

$$B = \{\sum_{i=1}^{n} Z_i = \sum_{i=1}^{n} T_i - n\mu \geq \varepsilon\},$$

$$C = \left\{\left|\frac{1}{n}\sum_{i=1}^{n} Z_i\right| = |\hat{\mu} - \mu| \geq \varepsilon\right\},$$

$$D = \{|\sum_{i=1}^{n} Z_i| = |\sum_{i=1}^{n} T_i - n\mu| \geq \varepsilon\} \tag{5.63}$$

In terms of the analysis of the bounds, the right tail of distributions of the events A, B, C, D (namely $\bar{F}_A, \bar{F}_B, \bar{F}_C, \bar{F}_D$) were investigated and a pairwise comparison with the derived bounds has been made. The sample size n was subsequently varied to determine the impact of the sample size on the effectiveness and precision of the bound.

The following subsections reveal the information gained about the closeness of bounding the probability of the expected mean to deviate at least ε from its theoretical mean. In addition, the optimal sample size n has been calculated in order to achieve the desired probability of completion for n encryptions.

5.6.1. Bounding the overall time – Upper Bound

5.6.1.1. Chebyshev's – Cantelli's bounds

In this subsection, $\bar{F}_B$ is analyzed and compared with the corresponding bound in Equation (5.49). The case of interest will be Case 1, using the following encryption parameters. Algorithm: *AES*, Data size: *16 bytes*, Key size: *128 bits*, Mode of operation: *CBC*, Padding scheme: *No Padding*.

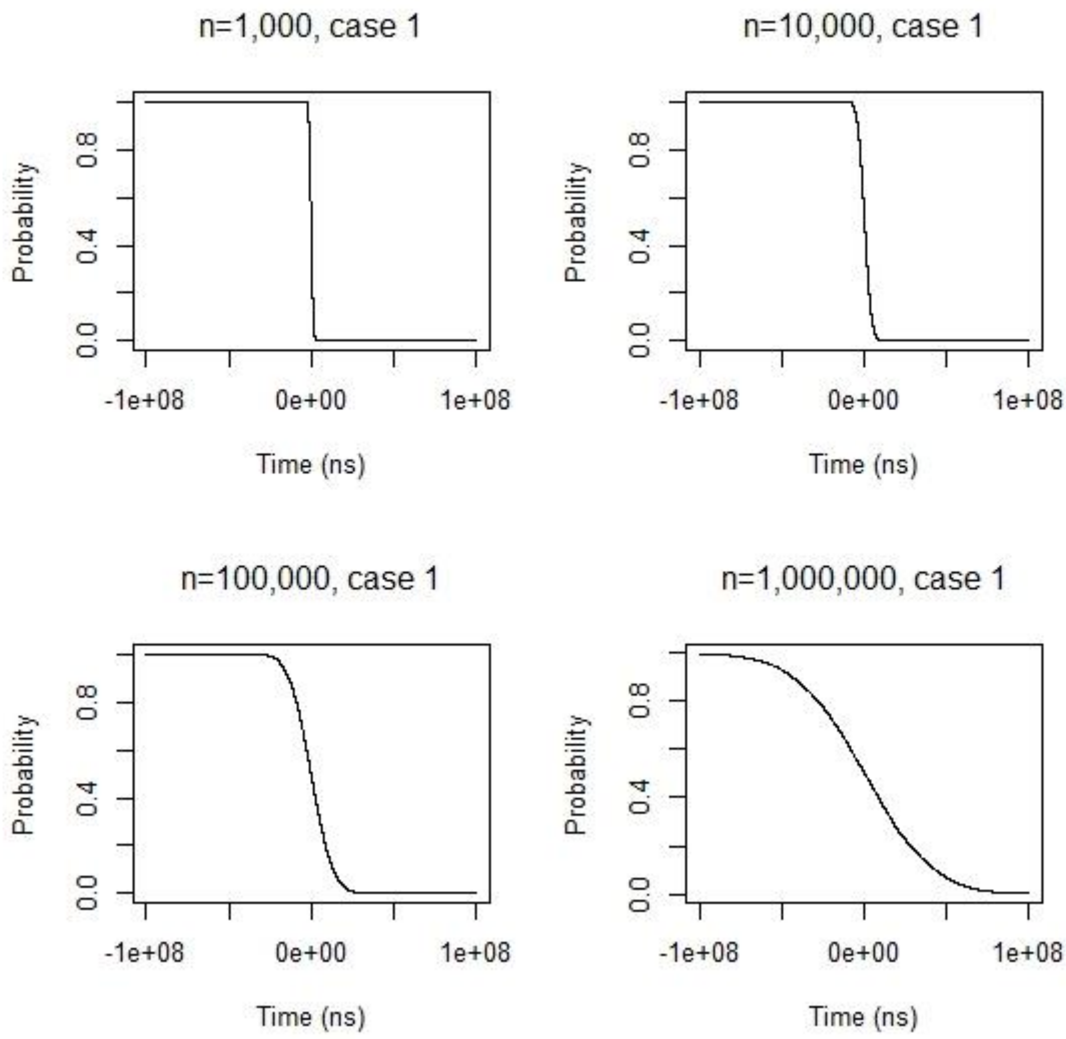

Figure 5.1: $\bar{F}_B$ for n variation – Case 1.

Under this encryption scheme, n encryptions will be generated from Case 1 ECDF to result the overall execution time. The difference between the sum of n encryptions and n times the mean encryption time of a single encryption is recorded. This process will be repeated $k = 10{,}000$ times, to derive the empirical cumulative distribution $\bar{F}_B$of the event B shown in Equation (5.63) and is plotted in Figure 5.1. The figure depicts the outcome of this process, as resulted from those k statistics, for $n = 1{,}000$, $n = 10{,}000$, $n = 100{,}000$ and $n = 1{,}000{,}000$. As illustrated in Figure 5.1, for all variations of n, the mean difference is zero and the ECDF is symmetric, as expected from CLT for large n. As shown in Figure 5.1, due to symmetry, for $\sum_{i=1}^{n} T_i - n\mu \geq \varepsilon$, ε appears equally likely to have a positive or negative value; however, of interest is the investigation of $\sum_{i=1}^{n} T_i > n\mu + \varepsilon$, i.e. the right tail of $\bar{F}_B$, since this means that the overall encryption time of those n encryptions will exceed the expected encryption time by ε.

Figure 5.2 depicts the right tail of $\bar{F}_B$, i.e. the positive values that will be bounded. The ECDF symmetry discussed earlier is also apparent in Figure 5.2, as for $\varepsilon > 0$, the highest probability of exceeding ε is 0.5 and decays as ε increases. As mentioned earlier, of interest is the bound on the probability that the overall encryption time of n encryptions will exceed the expected encryption time by ε; therefore, bounding the right tail is the main objective of this investigation, and the decay of the probability of exceeding ε, as ε increases, will be further examined.

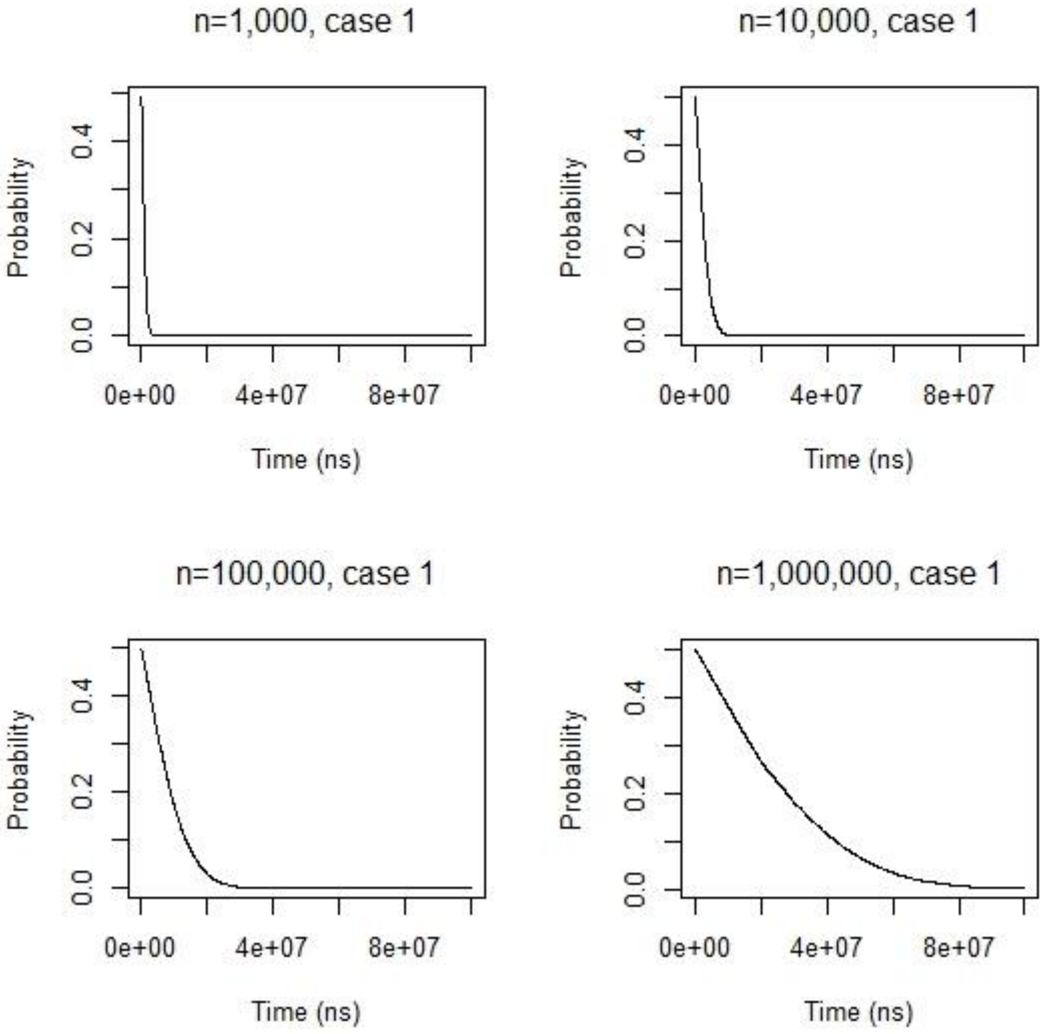

Figure 5.2: Right tail of $\bar{F}_B$ for n variation – Case 1.

By applying the appropriate values of n and σ^2 in Equation (5.14) and for different values of ε, ranging from 0 to 8e+07 with a time step of 1000 ns, a point-wise comparison between the right tail of $\bar{F}_B$ and Chebychev's inequality is presented in Figure 5.3. The point-wise comparison allows for displaying the two lines and their alignment, arranged for visual inspection. Therefore, for a specific ε value in the time axis, the probabilities of the right tail of $\bar{F}_B$ and Chebychev's inequality can be compared. As shown in Figure 5.3, Chebyshev's inequality does not provide useful information, as the corresponding value remains close to 1 even for $n = 1{,}000{,}000$ and large ε.

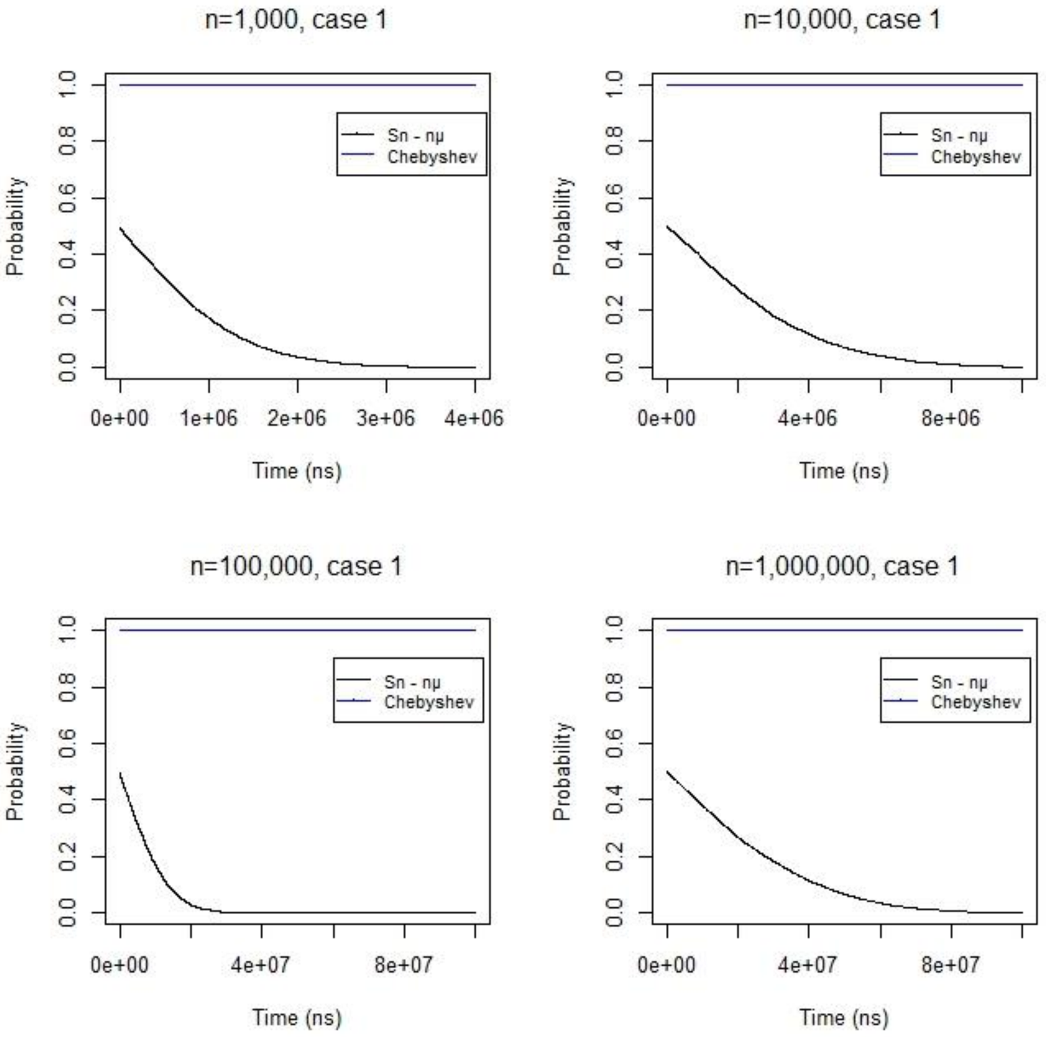

Figure 5.3: Right tail of $\bar{F}_B$ and Chebyshev's bound – Case 1.

This is illustrated in the bottom right graph in Figure 5.3, where although the probability that the n encryptions will not exceed ε=8e+07 is close to zero, the corresponding value of Chebychev's inequality for 8e+07 *ns* is close to 1.

5.6.1.2. Hoeffding's bound

Following similar methodology, and in the need to derive an inequality that decays exponentially following the shape of the right tail of $\bar{F}_B$, Hoeffding's inequality (5.34) will be investigated.

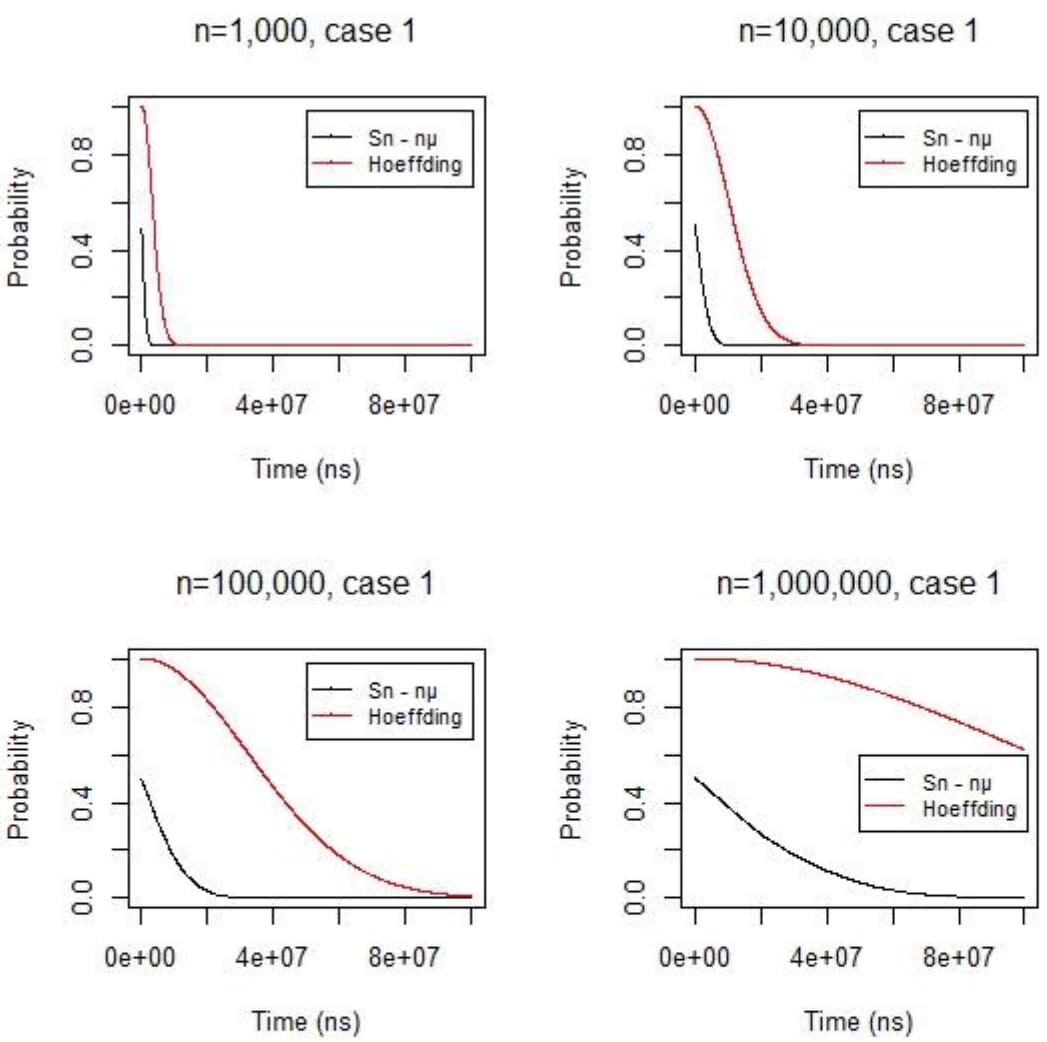

Figure 5.4: Right tail of $\bar{F}_B$ and Hoeffding's bound – Case 1.

From Figure 5.4, the tightening of the bound with increasing sample size may be clearly seen. The curve of Hoeffding's bound decays exponentially, which is an improvement compared to Chebyshev's. Hoeffding's somehow mimics the curve of $\bar{F}_B$, but that tightness comes to a big cost of sample size.

In other words, when it is necessary to provide a tight bound on the tail probability, there is an inherent need to employ an adequate sample size. In the specific case of encryption, this means that the degree of tightness depends highly on the number of encryptions that have to be considered.

This thus indicates that Hoeffding's bound although very useful for bounding the tail probabilities, will only deliver an effective and adequately tight bound when the minimum size required in Equation (5.35) is adhered to.

In this specific example of security mode 1, for a confidence $\gamma = 0.05$ and $\varepsilon = 8 \times 10^7$, the minimum sample size n from Equation (5.35) is $102{,}454$. The scale of this sample size, although large, may be a representative example of a viral application, such as Facebook, Google, or Amazon, where the sample may be the number of encryptions executed during a time window of several minutes.

It would also be of interest to ascertain the impact of the sample size on the probability γ for moderate sample size. In other words, it is attempted to upper bound the right tail of $\bar{F}_B$ which is $P(event\ B) \leq \gamma$, where B comes from Equation (5.63), for a sample size of order $n = 1{,}000$ and $n = 10{,}000$. As shown in Figure 5.4, Hoeffding's inequality is not tight in that scale. Hopefully, by considering the variance, Bernstein's and Bennett's inequalities can be tighter than Hoeffding's, as presented in the next subsection.

5.6.1.3. Bernstein's and Bennett's bounds

Although an exponential bound has already been derived in the previous subsection, an attempt to make the bound even tighter will be made using equations (5.55) and (5.52).

Figure 5.5 depicts the tightening of the bound with increasing sample size. The curve of Bernstein's bound decays exponentially with a faster rate than Hoeffding's bound.

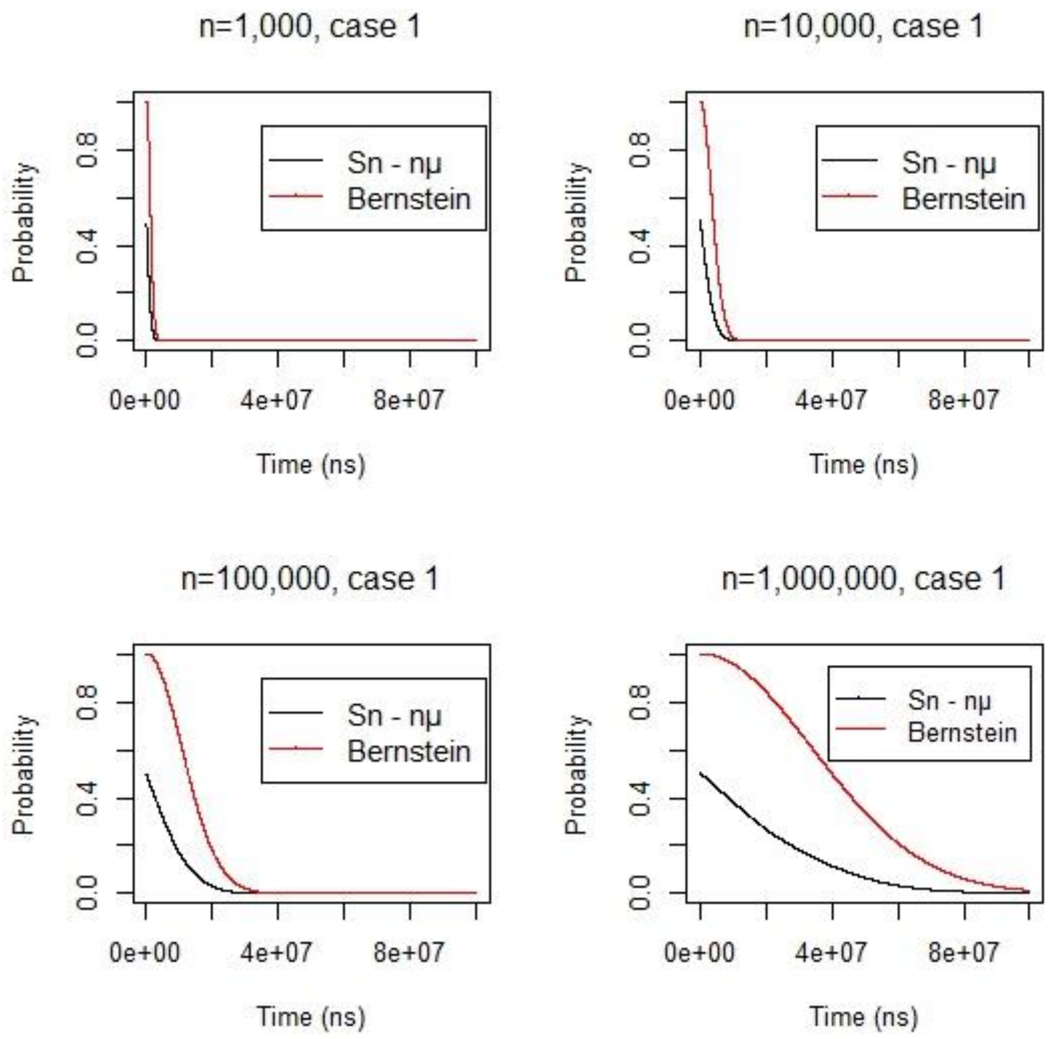

Figure 5.5: Right tail of $\bar{F}_B$ and Bernstein's bound – Case 1.

By comparing Figure 5.4 and Figure 5.5 it is clear that for the same sample size n, the probability computed by Equation (5.55) is more accurate than the one computed by Equation (5.34). It has to be noted that Bernstein's curve approaches $\bar{F}_B$ even for small sample size $n = 1{,}000$.

In this specific example of security mode 1, for a confidence $\gamma = 0.05$ and $\varepsilon = 8 \times 10^6$, the minimum sample size n from Equation (5.56) is $8{,}850$. It is notable how much tighter this result is compared to Hoeffding's, since from Equation (5.34) the resulting value is 0.7. This is also shown in Figure 5.4 where $n = 10{,}000$.

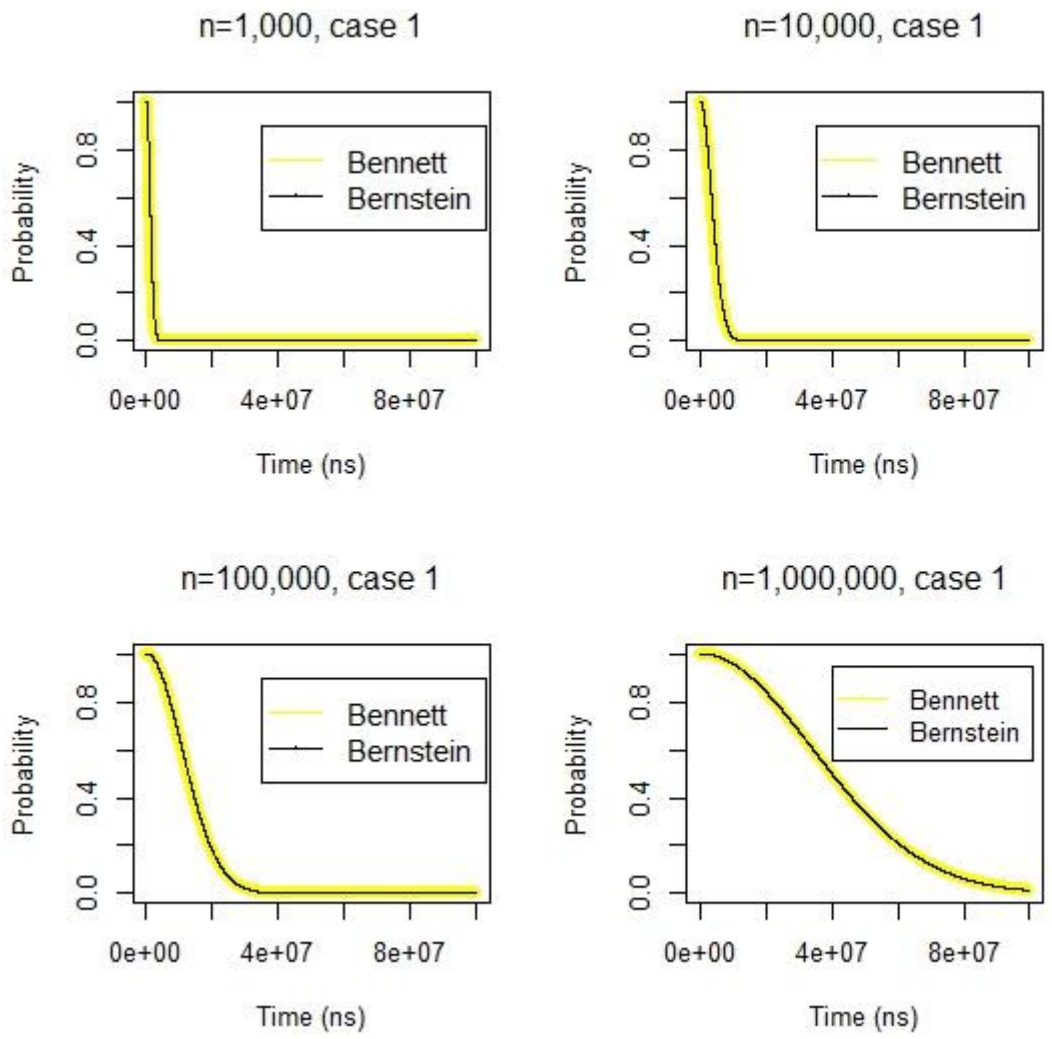

Figure 5.6: Bernstein’s and Bennett’s bound – Case 1.

By plotting equations (5.52) and (5.55) for various sample sizes, it is shown that indeed Bennett’s and Bernstein’s inequalities are roughly the same. This was expected, since the approach followed to derive Equation (5.55) by further bounding Equation (5.52) has been made in purpose, to emphasize the similarities of those two inequalities. As shown in Figure 5.6, where the thick lines represent Bennett’s bound, the two curves are roughly the same.

Table 5.1 presents a point-wise comparison of the two bounds for variations of n and ε, where even for four decimal points, the two inequalities converge numerically.

Table 5.1: Bennett's and Bernstein's values for various n and ε.

n	$\varepsilon\ (ns)$	Bennett's bound	Bernstein's bound
$1{,}000$	2×10^6	0.204	0.206
$1{,}000$	4×10^6	0.0028	0.0030
$10{,}000$	4×10^6	0.505	0.505
$10{,}000$	8×10^6	0.688	0.690
$100{,}000$	4×10^7	0.001	0.001
$100{,}000$	8×10^7	2.4×10^{-12}	2.45×10^{-12}
$1{,}000{,}000$	4×10^7	0.4969	0.4964
$1{,}000{,}000$	8×10^7	0.0626	0.0626

5.6.1.4. Upper bound comparison for sum

In this section, a comparison of the efficiency and tightness between the analyzed bounds will be presented, for several sample sizes. A visualization of the point-wise comparison of $\bar{F}_B$ and the bounds is presented in Figure 5.7.

As discussed in previous subsections, Hoeffding's, Bennett's and Bernstein's bound decay exponentially, with Bennett's and Bernstein's revealing a remarkable faster rate of convergence to the right tail $\bar{F}_B$, even for relatively small sample size.

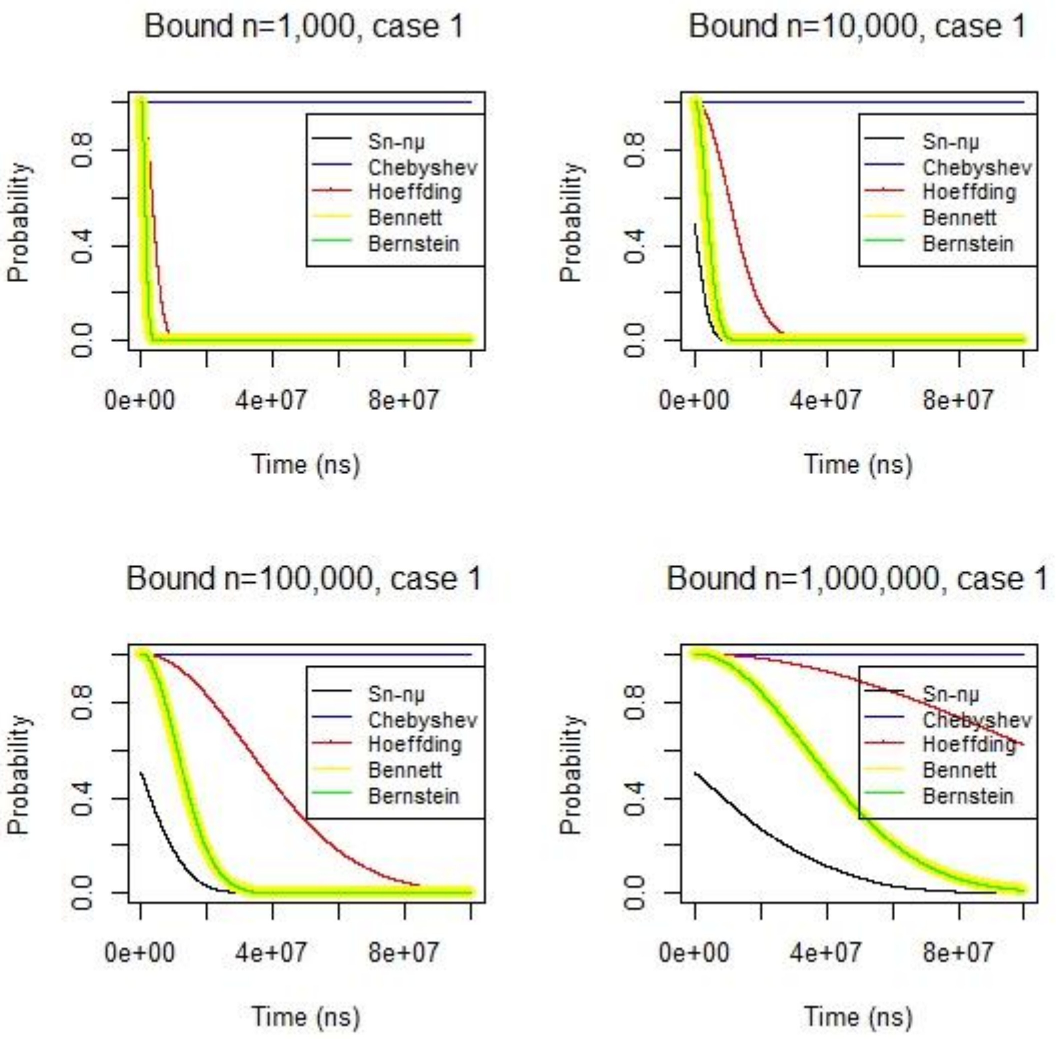

Figure 5.7: Right tail of $\bar{F}_B$ vs Bounds – Case 1.

For all variations of parameters n and ε the following relation holds:

$$\bar{F}_B = P(S_n - n\mu \geq \varepsilon) \leq P_{Bennett} \approx P_{Bernstein} < P_{Hoeffding} \ll P_{Chebyshev} \quad (5.64)$$

where $S_n = \sum_{i=1}^{n} T_i$, $\mu = E(T_i)$ and $T_i \sim ECDF_{Case\ 1}$ is the r.v. representing the execution time of the i^{th} encryption.

Table 5.2 presents a point-wise comparison of the relation/ranking given by (5.64), for ε being of order α times μ. Therefore, several values of ε have been used as an input to illustrate the differences among bounds.

Table 5.2: $P(S_n - n\mu \geq \varepsilon)$ investigation using n and a variation.

$\boldsymbol{n}$	$\boldsymbol{\alpha}$	$\boldsymbol{\varepsilon = \alpha\mu\ (ns)}$	$\boldsymbol{\bar{F}_B}$	$\boldsymbol{P_{Bennett}}$	$\boldsymbol{P_{Bernstein}}$	$\boldsymbol{P_{Hoeffding}}$	$\boldsymbol{P_{Chebyshev}}$
1,000	20	2,610,166	0.01	0.071	0.073	0.72	0.9
1,000	30	3,915,248	0.0001	0.0035	0.0038	0.47	0.9
10,000	60	7,830,497	0.011	0.076	0.077	0.74	0.9
10,000	100	13,050,828	0	0.0009	0.0009	0.44	0.9
100,000	200	26,101,656	0.009	0.053	0.053	0.72	0.9
100,000	300	39,152,484	0.0001	0.0014	0.0014	0.47	0.9
1,000,000	700	91,355,796	0.0021	0.02	0.02	0.6	0.9
1,000,000	1,000	130,508,280	0	0.0006	0.0006	0.44	0.9

As shown in this table, Bennett's and Bernstein's inequalities provide a tight bound for $\bar{F}_B$ at the right even for $n = 1{,}000$. The probability of the required time of 1,000 encryptions will not deviate more than $\varepsilon = 2 \times 10^6$ ns from 1000 times (Table 5.2) the average encryption time of a single encryption as suggested by the generated ECDF $\bar{F}_B$. For the same parameterization, Bennett's and Bernstein's result that this probability will not exceed 0.07.

5.6.2. Bounding the overall time – Two-sided bound

In this subsection, an analysis will be made on the behavior of event D in Equation (5.63). The probability of the absolute deviation of the sum on n encryption times from n times the average execution time of a single encryption will be bounded using the two-sided versions of Chebyshev's, Hoeffding's and Bernstein's inequalities.

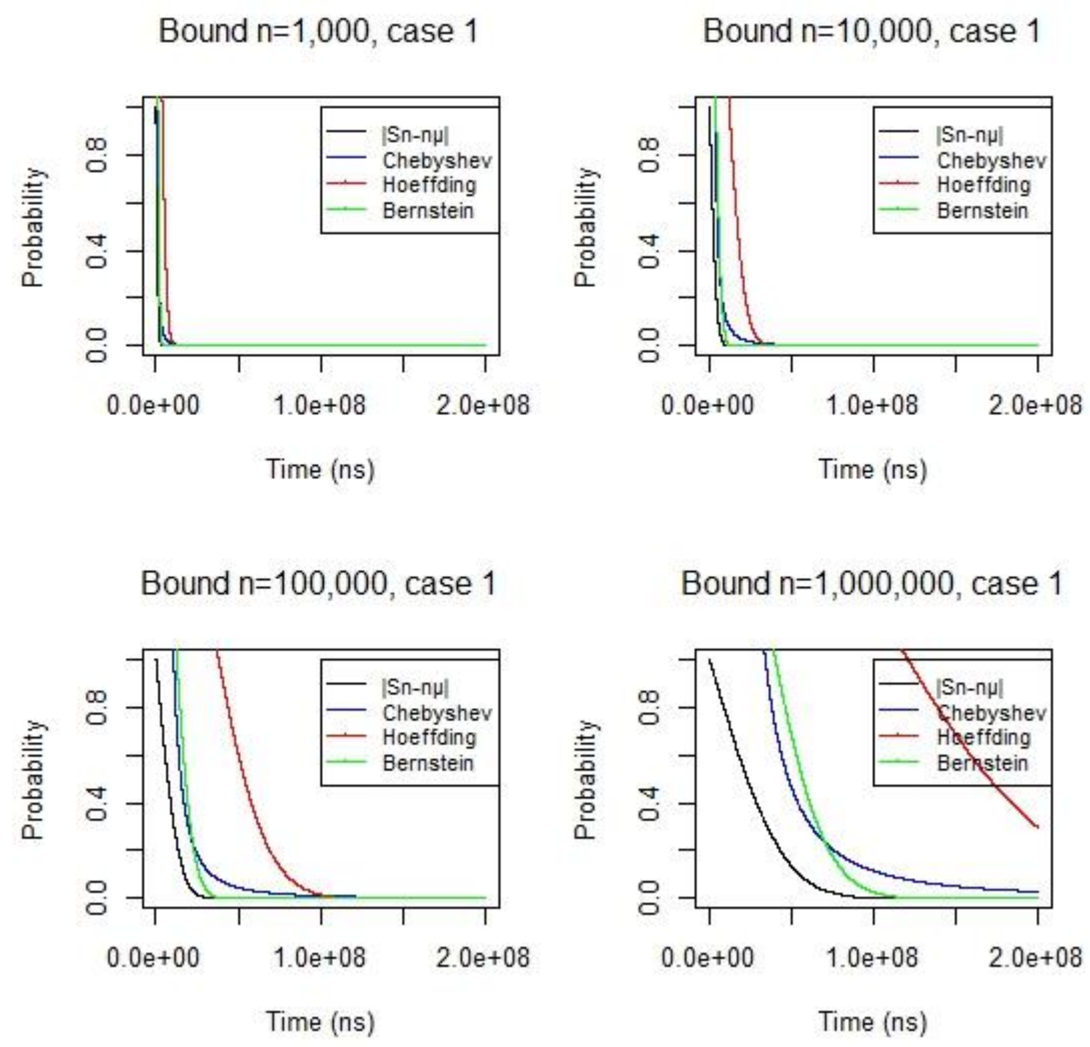

Figure 5.8: Right tail $\bar{F}_D$ and bounds – Case 1.

Figure 5.8 illustrates the right tail $\bar{F}_D$ bounded by the inequalities shown in equations (5.8), (5.37) and (5.57). For all variations of the sample size n, these inequalities reveal an exponential decay, which is desired, with Bernstein's and Chebyshev's demonstrating the ability to mimic the rate of $\bar{F}_D$.

Bernstein's curve provides the tightest bound, whilst Hoeffding's is relatively loose. Considering that the right tail of $\bar{F}_D$ is examined, i.e. ε is large enough, the following relation holds.

$$\bar{F}_D = P(|S_n - n\mu| \geq \varepsilon) \leq P_{Bernstein} \leq P_{Chebychev} \ll P_{Hoeffding} \tag{5.65}$$

Table 5.3: $\bar{F}_D$, Bernstein's, Chebyshev's and Hoeffding's values for various n and ε.

n	$\varepsilon = \alpha\mu\ (ns)$	$\bar{F}_D$	$P_{Bernstein}$	$P_{Hoeffding}$	$P_{Chebyshev}$
1,000	3×10^6	0.005	0.06	0.1	1
1,000	4×10^6	0.0001	0.006	0.07	0.9
10,000	1×10^7	0.002	0.03	0.1	1
10,000	2×10^7	0	2×10^{-7}	0.02	0.2
100,000	3×10^7	0.005	0.04	0.1	1
100,000	4×10^7	0.0004	0.002	0.07	0.9
1,000,000	1×10^8	0.003	0.02	0.1	1
1,000,000	2×10^8	0	6×10^{-8}	0.02	0.2

Table 5.3 presents a point-wise comparison of the right tail of $\bar{F}_D$ and the three bounds. Clearly, Bernstein's appears tight to the tail of $\bar{F}_D$.

5.6.3. Bounding the mean time – Upper Bound

5.6.3.1. Chebyshev's – Cantelli's bounds

This subsection presents the analysis of $\bar{F}_A$, as well as the comparison with the corresponding bound in Equation (5.17). In this section, Case 1 will be used to demonstrate and compare the bounds on the mean encryption time. Therefore, the analysis refers to Case 1 of the security scenario. Under this encryption scheme, n encryptions will be generated from Case 1 ECDF to result the mean execution time of those n encryptions.

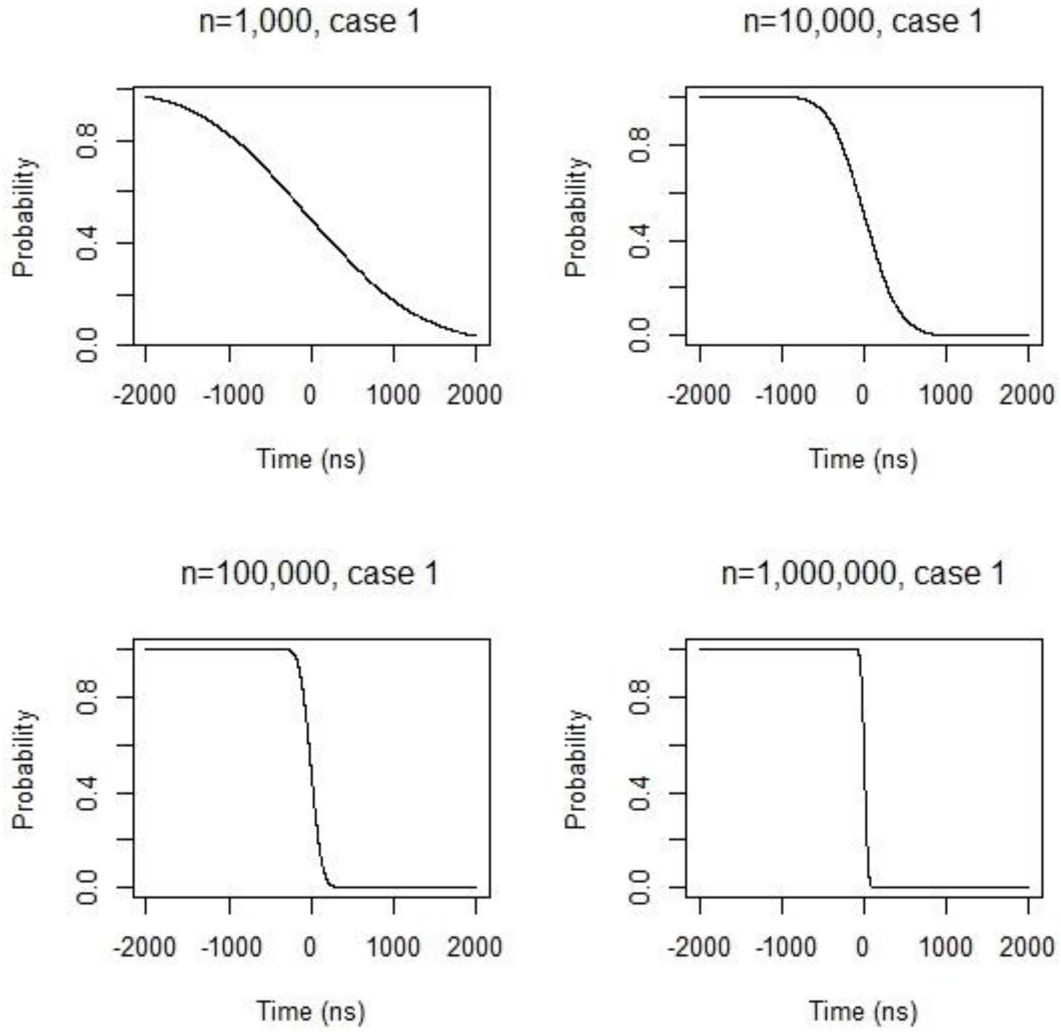

Figure 5.9: $\bar{F}_A$ for n variation – Case 1.

The difference between the mean of n encryptions and the mean encryption time of a single encryption is recorded. This process will be repeated $k = 10{,}000$ times, to derive the empirical cumulative distribution $\bar{F}_A$of the event A shown in Equation (5.63) and is plotted in Figure 5.9. The figure depicts the outcome of this process, as resulted from those k statistics, for $n = 1{,}000$, $n = 10{,}000$, $n = 100{,}000$ and $n = 1{,}000{,}000$. As illustrated in Figure 5.9, for all variations of n, the mean difference is zero and the ECDF is symmetric, as expected from CLT for large n. As shown in Figure 5.9, due to symmetry, for $\hat{\mu} - \mu \geq \varepsilon$, ε appears equally likely to have a positive or negative value; however, of interest is the investigation of $\hat{\mu} \geq \mu + \varepsilon$, i.e. the right tail of $\bar{F}_A$, since this means that the mean encryption time of those n encryptions will exceed the expected encryption time by ε.

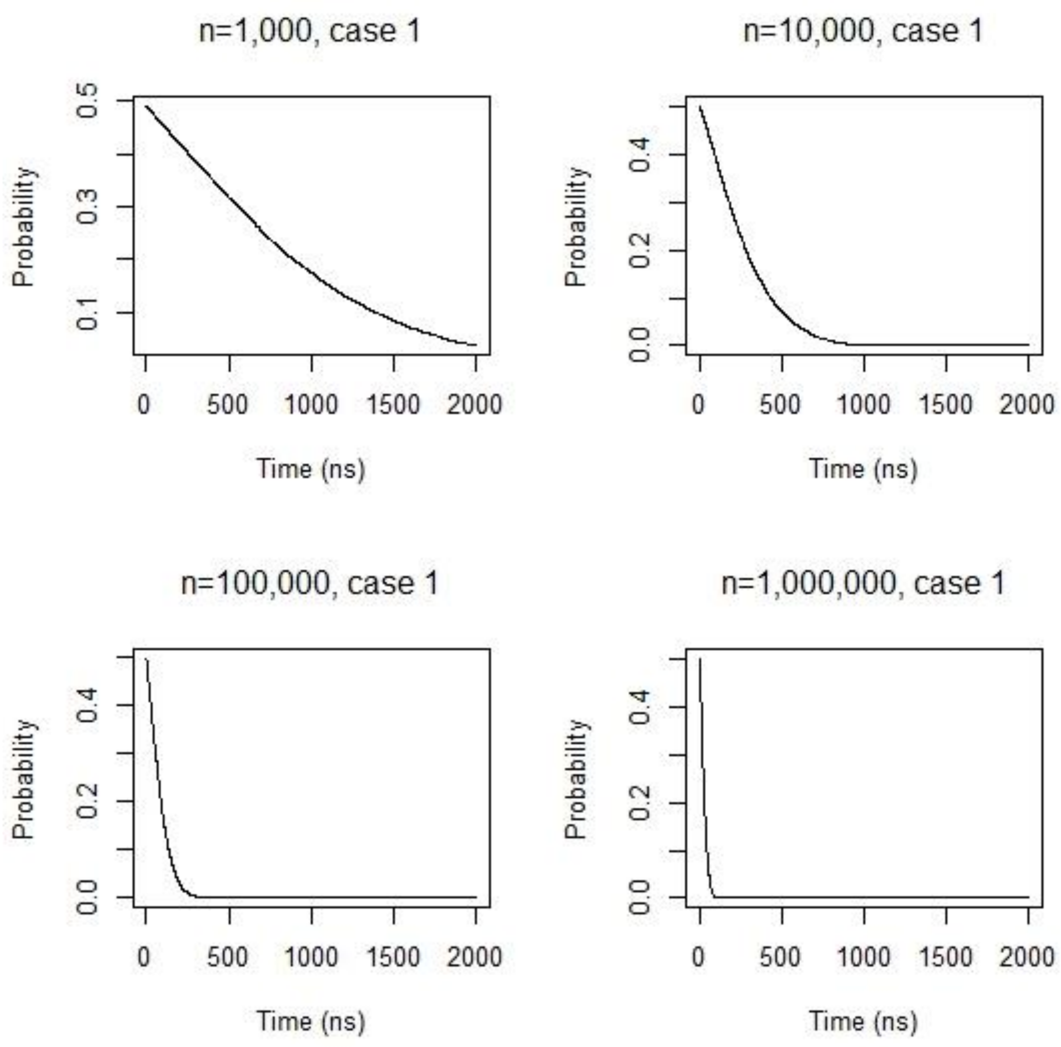

Figure 5.10: Right tail of $\bar{F}_A$ for n variation – Case 1.

Figure 5.10 depicts the right tail of $\bar{F}_A$, i.e. the positive values that will be bounded. The ECDF symmetry discussed earlier is also apparent in Figure 5.10, as for $\varepsilon > 0$, the highest probability of exceeding ε is 0.5 and decays as ε increases. As mentioned earlier, of interest is the bound on the probability that the mean encryption time of n encryptions will exceed the expected encryption time by ε; therefore, bounding the right tail is the main objective of this investigation, and the decay of the probability of exceeding ε, as ε increases, will be further examined.

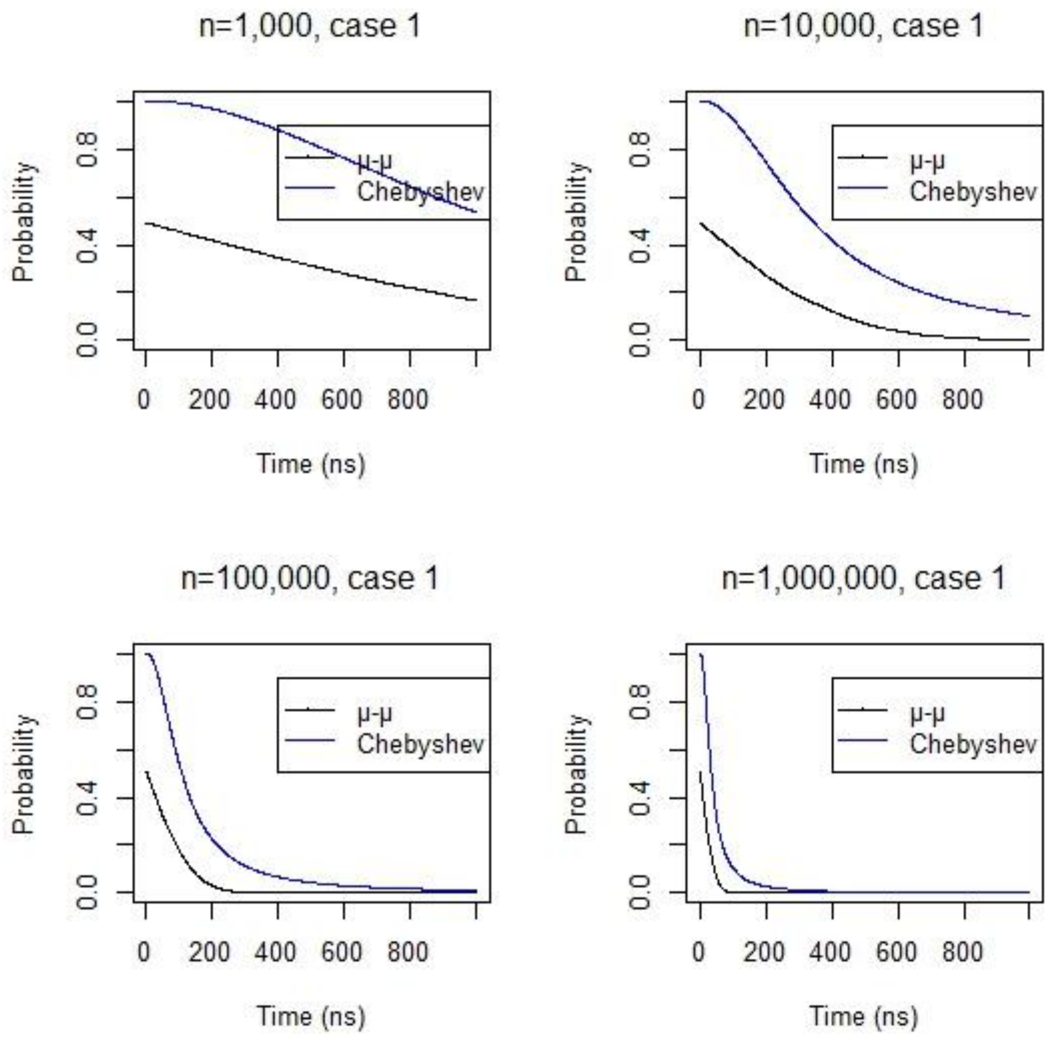

Figure 5.11: Right tail of $\bar{F}_A$ and Chebyshev's bound – Case 1.

A point-wise comparison of the right tail of $\bar{F}_A$ and Equation (5.17) for the values of n shown in Figure 5.10 is presented in Figure 5.11. By applying the appropriate values of n and σ^2 in Equation (5.17) and for different values of ε, ranging from 0 to 4,000 ns with a time step of 10 ns, a point-wise comparison between the right tail of $\bar{F}_A$ and Chebychev's inequality is presented in Figure 5.11. The point-wise comparison allows for displaying the two lines and their alignment, arranged for visual inspection.

Therefore, for a specific ε value in the time axis, the probabilities of the right tail of $\bar{F}_A$ and Chebychev's inequality can be compared. As suggested form Figure 5.11, Chebyshev's inequality captures the slope of $\bar{F}_A$ even for small sample size.

From Equation (5.18), to assure a probability 0.05 for the deviation between sample and true mean to be more than 4,000 *ns*, the required number of encryptions is 1,367. For the difference being of the order 2,000 *ns*, 5,466 encryptions are required to achieve the desired probability which is equal to 0.05. To ascertain the impact of the sample size on the probability γ, from Equation (5.18) and by varying the value of ε, the sample size was calculated and a plot of the probability γ against the sample size *n* was produced.

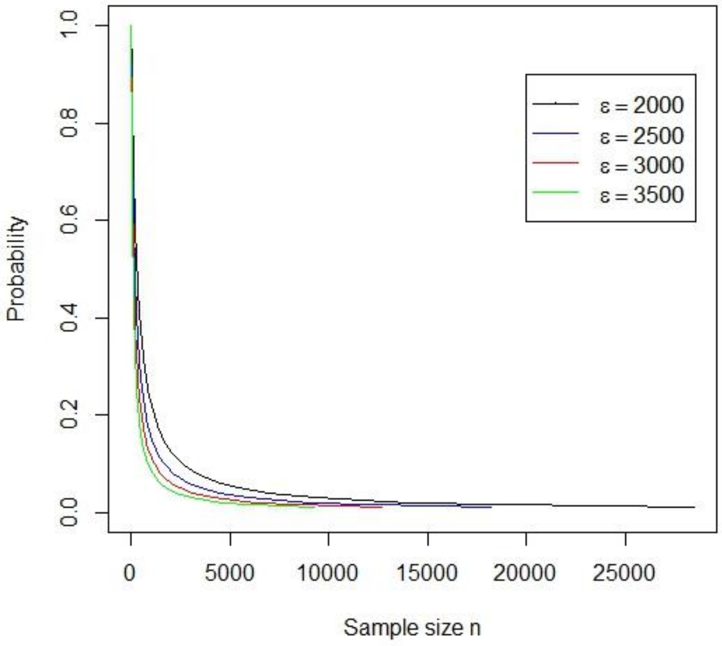

Figure 5.12: Probability γ vs sample size n for Chebychev's bound – Case 1.

As shown in Figure 5.12, the two variables are inversely associated, since when the value of *n* increases, the value of the probability γ decreases. The impact of the variation of ε on the sample size is also depicted. The relationship between the latter is inverse as well, since when the value of ε increases, the probability γ tends to zero for smaller n.

This suggested investigation of the relationship between ε and γ. The values of ε were calculated by fixing n and applying Equation (5.19) to deliver a value for a chosen probability γ. As shown in Figure 5.13 there is an inverse association between ε and γ as well. For higher values of ε, the probability γ decreases. The impact of the variation of n on ε is also depicted. The inverse relationship between n and ε is apparent, as for higher sample sizes the probability γ tends to zero for lower values of ε.

From both the above investigations, as well as from Equations (5.18) and (5.19) as derived from the bound shown in Equation (5.17), it is concluded that there is an inverse association between n and ε.

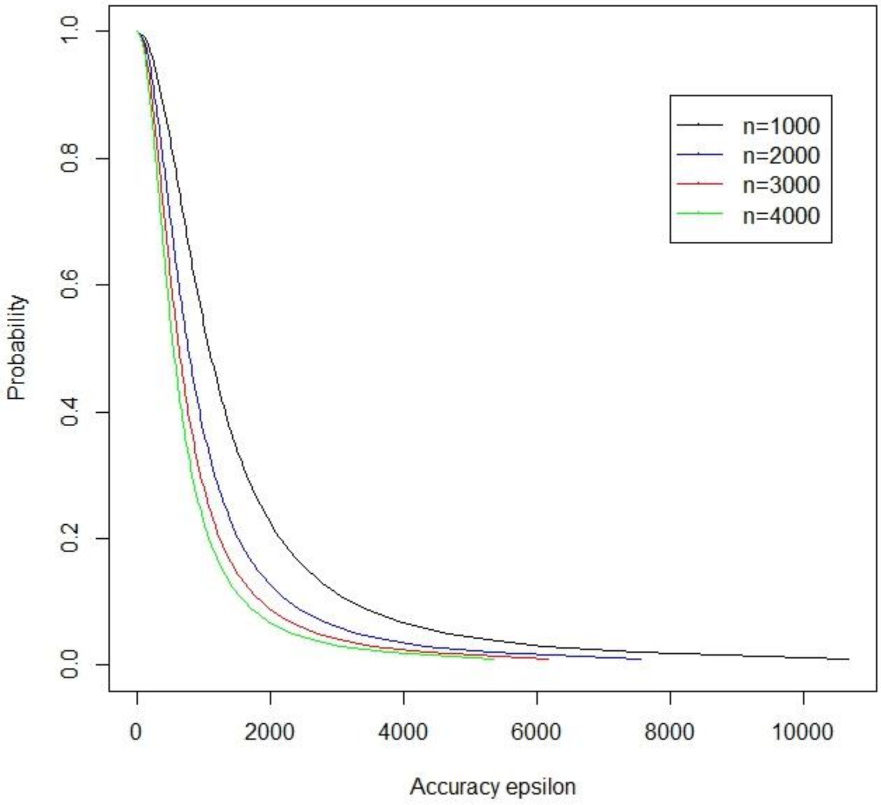

Figure 5.13: Probability γ vs accuracy ε for Chebyshev’s bound – Case 1.

5.6.3.2. Hoeffding's bound

Following similar methodology, an investigation of Hoeffding's bound (5.26) will be implemented. In Figure 5.14 a visual comparison of $\bar{F}_A$ and Hoeffding's bound as resulted from Equation (5.26) is presented. The decay of Hoeffding's curve is somewhat exponential, but fails to mimic the rate of $\bar{F}_A$, as the bound appears relatively loose. For Case 1, $\gamma = 0.05$ and $\varepsilon = 2{,}500$ ns, the minimum number of encryptions n from Equation (5.27) is 9,994 – for a=100,000 ns and b=304,200 ns.

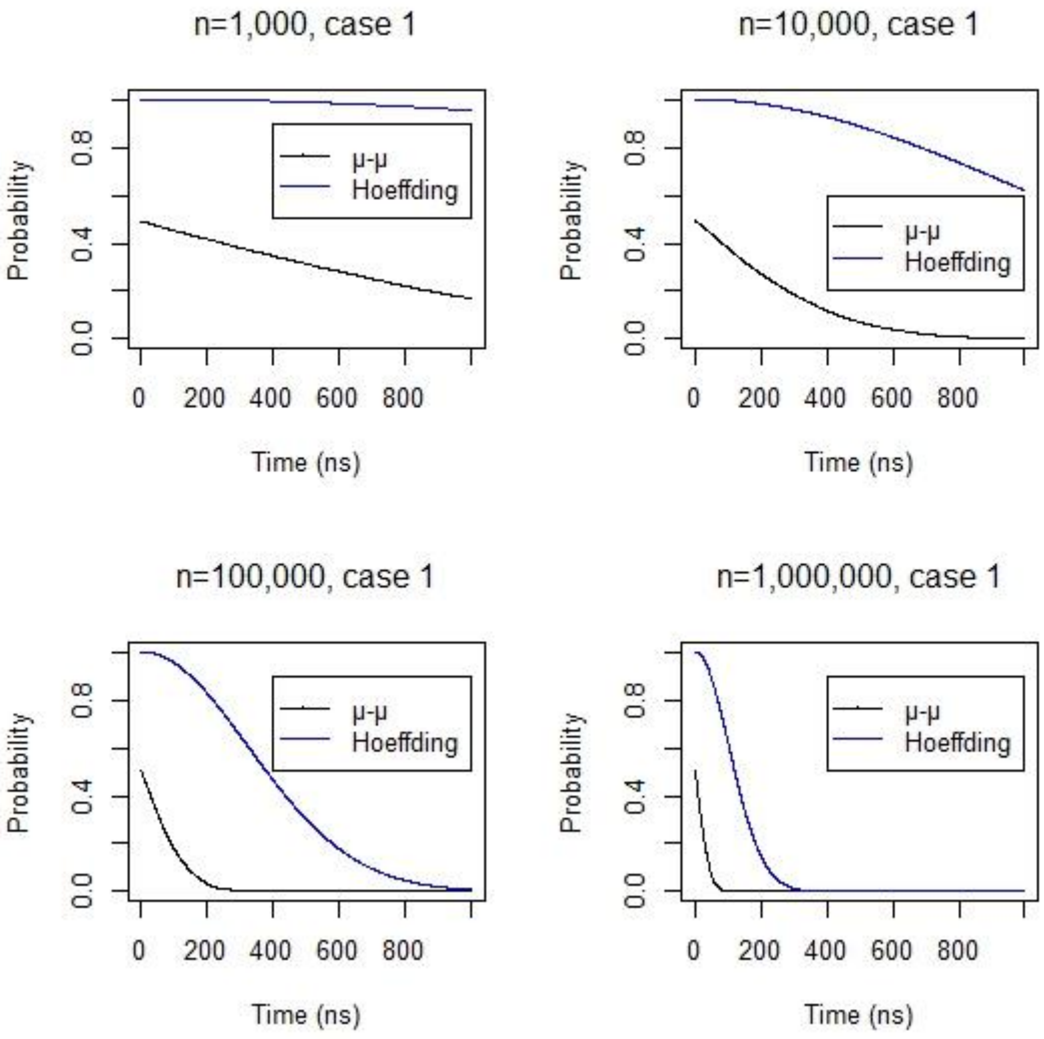

Figure 5.14: Right tail of $\bar{F}_A$ and Hoeffding's bound – Case 1.

A further investigation on the impact of the number of encryptions n on the probability γ was made from Equation (5.27) for variations of ε, as shown in Figure 5.15. There is therefore an inverse association between n and γ. Comparing Figure 5.12 to Figure 5.15, it can be seen that the impact of n on Chebyshev's inequality is stronger than Hoeffding's, as for any ε and n fixed, $\gamma_{Chebyshev} < \gamma_{Hoeffding}$. The inverse association between ε and γ is shown in Figure 5.15.

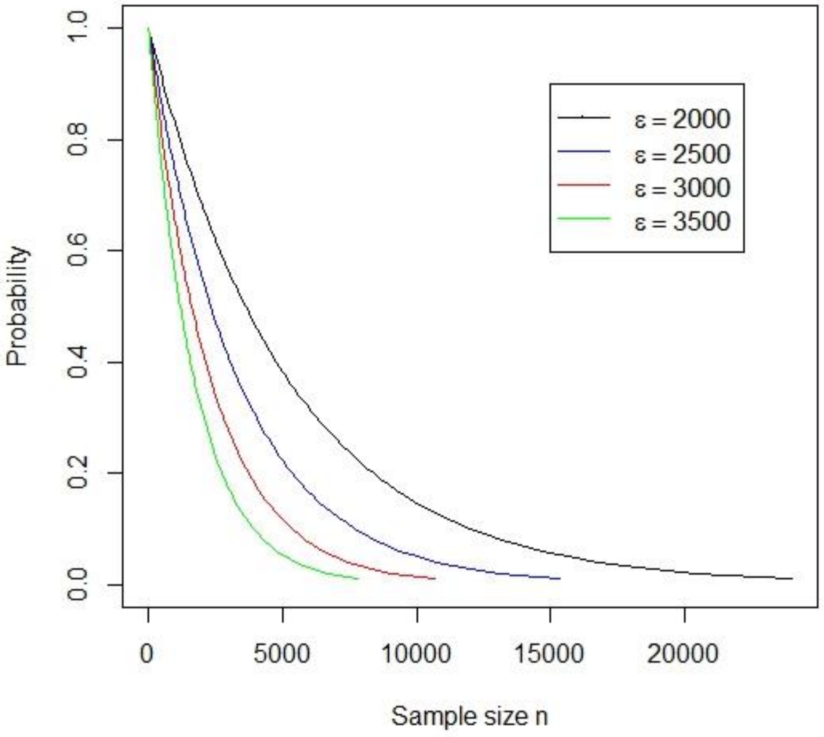

Figure 5.15: Probability γ vs sample size n for Hoeffding's bound – Case 1.

From Equation (5.28), Figure 5.16 is derived, that highlights this relationship for variations of n. From Figure 5.13 and Figure 5.16, it is clear that the accuracy of Chebyshev's is superior to Hoeffding's for any variation of n.

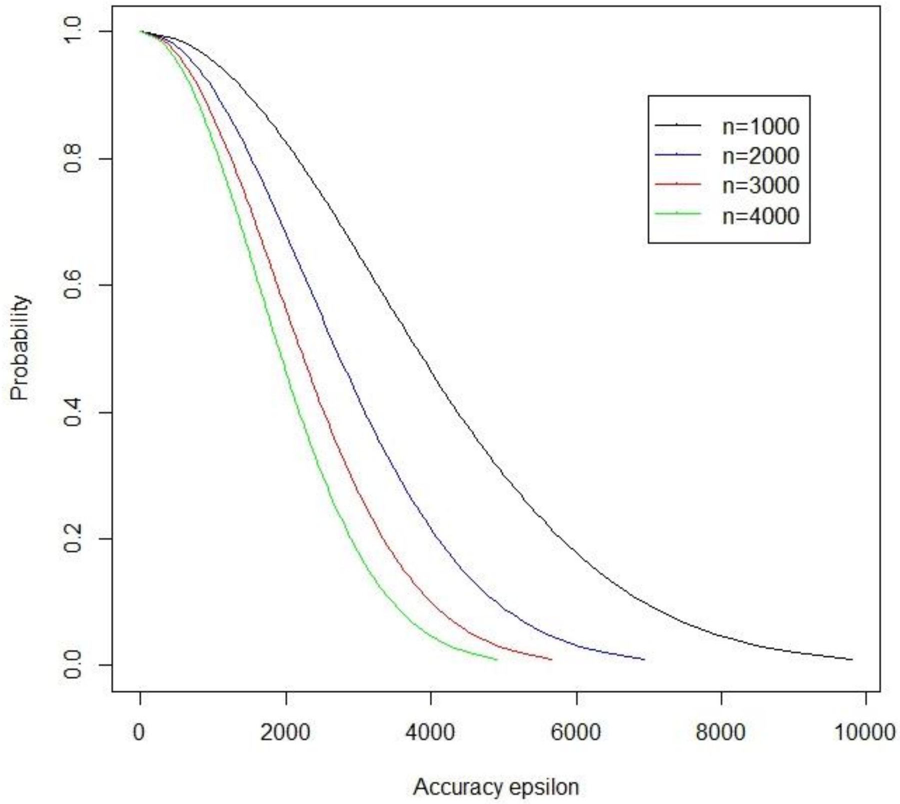

Figure 5.16: Probability γ vs accuracy ε for Hoeffding's bound – Case 1.

5.6.3.3. Bennett's – Bernstein's

So far, two bounds have been investigated, namely Chebyshev's and Hoeffding's inequalities. It is already clear that the first inequality allows for better predictions, which means that the value resulted from Equation (5.17) is closer to the probability of $\hat{\mu} - \mu$ deviating more than ε, given by $\bar{F}_A$. In what follows, Bennett's and Bernstein's inequalities (5.53), (5.58) will be analyzed, to provide sharper bounds. Figure 5.17 depicts the tightness between Bennett's and Bernstein's curves. Both curves decay exponentially, approaching the curve of $\bar{F}_A$ for all variations of n.

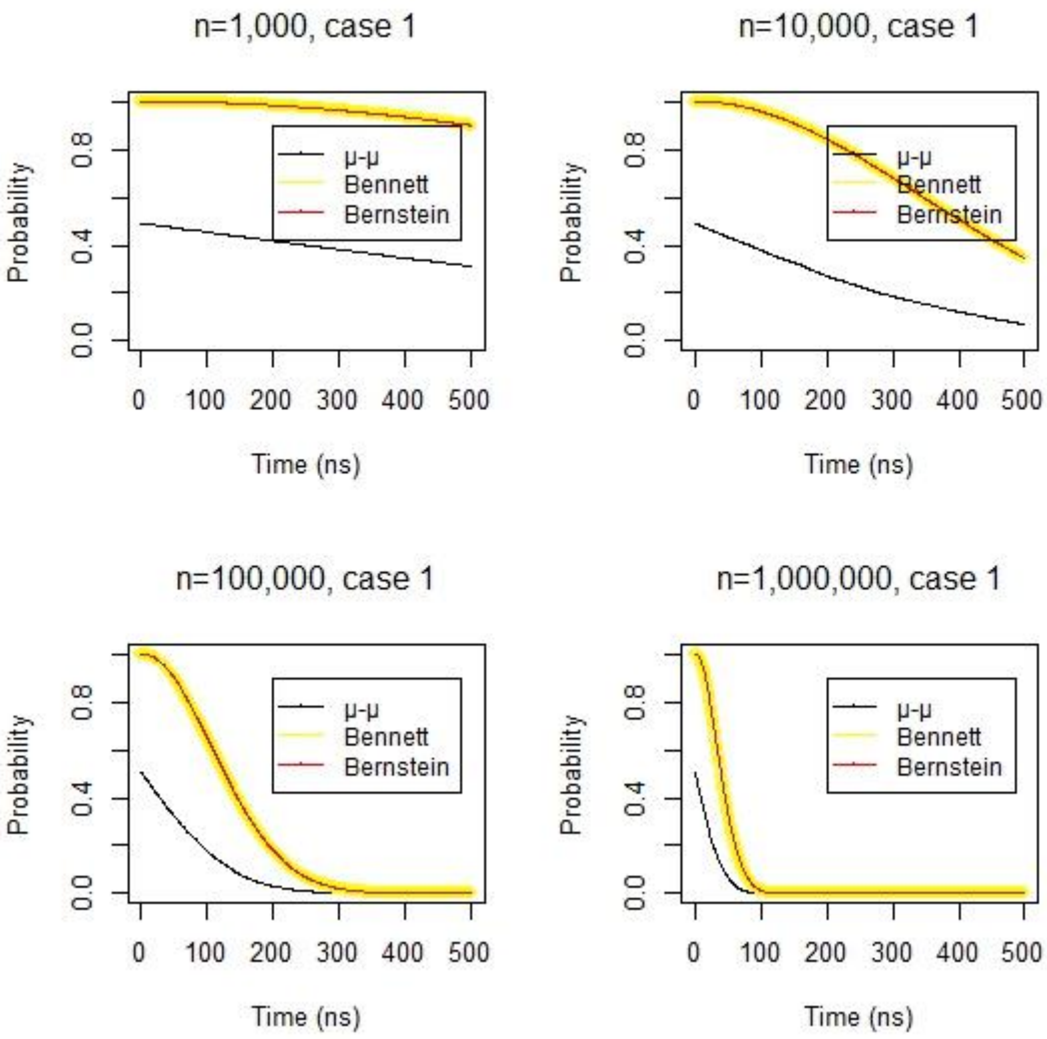

Figure 5.17: Right tail of $\bar{F}_A$, Bennett's and Bernstein's bounds – Case 1.

From Equation (5.60), to assure a probability 0.05 for $\hat{\mu} - \mu$ being more than 4,000 (ns), the required n value is 518, roughly 10 times less than the minimum n calculated from Equation (5.18) for Chebyshev's inequality. Further investigation on the impact of both n and ε on γ was implemented using Equation (5.60). Figure 5.18 visualizes this relationship.

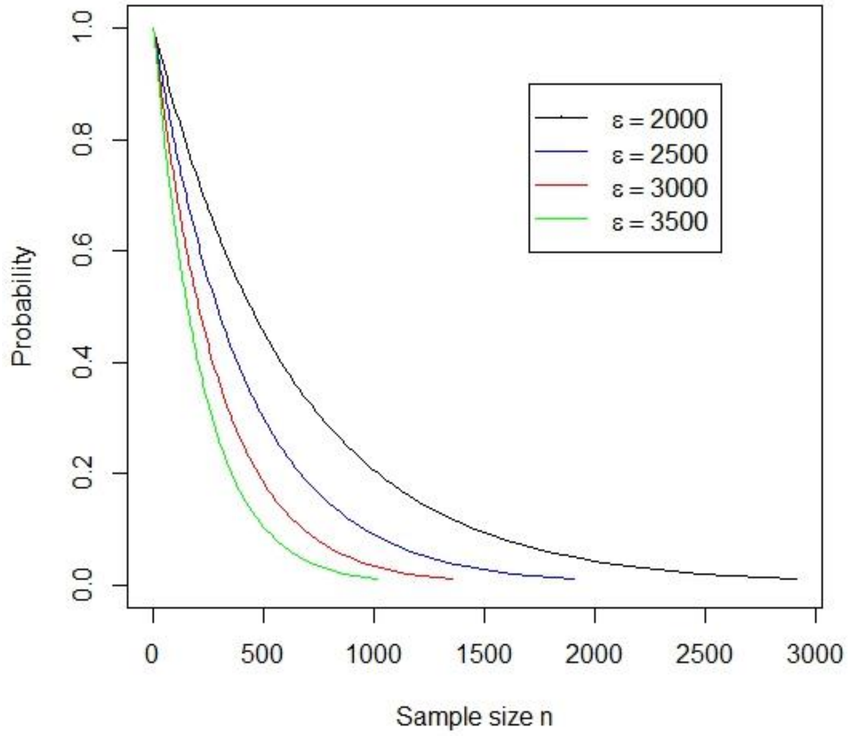

Figure 5.18: Probability γ vs sample size n – Case 1.

As n gets larger, the probability γ gets closer to zero and the rate of the convergence of γ to zero increases by increasing the magnitude of the accuracy ε. In other words, as the number of encryptions increases, it is possible to establish a desired error ε that will enable the control of the threshold probability γ. This γ will upper bound the right tail of $\bar{F}_A$ of event $\{\hat{\mu} - \mu \geq \varepsilon\}$ and the prediction will be accurate.

5.6.3.4. Upper bound comparison for mean

This subsection summarizes and compares the accuracy of each inequality that was investigated in this chapter. For various sample size values n, a point-wise comparison of the right tail of $\bar{F}_A$ and the bounds is presented in Figure 5.19.

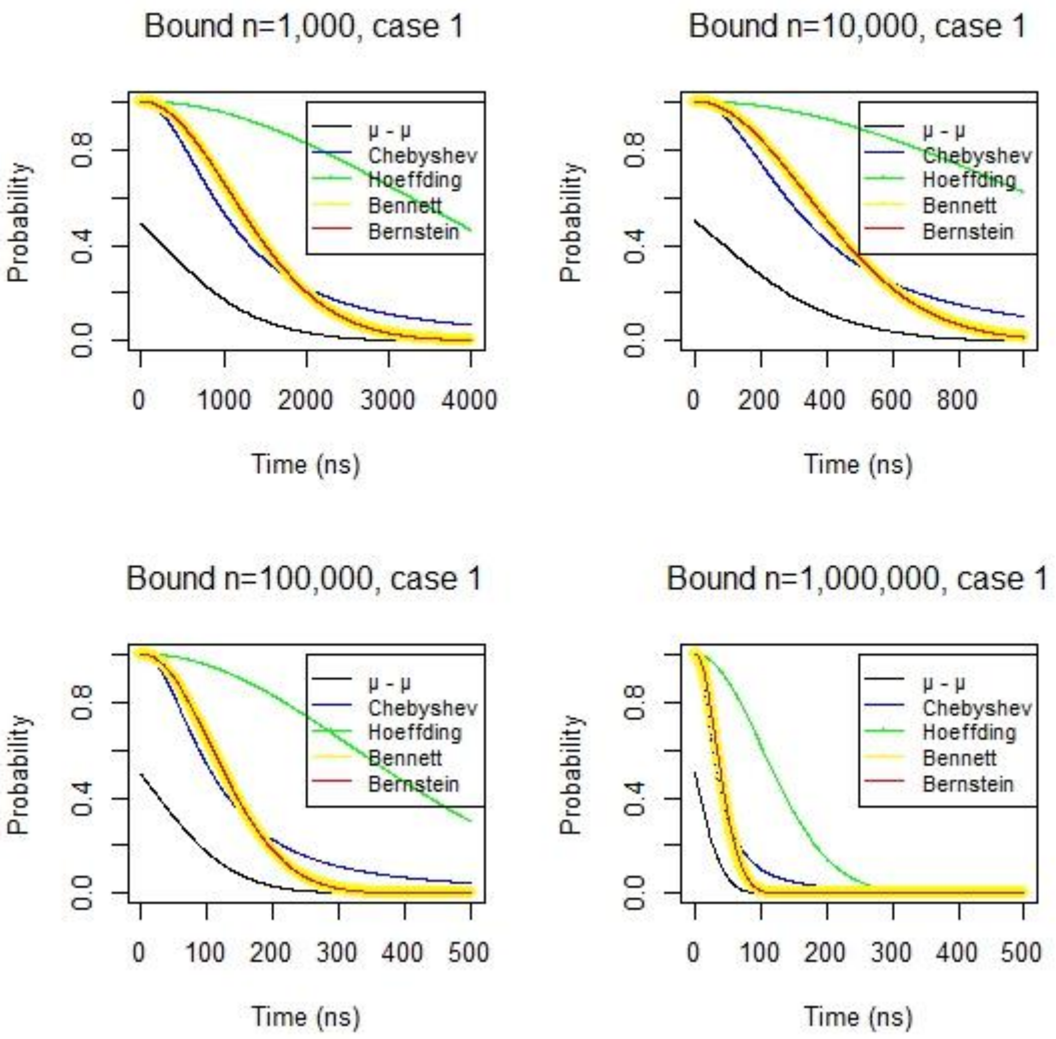

Figure 5.19: Right tail of $\bar{F}_A$ vs bounds – Case 1.

In Figure 5.19 it is shown that Bennett's and Bernstein's inequalities admit a fast decay. Chebyshev's starts somewhat sharp, but fails to follow the curve of Bernstein's and Bennett's that mimic the exponential decay of $\bar{F}_A$. Hoeffding's decay is relatively slow and for small *n*, fails to provide a tight upper bound on $\bar{F}_A$. Conditioning that the tail of $\bar{F}_A$ is of interest, it is derived that for any magnitude of *n*, the following relation holds.

$$\bar{F}_A = P(\hat{\mu} - \mu \geq \varepsilon) \leq P_{Bennet} \approx P_{Bernstein} < P_{Chebyshev} \ll P_{Hoeffding} \quad (5.66)$$

Table 5.4: $P(\hat{\mu} - \mu \geq \varepsilon)$ investigation for n and α variation.

$\boldsymbol{n}$	$\boldsymbol{\alpha}$	$\boldsymbol{\varepsilon\ (ns)}$	$\boldsymbol{\bar{F}_B}$	$\boldsymbol{P_{Bennett}}$	$\boldsymbol{P_{Bernstein}}$	$\boldsymbol{P_{Chebyshev}}$	$\boldsymbol{P_{Hoeffding}}$
1,000	0.02	2,610	0.01	0.07	0.07	0.14	0.7
1,000	0.05	6,525	0	$6E-07$	$9E-07$	0.026	0.1
10,000	0.006	783	0.01	0.07	0.077	0.15	0.7
10,000	0.01	1,305	0	0.0009	0.0009	0.06	0.4
100,000	0.002	261	0.009	0.05	0.05	0.14	0.7
100,000	0.003	391	0.0001	0.0014	0.0014	0.06	0.47
1,000,000	0.0007	91.3	0.002	0.02	0.02	0.12	0.6
1,000,000	0.001	130.5	0	0.0006	0.0006	0.06	0.4

where $\hat{\mu} = \frac{1}{n}\sum_{i=1}^{n} T_i$ is the average time of n encryptions, $\mu = ET_i$ is the mean encryption time of a single execution and $T_i \sim ECDF_{Case\ 1}$ is a r.v. representing the execution time of the ith encryption.

Table 5.4 presents a point-wise comparison of the relation/ranking given by Equation (5.66), for ε being of order α times μ, $0 \leq \alpha < 1$.

Bennett's and Bernstein's inequalities provide a tight bound for $\bar{F}_A$ for all variations of n. For example, if 1,000 encryptions are implemented, then from Table 5.4,

$$\bar{F}_A = P(\hat{\mu} - \mu > 2{,}610) = 0.01 \leq 0.07 \leq 0.14 \leq 0.7$$

The value 0.07 resulted from Bennett's (5.53) and Bernstein's (5.58) inequalities, is tight to the probability generated by $\bar{F}_A$. Chebyshev's value on the other hand is 0.14, which is relatively close to $\bar{F}_A$, however it is weaker compared to the first two bounds.

Hoeffding's curve decays exponentially, but it is not sharp enough to capture the tail of the distribution of $\bar{F}_A$. Also, compared to the first three inequalities, tightness is achieved for larger sample sizes *n*.

5.6.4. Bounding the mean time – Two-sided bound

In this subsection, the behavior of event C in Equation (5.63) is examined. The probability of the absolute difference between the sample mean $\hat{\mu}$ and the theoretical mean μ not deviating more than ε, where $\varepsilon \geq 0$, will be bounded using the two-sided versions of Chebyshev's (5.9), Hoeffding's (5.30) and Bennett's (5.59) inequalities.

By the strong law of large numbers, it is obtained that

$$P\left[\lim_{n\to\infty}\frac{1}{n}\sum_{i=1}^{n}(|T_i - E[T_i]|) = 0\right] \to 1 \qquad (5.67)$$

The probability in Equation (5.67) indicates that with enough samples, the empirical mean is a good approximation to its true mean. According to the strong law of large numbers, the sample mean converges almost surely to the expected mean. A quantitative version of the law of large numbers for bounded variables is the investigation of that rate of convergence [85], by bounding the tail of $\bar{F}_C$. The decay rate, given by a bound, will provide useful information about the impact of finite values *n* on the convergence $|\hat{\mu} - \mu| \to \varepsilon,\ \varepsilon \geq 0$.

Figure 5.20 depicts the right tail $\bar{F}_C$ bounded by inequalities given by (5.9), (5.30) and (5.59). For all variations of the sample size *n*, the three inequalities admit an exponential decay, with Bernstein's and Chebyshev's being able to mimic the tail of $\bar{F}_C$. Bernstein's appears sharper to the tail, while Hoeffding's performance is somewhat

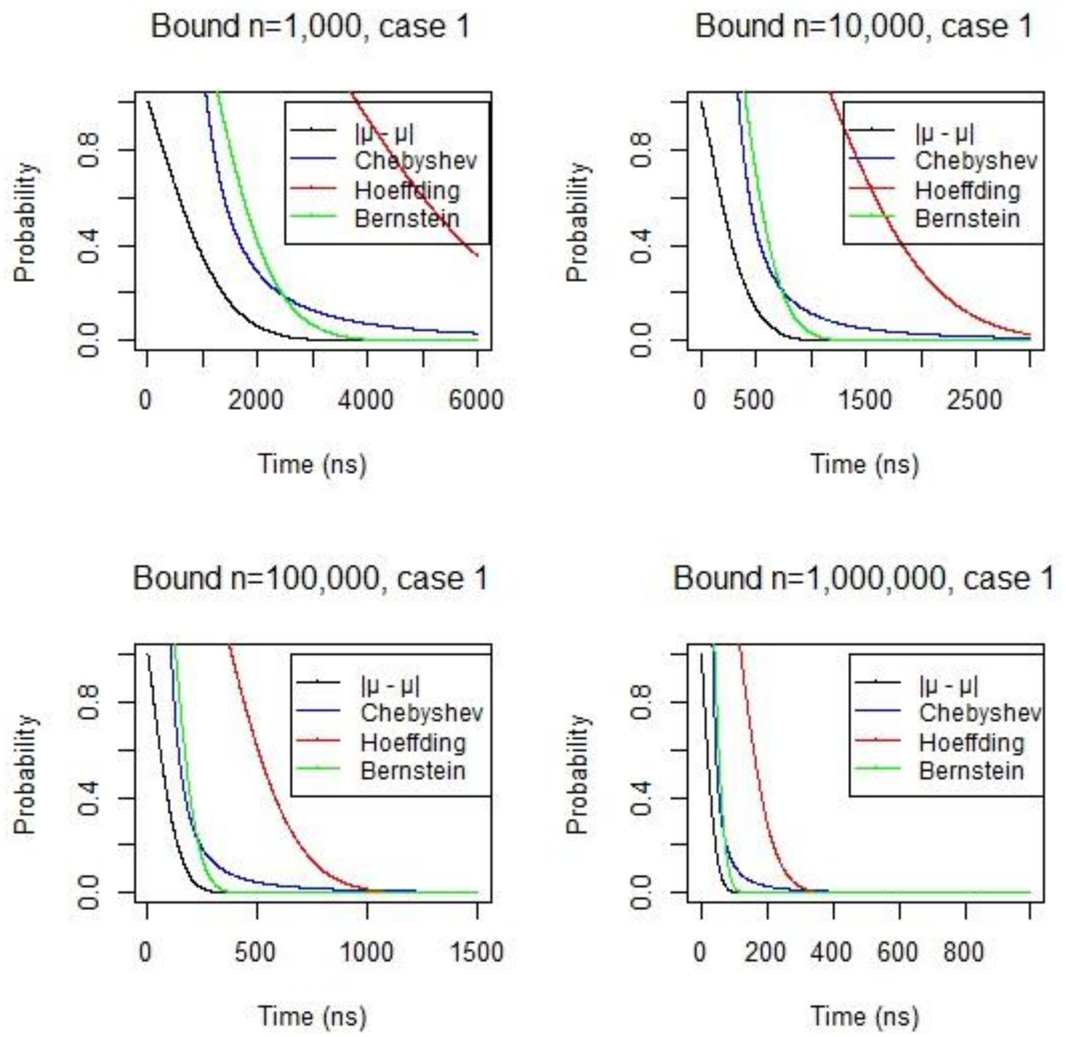

Figure 5.20: Right tail of $\bar{F}_C$ vs bounds – Case 1.

poor compared to the rest of the inequalities. Therefore, under the condition that the right tail of $\bar{F}_C$ is examined, i.e. ε is large enough, the following relation holds

$$\bar{F}_C = P(|\hat{\mu} - \mu| \geq \varepsilon) \leq P_{Bernstein} \leq P_{Chebyshev} \ll P_{Hoeffding} \qquad (5.68)$$

Table 5.5 highlights Bernstein's ability to provide a tight bound even for small n values. For 1,000 encryptions, $P(|\hat{\mu} - \mu| \geq 3{,}000) = 0.005$ as resulted from the generated distribution $\bar{F}_C$. Under the same parameterization on n and ε, Bernstein's value is 0.06, while Hoeffding's fails to follow and Chebyshev's performance is weak.

Table 5.5: $\bar{F}_C$, Bernstein's, Chebyshev's and Hoeffding's values for various n and ε.

n	$\varepsilon\ (ns) = \alpha\mu$	$\bar{F}_C$	$P_{Bernstein}$	$P_{Chebyshev}$	$P_{Hoeffding}$
1,000	3,000	0.0005	0.06	0.1	1
1,000	4,000	0.0001	0.006	0.07	0.9
10,000	1,000	0.002	0.03	0.1	1
10,000	1,200	0.0003	0.005	0.08	1
100,000	300	0.005	0.04	0.1	1
100,000	400	0.0002	0.002	0.07	0.9
1,000,000	100	0.003	0.02	0.1	1
1,000,000	150	0	0.0001	0.05	0.6

Chebyshev's values are relatively close to Bernstein's, in terms of tightness, for small or moderate *n*, with the tightness relaxed when *n* gets larger, failing to follow Bernstein's sharpness.

5.7. Discussion

In this chapter an adaptive security scheme that suggests a new approach in the area of low energy encryption has been presented. The method relies on the application of Chernoff – type bounds on the tail probability that the mean time of *n* encryptions will exceed the expected time by a desired threshold so that the probability on an extreme value can be bounded. This method has also been used to provide a bound for the probability that the overall execution time of *n* encryptions will not exceed *n* times the mean encryption time of a single execution. Two-sided inequalities have been calculated and presented for those exponential bounds, for both the mean and overall encryption time. This provides knowledge about the rate of convergence of the sample mean to the true mean or the sum of *n* encryptions times to *n* times the mean, as concentration inequalities quantify what is known from the law of large numbers. Another advantage of the presented methodology is that relaxes the CLT's assumption in Chapter 4, as number of executions can be considered finite. CLT presented in Chapter 4, is an asymptotic result and assures that the time distribution of the *n* encryptions is approximately Gausian as $n \to \infty$. In this chapter, the scope was to develop a framework for investigating the trade-off between the number of encryptions *n*, the level of accuracy ε and the tightness of each inequality to the right tail of the distribution.

Performance evaluation of the probabilistic concentration inequalities was presented for both upper and two-sided bounds of mean and sum. This framework is flexible, Chebyshev's, Hoeffding's, Bennett's and Bernstein's bounds are distribution free and no assumption needs to be made for the distribution.

Bennett's and Bernstein's inequalities are approximately equal as presented. The investigation highlighted their superior performance compared to the other two inequalities, as they bound tightly the right tail of the distribution even for small or moderate sample size *n*. This makes it possible to set up and optimise energy efficient encryption schemes for policy makers with relatively small number of expected encryptions. A typical example of an encryption scheme with sample size equal to 1,000 executions might be the daily contactless payments of a central coffee shop or a supermarket. On the other hand, an expected number of transactions of order 10^6 or more, might be the daily usage of contactless passes in London's tube. In both situations, Bernstein's bound will result in tight predictions.

Although Hoeffding's inequality did not perform well compared to Bernstein's, it could contribute at the early stages of an encryption scheme, where no data may be available for a performance analysis. This inequality requires the time to be independent random variable and bounded, i.e. the best and worst case scenario of the algorithm's encryption time. Having that knowledge, rough approximations of the scheme's performance could be made for *n* encryptions that the algorithm is expected to be executed during a specified time window, and by incorporating knowledge from Chapter 4 regarding the limiting behaviour of scheme's encryption time, an optimal set up will be enabled. During the operation of the scheme, the recorded data will provide the decision maker with statistical confidence that the sample variance is a valid approximation of the true variance. Finally, having knowledge of the variance, tighter predictions can be made by applying Bernstein's bound, while any modifications at the scheme will be made at that stage.

Chapter 6 - Statistical considerations

Traditional approaches that evaluate the encryption performance in terms of the energy cost, mainly compare different algorithms and/or parameters in terms of effectiveness and provide results on their behaviour with respect to their impact on energy consumption [3]. However, the correlation between encryption parameters and their overall impact on energy consumption is not utilized to provide a unified adaptive security scheme.

It is envisioned that the offset between minimum energy consumption and maximum encryption strength will be essential to provide low energy encryption solutions. Furthermore, the relationship between energy consumption and functional encryption parameters, as well as the dependencies between the latter, has to be taken into consideration to efficiently adjust encryption parameters in an adaptive security scheme.

In this chapter, a statistical analysis technique is presented that identifies the impact of the encryption parameters on the energy consumption of the system, both individually and as a total. Specifically, multiple linear regression is utilised to determine whether energy consumption (*ENERGY*) can be predicted from the encryption parameters of the model, namely data size (*DATA*), key size (*KEY*), mode of operation (*MODE*) and padding scheme (*PADDING*).

6.1. Variables and transformation

The analysis is based on a data of 57600 sample size - 100 trials of a sample of 576 encryption scenarios - simulated for all possible combinations between encryption parameters. The results of the regression analysis are presented and explained later in this chapter. The independent variables *MODES*, *PADDING*, *KEY* and *DATA* were tested for correlation. The correlations that occurred among the variables were taken in account in the estimation of the coefficients by SPSS.

Since all predictors apart from the data size are categorical variables, they do not convey numeric information and therefore they should not be included in the regression model. Instead, each value of the categorical variable is represented in the model with an indicator variable [86]. The nominal variable *MODE* takes on four levels that have been coded as *CBC, CFB, ECB* and *OFB*. *CBC* is the reference category and therefore the coefficients of the other three variables are interpreted in comparison to that one, the impact of which is included in the constant coefficient. Similarly, the nominal variable *PADDING* takes on three levels that have been coded as *PKCS5*, *NO_PADDING* and *ISO*, with the first of these three being the reference category. Finally, the ordinal variable *KEY* has been coded as *KEY_SIZE_1* (reference category), *KEY_SIZE _2* and *KEY_SIZE _3*.

6.2. Assumptions and exploratory data analysis

Linear regression models rely upon five principal assumptions, namely linearity, normality, independence, homoscedasticity and non multi-collinearity, about the predictor variables, the response variable and their relationship [87]. The validity of the results requires that these assumptions be satisfied [88]. If any of these assumptions is violated, then the results may not be trustworthy.

Some methods for assessing the assumptions of the regression model are based on the residual analysis. The residual represents the difference between the regression predictions and the actual data [89]. Standardized residuals, which are calculated by dividing each residual by its standard error, are plotted against the predicted values to assess the assumptions of the regression model and also to evaluate the goodness of fit.

In this section, the assumptions of the regression model are investigated.

6.2.1. Linearity

Regression models assume that there is a linear relationship between the dependent and each independent variable. However, when working with real world data, the assumption of linearity might not be met and the results of the raw data may be untrustworthy.

Transformation of the data is the one of the most common ways to deal with this problem. For non-linearity problems, transformation of the dependent variable, independent variables, or both, may be necessary [90]. In this study, evaluation of linearity led to the log transformation of *DATA* and *ENERGY* [91]. Specifically, the decimal logarithm of base 10 has been used in order to efficiently handle the relationship between the predictor variable *DATA* and the response variable.

The assumption of linearity is assessed by examining the relationship between the response variable and the predictors. Review of the partial scatterplot of the independent variable *DATA* and the dependent variable *ENERGY* indicates linearity is a reasonable assumption. The boxplot in Figure 6.1 shows the linear relationship between the logarithm of *ENERGY* and the logarithm of *DATA* – as the data size increases, there is an upward trend in the energy consumption. The four rectangles represent the second and third quartiles for the four different data sizes (16, 1000, 2000 and 4000 bytes respectively) in bytes. The first and fourth quartiles are shown by the lines (whiskers) extending vertically from the boxes and indicate variability outside the upper and lower quartiles. The data points beyond the whiskers represent the outliers. Although outliers exist for all data sizes, the median of each data size, which is represented by the horizontal line that divides the box into two parts, increases linearly with the increase of the energy consumed during the encryption.

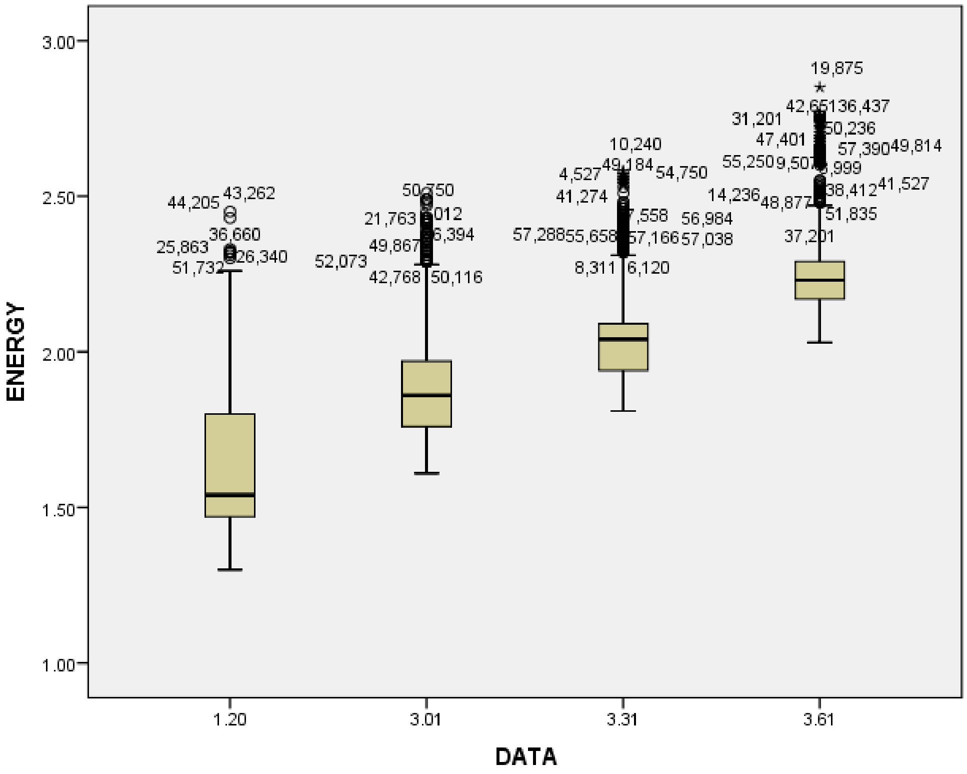

Figure 6.1: Boxplot of ENERGY and DATA.

However, categorical variables are nominal-levelled, with more than two groups and therefore their relationship with the continuous variable *ENERGY* cannot be described meaningfully as a linear one [92]. Instead, a Pearson’s correlation test [93] was run to measure the strength of the linear relationship between the dependent and each of the independent variables. The test indicated that there is a statistically significant linear relationship between the output variable and the predictors [93]. In order to decide whether there is any or no evidence to suggest that linear correlation is present in the population, a significance test is performed. The null hypothesis H_0 is tested that there is no correlation in the population against the alternative hypothesis H_1 that there is correlation. SPSS reports the p-value for this test always ≤ 0.006 as shown in Table 6.1, where the total number of points tested (N) was $100 \times 576 = 57600$ in all cases. With a p-value below 0.05, the hypothesis test H_0 is rejected, meaning that there is strong evidence of linearity between the variables.

Table 6.1: Correlations

ENERGY	ENERGY	DATA	CFB	ECB	OFB	NO PADDING	ISO	KEY SIZE 2	KEY SIZE 3
Pearson Correlation	1	.787**	.026**	-.062**	.025**	-.012**	.011**	.044**	.054**
Sig. (2-tailed)		.000	.000	.000	.000	.005	.006	.000	.000

**. Correlation is significant at the 0.01 level (2-tailed).

6.2.2. Normality

The most common method to assess the normality assumption is to study how close the distribution of the residuals conforms to a normal distribution. The normality assumption is assessed by examining the normal probability plot and the histogram of the residuals [64]. A normal probability plot is obtained by plotting the residuals against the associated values from a theoretical standard normal distribution in a way that data points should form a diagonal and approximately straight line. If random errors are normally distributed, the data points will lie close to the line. Instead, if they deviate significantly from the diagonal line, then the residuals might not come from a normal distribution [64]. A histogram is another way to graphically represent the distribution of the data. It is constructed by splitting the data range into bins of equal size and measuring the frequency of the data points for each bin [64].

In this study, examination of the normal probability plot of residuals and the residual histogram indicated normal distributional shape and therefore normality was a reasonable assumption [93], as shown in Figure 6.2 and 6.3 respectively. Both figures have been obtained using SPSS and the whole data set of 57600 instances has been used.

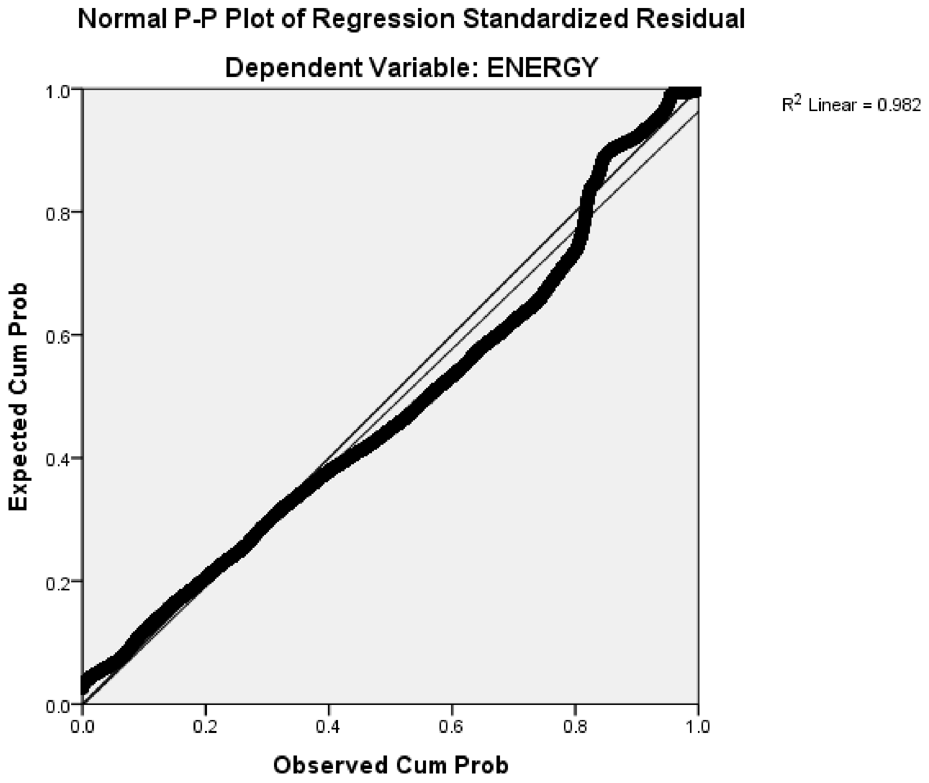

Figure 6.2: Normal probabiltiy plot.

The normal P-P plot in Figure 6.2 plots the observed cumulative probability of the residuals against the predicted cumulative probability. The X-axis represents the observed cumulative probability of the residuals, as resulted from their frequency distribution. The Y-axis represents the cumulative density of a standard normal distribution. The plot indicates that it is reasonable to assume that random errors are drawn from approximately normal distributions. When data is normally distributed, the result is a diagonal line in a linear fashion, where the data points lie close to the diagonal line. R^2 measures the proportion of the total variation in the dependent variable that is explained by variation in the independent variables [100]. As shown in the output, $R^2 = 0.98$ indicates relatively high linearity of the fitted line and therefore the P-P plot suggests that the normality assumption is met.

In Figure 6.3 the histogram of the standardized residuals is used to evaluate the normality assumption. As long as the histogram matches the bell-shaped curve, which is the density of the standard normal distribution, the residuals follow a standard normal distribution (*mean=0*, *σ=1*) [94]. As shown in the figure, a histogram distributed evenly around zero, with most of the frequencies gathered in the centre, indicates that the normality assumption holds.

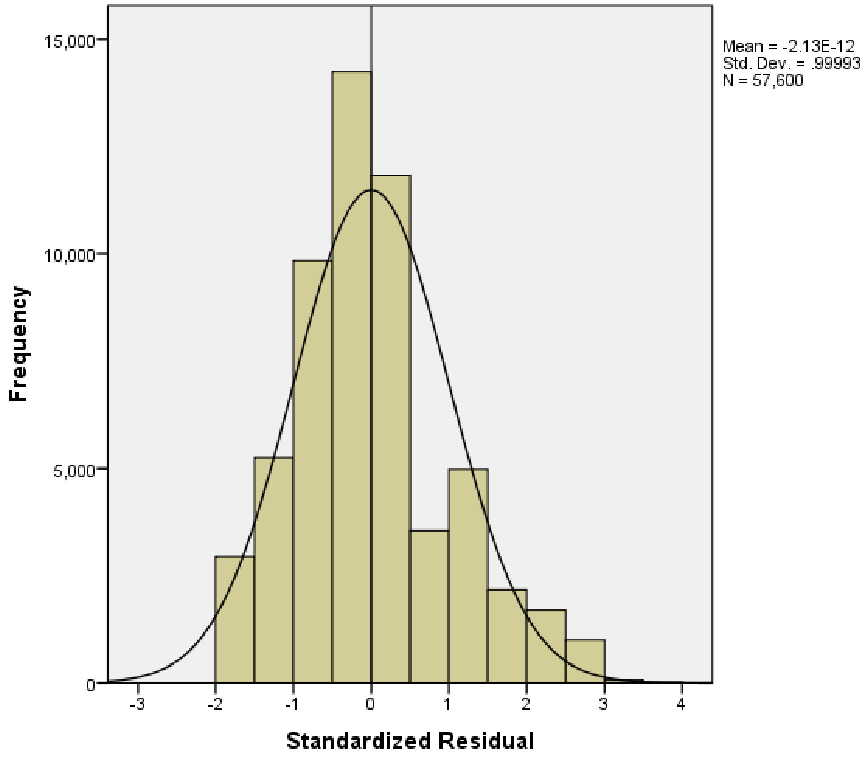

Figure 6.3: Standardized residuals histogram.

Although there is a low positive asymmetry in the residual distribution with a slight right skew, only few observations fall out of the [-3, 3] range as shown in the histogram, confirming the conclusions from the P-P plot regarding the normality assumption. This is also depicted in statistics Table 6.2 below.

Table 6.2: Statistics

Standardized Residual

Mean		.0000000
Std. Deviation		.99993055
Percentiles	68	.2652233
	95	1.9344824
	99.7	2.704627

According to the 68-95-99.7 rule, in a normal distribution 68% of the data points lie within 1 standard deviation of the mean, 95% are located within 2 standard deviations and 99.7% of the values are located within 3 standard deviations of the mean [95]. Therefore, for an observation x from the distribution with μ=0 (mean of the distribution) and σ=1 (standard deviation), it is expected that:

$$P(\mu - \sigma \leq x \leq \mu + \sigma) = P(-1 \leq x \leq 1) \approx 0.68 \qquad (6.1)$$

$$P(\mu - 2\sigma \leq x \leq \mu + 2\sigma) = P(-2 \leq x \leq 2) \approx 0.95 \qquad (6.2)$$

$$P(\mu - 3\sigma \leq x \leq \mu + 3\sigma) = P(-3 \leq x \leq 3) \approx 0.997 \qquad (6.3)$$

which is in line with the results shown in Table 6.2 and therefore it can be concluded that the residuals come from a roughly normal distribution.

6.2.3. Independence

Another assumption of the regression model that has to be assessed is the independence of variables. The validity of independence is mainly based on unbiased sampling, meaning that the observations are not related to one another. If knowing the value of one variable does not reveal any information about the prediction of any other variable, then the variables are independent of each other [96]. In this study, the assumption of independence was assured by the way the experiment was conducted.

Apart from the examination of the way data was collected, another method to assess the independence of variables is to plot the residuals against the case identification number, or their collection order etc. A dependency is shown by an upward or

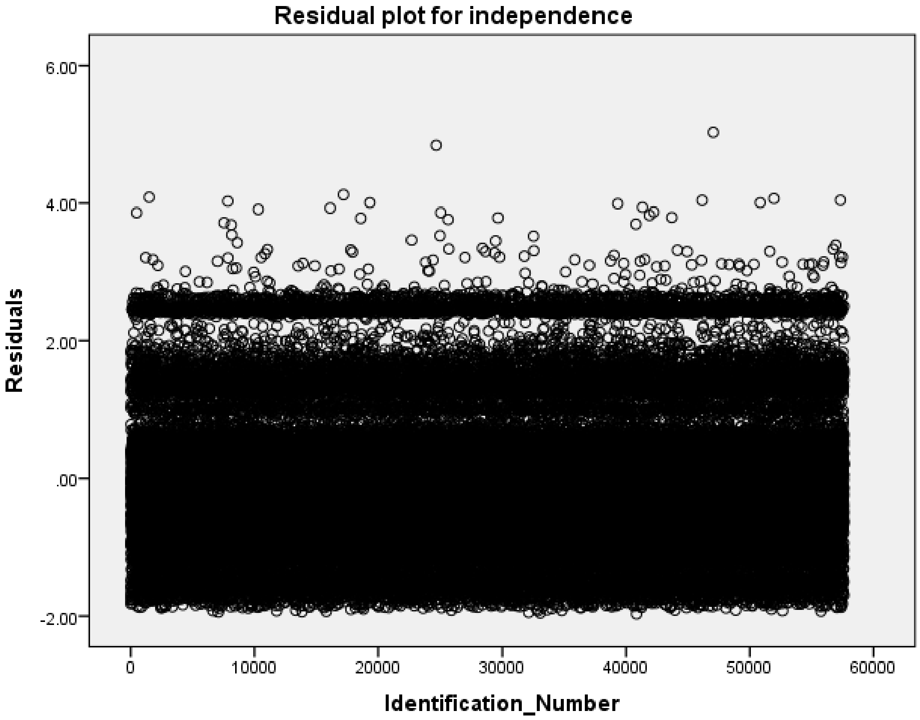

Figure 6.4: Residual scatterplot.

downward trend [97] and the assumption of independence is valid if there is no pattern in the plot.

In this study, review of the standardized residuals scatterplots which indicated an evidence of independence, as shown in Figure 6.4. The plot shows the relative randomness of the residuals above and below zero, validating the assumption of independence.

Finally, another statistical examination of independence is to run a Durbin-Watson test on the residuals. The Durbin-Watson test determines whether there is a relationship between consecutive residuals and is defined as in [96]

$$d = \frac{\sum_{i=2}^{n}(e_i - e_{i-1})^2}{\sum_{i=1}^{n} e_i{}^2} \tag{6.4}$$

Where $0 \leq d \leq 4$ and i is the time period between consecutive residuals e_i and e_{i-1}.

Table 6.3: Model summary and Durbin-Watson test for independence

R	R Square	Adjusted R Square	Std. Error of the Estimate	Durbin-Watson
.796[a]	.633	.633	.17908	1.998

As shown in Table 6.3, the observed $d = 1.998$, means that there is no cause for concern regarding the independence assumption, as it is within the [1.5 - 2.5] accepted limits [97].

6.2.4. Homoscedasticity

Homoscedasticity is another assumption of the regression model that has to be assessed. Homoscedasticity holds if the variance of the residuals is constant. Constant variance is assessed by examining the residuals of the fitted model. If the variance of the residuals is not constant, the assumption is violated and in the residual plot the non-constant variance – also known as heteroscedasticity – is indicated by a cone-shaped pattern [97].

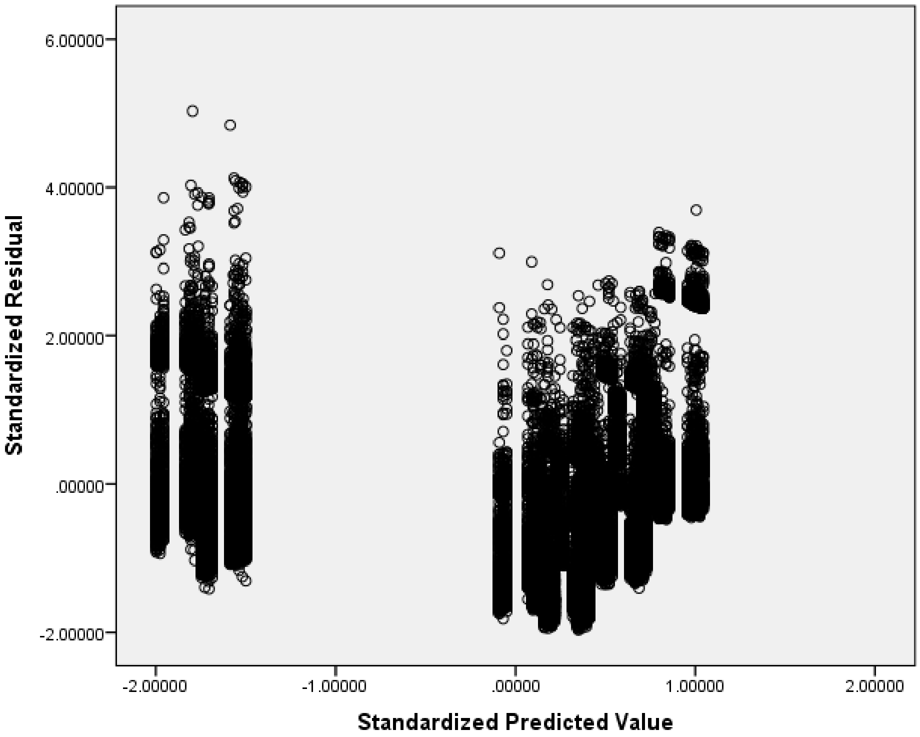

Figure 6.5: Standardized residual scatterplot.

The plot of the standardized residuals against standardised predicted values in Figure 6.5 indicates that the homoscedasticity assumption is valid, as no cone-shaped pattern is shown. The figure has been obtained using SPSS and the whole data set of 57600 instances has been used. The figure shows a scatterplot of the standardized residuals to the predicted values, where the residuals lie in the range [-3, 3] according to the 68-95-99.7 rule [95]. The residuals appear to be randomly scattered over and below zero, suggesting that there is no violation of the homoscedasticity assumption [94]. The homogeneity of variance across the entire range of the fitted values and the lack of patterns (i.e. higher predicted values have lower residuals) provided evidence of homoscedasticity [97].

6.2.5. Non multi-collinearity

Multi-collinearity refers to the inter-correlation among the predictor variables of the regression model, such that their effects cannot be separated because they explain the same variability in the predicted outcome [98].

Table 6.4: Coefficients and Collinearity statistics

	Unstandardized		Standardized	t	Sig.	Collinearity	
	B	Std. Error	Beta			Tolerance	VIF
(Constant)	1.229	.003		401.890	.000		
DATA	.248	.001	.787	312.157	.000	1.000	1.000
CFB	.007	.002	.011	3.525	.000	.667	1.500
ECB	-.038	.002	-.055	-17.799	.000	.667	1.500
OFB	.007	.002	.010	3.177	.001	.667	1.500
KEY_SIZE_2	.059	.002	.094	32.424	.000	.750	1.333
KEY_SIZE_3	.063	.002	.101	34.730	.000	.750	1.333
NO_PADDING	-.005	.002	-.008	-2.755	.006	.750	1.333
ISO	.005	.002	.007	2.534	.011	.750	1.333

a. Dependent Variable: ENERGY

Collinearity diagnostics test are run in order to measure the strength of the correlation among predictor variables and how this affects the stability and variance of the regression estimates [99]. The severity of multi-collinearity is quantified by three measures – tolerance, variance inflation factor (VIF) and condition index. Tolerance is the percentage of the variance in a predictor variable that cannot be explained by other variables. When tolerance is ≤0.1, it is considered problematic, as it reveals multi-collinearity among variables [98]. VIF measures how much the variance of the regression estimates is increased because of collinearity. A VIF >10 indicates multi-collinearity as well [98]. The condition index indicates the severity of multi-collinearity. A condition index >30 suggest a serious problem with multi-collinearity, while a condition index >15 indicates possible multi-collinearity issues [98].

The collinearity statistics shown in Table 6.4 have been obtained using SPSS and the 57600 instance data set has been used. The table provides information about several aspects of multiple linear regression. Specifically, *B values*, or coefficients, provide information about the relationship between energy consumption and each predictor variable – in this case encryption parameters and their variation. It measures the impact of each predictor variable on the predicted value of the response variable [99]. A positive value demonstrates a positive relationship between the predictor and the response variable, whilst a negative value represents a negative relationship. The associated *standard error* indicates to what extend each coefficient vary across different samples. The *significance* of the *t-test* associated with a B value indicates whether the predictor variable contributes significantly to the accuracy of the model [99]. The smaller the value of significance, the greater the contribution of the predictor variable to the model [100]. If the significance value is less than 0.05, the t-test is significant, which means that the predictor variable contributes significantly to the model [101]. The *standardized Beta* values do not depend on the units of measurement of the variables and are therefore easier to interpret and compare. These values provide the number of standard deviations that the response variable will change when the predictor variable changes by one standard deviation [102].

Table 6.5: Collinearity diagnostics

Eigenvalue	Condition Index	Variance Proportions								
		(Constant)	DATA	CFB	ECB	OFB	ISO	KEY_S	KEY_S	NO
4.168	1.000	.00	.01	.01	.01	.01	.01	.01	.01	.01
1.001	2.040	.00	.00	.04	.15	.04	.01	.12	.12	.01
1.000	2.042	.00	.00	.01	.03	.01	.23	.00	.00	.23
1.000	2.042	.00	.00	.25	.00	.25	.00	.00	.00	.00
.999	2.043	.00	.00	.04	.15	.04	.01	.13	.13	.01
.333	3.536	.00	.00	.00	.00	.00	.38	.37	.37	.37
.286	3.817	.00	.01	.28	.28	.28	.21	.21	.22	.22
.171	4.933	.03	.28	.30	.30	.30	.09	.09	.09	.09
.042	10.020	.96	.71	.08	.08	.08	.05	.05	.05	.05

Table 6.5 reports collinearity diagnostics derived from SPSS, namely eigenvalues of cross-products matrix, condition indices and variance-decomposition proportions for each predictor variable [103]. *Eigenvalue* is a measure of the variance contained in the correlation matrix, such that the sum of eigenvalues is equal to the number of variables of the model; in this case it is equal to 9. Eigenvalue is an indicator of the model's accuracy, as relatively similar values provide evidence that with small variations in the input data the model will be unchanged. The constant value is not considered in the comparison. As shown in Table 6.5, the eigenvalues of the 8 predictor variables are

considered relatively close. This was expected, since the maximum value for the condition index is ≤ 15, which is an evidence of non-collinearity. *Condition indices* are an alternative way of expressing these eigenvalues and represent the square root of the ratio of the largest eigenvalue to the eigenvalue of interest [103]. The *Variance proportions* shown in Table 6.5 show the proportion of variance for each predictor coefficient that is attributed to each conditional index. For a predictor variable with conditional index ≤ 15 and variance proportion > 0.9 there is an indication of an unacceptable level of multi-collinearity. This is not indicated in Table 5.6, since 71% of the variance in the regression coefficient of the variable DATA is associated with condition index equal to 10, while all the other predictor variables the corresponding variance proportion lies in the range of 0.05-0.08.

In this study, collinearity statistics revealed that tolerance was ≥0.6 and VIF was ≤1.5 for all variables, suggesting that multi-collinearity was not an issue [93], as shown in Table 6.4. Finally, the collinearity diagnostics shown in Table 6.5, confirm that the assumption is not violated, as the condition index was found to be ≤10, providing further evidence of non-multi-collinearity [93] as shown in Table 6.5. Table 6.5 has been obtained from SPSS, using the 57600 instances data set.

Although the above preliminary analysis regarding the multiple linear regression assumptions is reasonable, an in depth attempt to provide sufficient evidence that the regression assumptions have been met will be presented after the model fit, mainly based on the residual analysis.

6.3. Regression model

Regression methods control how variables are included into the model. In this study, all variables are entered in the model in one step and therefore the Enter method was selected to investigate the influence of all the predictor variables on the output variable [98]. To test how encryption parameters affect the energy consumption, the null hypothesis testing H_0 is used. First, the null hypothesis is stated by assuming that encryption parameters have no impact on energy consumption. Following this, the test assesses whether evidence obtained from the data does or does not support this hypothesis and rejects or accepts the test accordingly [57]. The results of the multiple linear regression analysis in Table 6.4 indicate that all predictors contribute significantly in the prediction of the energy consumption.

The correlation coefficient R expresses the strength of the linear relationship between variables and is given by [100]

$$R_{xy} = \frac{\sum((x_i - \bar{x})(y_i - \bar{y}))}{\sqrt{\sum(x_i - \bar{x})^2 \sum(y_i - \bar{y})^2}} \tag{6.5}$$

This measure can take values between -1 and 1, with the sign defining the positive or negative relationship. The closest the correlation coefficient to $|1|$, the stronger the relationship, while a correlation coefficient close to zero indicates a weak relationship [65]. According to the model summary shown in Table 6.3 the multiple correlation coefficient R = 0.796 indicates that the observed energies and those predicted by the regression model are strongly correlated.

It is also important to examine the coefficient of determination - R square - which is calculated by squaring the correlation coefficient and measures the proportion of the variance in one variable that is accounted for by another variable [100]. R square is a statistical measure of how close the data is to the fitted regression line. Therefore, in terms of variability in the observed energy consumption accounted by the fitted model, the percentage of the response variable variation that is explained by the model is R^2 = 0.634. This means that 63% of the variation in *ENERGY* is explained by the model.

Table 6.6: ANOVA table

	Sum of Squares	df	Mean Square	F	Sig.
Regression	3188.895	8	398.612	12449.477	.000[b]
Residual	1843.969	57591	.032		
Total	5032.864	57599			

a. Dependent Variable: ENERGY

b. Predictors: (Constant), KEY_SIZE_3, DATA, ISO, OFB, ECB, NO_PADDING, KEY_SIZE_2,

Table 6.6 is the ANOVA (Analysis of Variance) table obtained using SPSS. The ANOVA table reports how well the regression equation fits the data, by analysing the breakdown of variance in the response variable. The total variance is partitioned into the variance that can be explained by the predictor variables used in the regression model and the variance which is not explained by the predictor variables (residuals) [98]. The column *Sum of Squares* describes the variability in the response variable Energy. The sum of squares in Table 6.6 indicates that the vast majority of the total variability is explained by the model. By averaging the sum of squares of regression and residuals by their corresponding number of observations (degrees of freedom - *df*) column, the *mean square* is derived. The ratio of the mean square of the regression and the mean square of the residuals gives the *F-statistic* [98].

The F-statistic tests the hypothesis that the predictors show no relationship to the response variable [98] and therefore that they do not contribute to the model's ability to explain the variance of the response variable. If the p-value is lower than the significance level (0.05), the null hypothesis is rejected. The analysis of variance shown in Table 6.6 reveals that the estimated multiple linear regression is considered statistically significant, F(8,57591) = 12449, $p<0.05$. Table 6.6 has been obtained from SPSS by running the analysis of variance test statistics, using the 57600 data set.

In addition, all the predictors define statistically significantly the energy consumption ($p<0.05$) as shown in the coefficients Table 6.4, indicating that all predictor variables contribute much to the model. The Beta weight measures the number of standard deviations change on the dependent variable that will be caused by a change of one standard deviation on the independent variable under examination [98]. The results of the multiple linear regression analysis suggest that a significant proportion of the total variation in encryption time was predicted by *DATA*, followed by the *KEY*, then *MODE* and finally the *PADDING*.

Based on the outcomes of the analysis above, a regression equation was derived. The dependent variable is denoted by y and the independent variables are $x_1, x_2, \dots, x_n$. When there are correlations between the dependent and independent variables, a multiple regression equation between y and $x_1, x_2, \dots, x_n$ according to [101] can be written as

$$y = a_0 + a_1x_1 + a_2x_2 + \cdots + a_nx_n \tag{6.6}$$

where $a_0, a_1, \dots, a_n$ are the regression coefficients.

A regression equation for the energy consumption assessment as derived from the coefficients in Table 6.4 is:

$$ENERGY = 10^{\mathrm{f}(\mathbf{x})} \tag{6.7}$$

$\mathrm{f}(\mathbf{x}) = \mathbf{a} \cdot \mathbf{x}^{\mathrm{T}}$

$\mathbf{x} = [1 \;\; \log DATA \;\; CFB \;\; ECB \;\; OFB \;\; PKC \;\; ISO \;\; KS2 \;\; KS3]$

$\mathbf{a} = [1.224 \;\; 0.248 \;\; 0.007 \;\; -0.038 \;\; 0.007 \;\; 0.005 \;\; 0.01 \;\; 0.059 \;\; 0.063]$

where vector **a** represents the constant value and the coefficients (B values – Table 6.4) for all the predictor variables used in the model and vector **x** represents the predictor variables.

The regression equation is the sum of all the products of each predictor variable-coefficient pair and the constant value that derived from the regression model.

First, the dependent variable *ENERGY* and the independent variable *DATA* are inversely transformed to return to the original scale as these were prior to the log-transformation. By substituting all independent control variables and their coefficients that resulted from the regression analysis in Equation (6.7), one can predict the energy for any set of encryption parameters.

6.4. Goodness of fit

Goodness of fit of a regression model attempts to evaluate how well the model fits a set of data, or how well it predicts a given set of variables. This section concerns the model evaluation. Residuals and diagnostic statistics are used to evaluate the model and ensure that the assumptions have been met.

Figure 6.5 shows a scatterplot of the standardized residuals to the predicted values, where the residuals lie in the range [-3, 3] according to the 68-95-99.7 rule [95]. The residuals appear to be randomly scattered over and below zero, suggesting that there is no violation of the assumptions presented in the previous section [94].

In addition, the normal probability plot of the residuals in Figure 6.2 shows the points close to a diagonal line; therefore, the residuals appear to be approximately normally distributed [101]. The criterion for normal distribution is the degree to which the plot for the actual values coincides with the diagonal line of expected values, with linearity *0.982*. For this study, the residual plot fits the linear pattern well enough to conclude that the residuals are normally distributed.

The histogram also indicates that it is reasonable to assume that the random errors for these processes are drawn from approximately normal distributions.

Although the histogram and normal P-P plot show that the relationship is not perfectly deterministic, the observation points form a nearly linear pattern, indicating a linear relationship and supporting the condition that the errors are normally distributed.

6.5. Discussion

Although the regression model fits the data that is around the mean well, this is not true for data points that deviate much from the mean. Figure 6.6 depicts the difference between the observed energy with the respective estimated values for 60 random cases of the data. Equation 6.7 has been used to predict the energy consumption for these 60 cases. For each case and the corresponding predictor variables (encryption parameters), equation 6.7 has been applied to estimate the energy consumption for each encryption, by adding the constant value and the coefficients of each encryption parameter of the case.

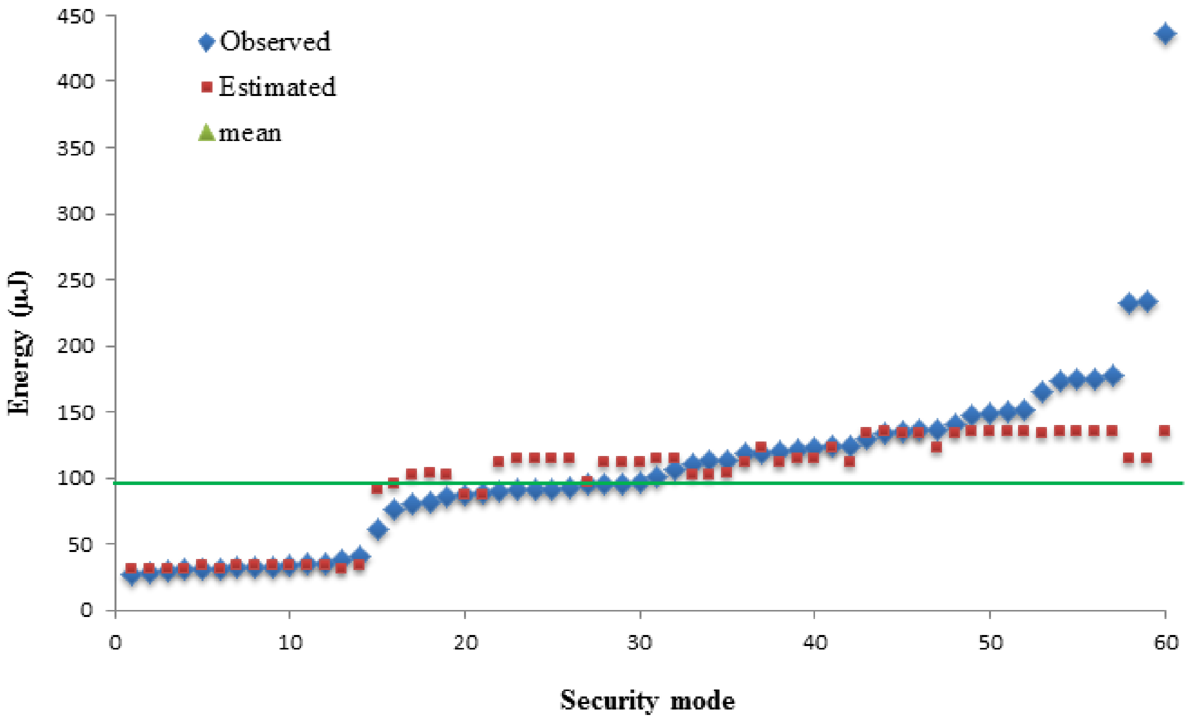

Figure 6.6: Estimated energy vs observed energy.

The predicted values are shown in red in Figure 6.6. The mean energy consumption of the 100 observed values for each one of the 60 cases have been used to compare the predicted and observed values. The observed values are shown in blue. A pointwise comparison is illustrated in Figure 6.6, where the mean observed energy for all 57600 observations is represented by the green horizontal line. The graph shows that for cases when the observed energy lay around the statistical mean, the energy estimated from the regression model is a good fit and the prediction is close enough to the simulated energy consumption. As the observed energy deviates from the statistical mean, the estimated values do not make such good predictions.

The reason is that data points that deviate more than 3 standard deviations from the mean, are considered extremes [102]. Specifically, in the above example the statistical mean is $\mu = 99.9$ and the standard deviation is $\sigma = 33.2$. To check whether a data point is extreme, the range around the mean ± 3 standard deviations $[\mu - 3\sigma,\ \mu + 3\sigma]$ is calculated. In this example, values outside the range [0, 200] are considered extremes. Hence, while the regression model is good fit for values that lay around the statistical mean, this is not true for extreme values [65].

The next chapter, which is the second statistical approach, examines the betweenness centrality of each encryption parameter. It is expected that the results will be in line with the ones obtained from the regression analysis, as both approaches lie in the area of statistical analysis. A comparison between the two statistical approaches is also presented.

Chapter 7 - Betweenness Centrality

In the proposed energy conscious adaptive security framework, energy consumption analysis is performed for the identification and selection of the most efficient security mode. It is critical to select a mode that provides adequate security within the constraint of energy and time requirements. In this chapter, a novel application of betweenness centrality to energy efficient mode selection is presented. By treating the encryption system as a graph, it is possible to identify high impact components in the encryption system by finding indices with high betweenness centrality in the graph. The efficiency of these high impact components is analyzed in order to conclude the optimal security mode.

Assessing the significance of encryption parameters against reliability of successful encryption in terms of energy consumption is a challenging task. Graph metrics, such as betweenness centrality, can give an indication of significant encryption parameters. The objective of this chapter is to extend the reliability framework presented earlier in this thesis, where reliability was employed to deduce a global quality factor for optimal security mode selection, with respect to energy consumption. An important operational concept for this analysis is centrality. Therefore, matching reliability with betweenness centrality, the subsequent approach incorporates the betweenness centrality into the reliability model. The results suggest that a centrality approach represents an alternative and meaningful direction to the study of the most efficient security mode selection.

7.1. Betweenness centrality of components

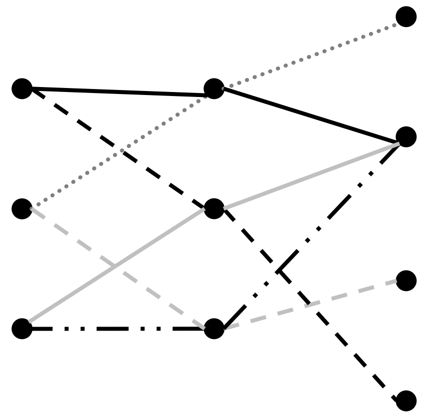

Figure 7.1: Possible paths in a graph.

By representing an encryption system as a graph, where the nodes depict the encryption components and the edges represent the flow of the encryption procedure, it will appear as a set of interconnected nodes, where the shortest paths in terms of lowest encryption time can be recovered. Figure 7.1 shows the type of different paths that can be detected in the graph.

In this approach, the observation that the components of the encryption system tend to associate with other components through a variety of paths, which often form the shortest possible paths, is exploited. The objective is to evaluate the significance of each component in the system. Betweenness centrality is the method that measures the significance of a component in the graph and therefore concludes the importance of the component in the system.

The betweenness centrality of the encryption components in this approach is examined using the R platform. As mentioned in Chapter 2, the betweenness centrality of a vertex is defined by the number of geodesies *g* going through this vertex *v* [69]. Therefore from (2.21), the betweenness of the encryption component represented by vertex *v* is calculated by

$sum(g_{s \rightarrow v \rightarrow t} / g_{s \rightarrow t}),\ s!=t, s!=v, t!=v$

where $g_{s \rightarrow v \rightarrow t}$ denotes the number of shortest paths between *s* and *t* passing through *v* and $g_{s \rightarrow t}$, the total number of shortest paths between *s* and *t*. The result in R is a numeric vector with the betweenness score for each vertex.

In the subsequent experiment, shortest paths are considered for only the cases that for a desired reliability level will finish the encryption procedure prior to the time threshold. Thus, the time threshold and reliability level will be the initial parameters that have to be set by the user. In the specific case of encryption, the total number of shortest paths is translated into the number of security modes that meet the reliability and time requirements, while the shortest paths that pass through *v* are considered the cases that encrypt using the parameter – vertex under investigation i.e. ECB, PaddingPKCS5 . The concept of computing the betweenness of the vertices is the following. First, the security requirements are set, i.e. the AES standard has to be used for the encryption. Following on, all shortest paths between vertices are identified. For each vertex *v*, the shortest paths that pass through it are counted. Finally, the fraction of shortest paths that pass through *v* and the total number of shortest paths is calculated, to conclude the betweenness centrality of the vertex.

The algorithm for the betweenness centrality implementation is shown in Figure 7.2.

```
Variables:  list Cases      # 576 cases data set
            list path       # list of vertices for a case
            list Paths      # list of paths
            float R         # reliability
            float B         # betweenness centrality

for case in Cases
    Calculate R
    if $R \neq R_{req}$
        delete case
    else
        add path to Paths
    end if
end for

for path in Paths
    for v in path
        Calculate B
    end for
end for
```

Figure 7.2: Betweenness centrality algorithm.

Figure 7.3 illustrates all possible paths of the encryption flow among encryption parameters. In this graphical representation of the encryption system, the encryption parameters are arranged into distinct layers of parameters. For example, the key size layer contains the available key sizes, i.e. 128, 192, 256 and is followed by the padding layer, which is composed by the NoPadding, the ISO and PKCS5 node, and in turn is followed by the mode of operation layer. This consists of the CBC, CFB, ECB, OFB modes. In this multilayer graph, a path can only traverse one node of each layer, i.e. there is no way to encrypt using two padding schemes, and a node can only have a directed connection to another node from the immediately subsequent layer.

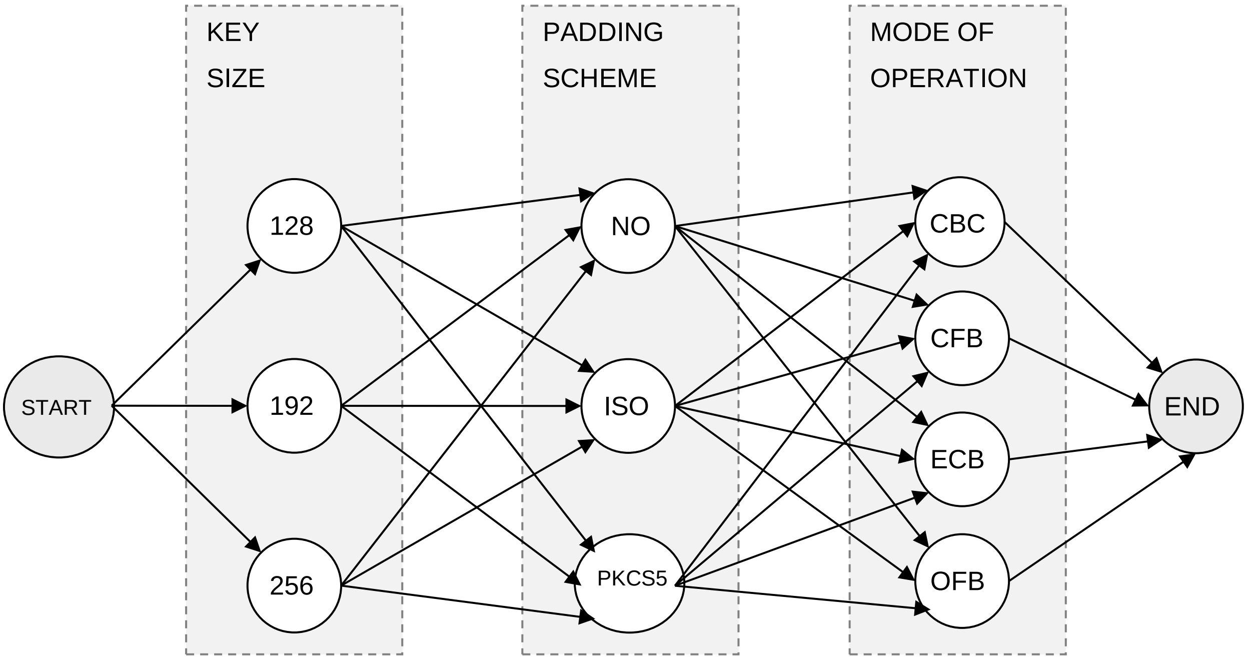

Figure 7.3: Encryption parameters' connectivity.

Therefore, the direction of the path cannot forward to a node that belongs to a precedent layer. In addition, in this chain-like graph, the two end points have no betweenness, as there are no other paths that pass through these points, apart from the paths that start from there. For this reason, the graph is formed as a sequence of vertices and edges, beginning and ending with the start and end node, respectively.

The next section is based on an example case study that examines the betweenness centrality of the encryption parameters, and discusses the results that derived from the experiment.

7.2. Implementation and results

The experiment was conducted based on the reliability concept that was presented in Chapter 3, using the initial dataset as resulted from the simulation. Betweenness centrality has been calculated and the relevant results were examined in terms of the effect of each encryption parameter on the final decision making for the optimal security mode selection. In addition, the variation of the reliability criterion was used to establish the significance of each parameter, and thus its impact on the overall behaviour of the system.

The following case study examines the centrality of the key size, padding scheme and mode of operation, based on a time threshold t = 300000 ns, for a requested encryption of 2048 byte data, using the AES encryption algorithm. The reliability level is varied between 0.4 and 0.1.

First, the reliability level is set equal to 40%, and the shortest paths are identified. Following on, the betweenness centrality is calculated for each node as follows. Let σ_{SE} be the total number of shortest paths from the start node *S* to the end node *E*, and $\sigma_{SE}(128)$ the number of shortest paths from *S* to *E* that pass through 128 bit key size. Then, from Equation (2.20),

$$C(128) = \frac{\sigma_{SE}(128)}{\sigma_{SE}} = 0.8$$

The rest of the results are shown in Table 7.1. Figure 7.4 shows the shortest paths of the system for the specified requirements. The yellow-shaded nodes represent the most central parameters of the encryption flow. The graph reveals the significance of the 128 key size node, as most of the shortest paths traverse this node. Thus, it is anticipated that the betweenness centrality of this node will be high. According to the results in Table 7.1, this expectation proved to be true, as node 128 showed the highest centrality among all components in the graph, followed by nodes PKCS5 and CFB.

Table 7.1: Betweenness centrality: R = 0.4.

Parameter	Betweenness
128	0.8
192	0
256	0.2
No Padding	0.2
ISO	0.2
PKCS5	0.6
CBC	0.2
CFB	0.4
ECB	0.2
OFB	0.2

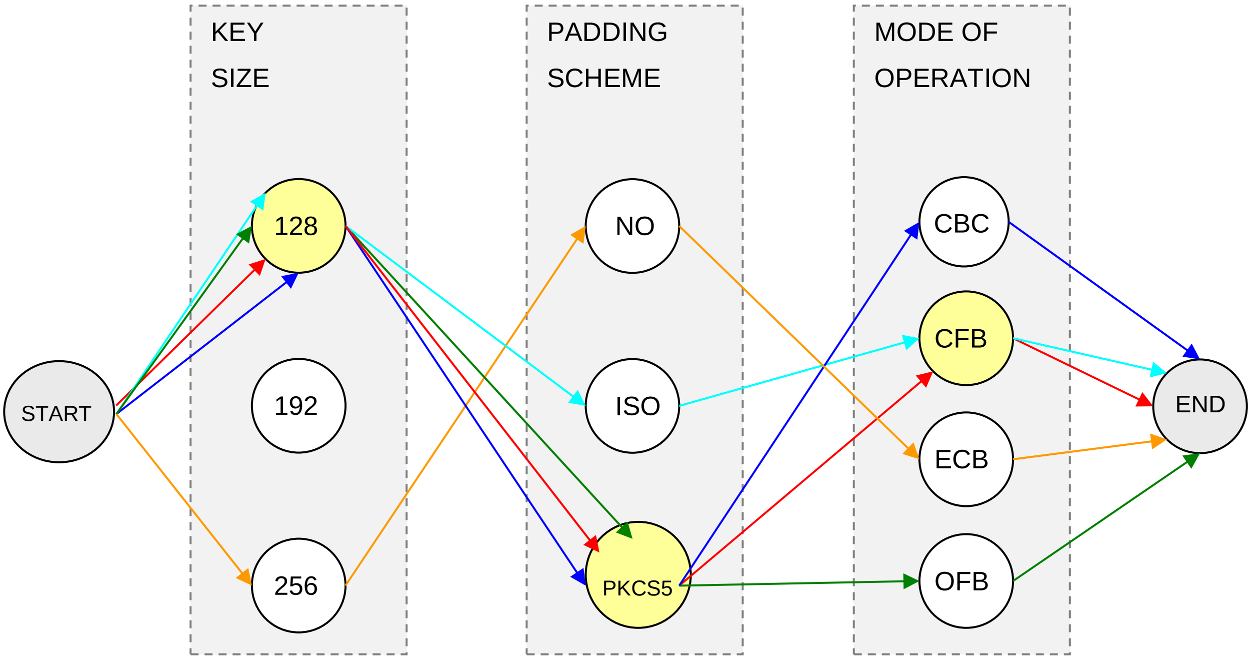

Figure 7.4: Shortest paths and most central nodes: R = 0.4.

On a parameter basis, the results show that, in terms of the key size, the 128 bit key has a significantly high centrality of $BC_{128} = 0.8$, compared to $BC_{192} = 0$ and $BC_{256} = 0.2$. Similarly, PKCS5 with $BC_{PKCS5} = 0.6$ is the most significant padding scheme compared to ISO and no padding that resulted in $BC_{NO} = BC_{ISO} = 0.2$. Finally, CFB with $BC_{CFB} = 0.4$ is the most frequent mode of operation to be present in the shortest paths of this service, followed by $BC_{CBC} = BC_{ECB} = BC_{OFB} = 0.2$.

As shown in Figure 7.4, although key size 256 is the key size that normally consumes the highest energy, it is included in one of the shortest paths. This can be explained by the fact that the reliability for this scenario is fixed and in the specific case it is quite high (40%).Therefore, even though other key sizes could be used instead, that consume less energy, these have been excluded from the shortest path list, as they do not meet the reliability requirement that has been set by the user. Although 128 bit key size appears in most of the shortest paths, it is used in combination with high consuming parameters i.e. CFB mode of operation.

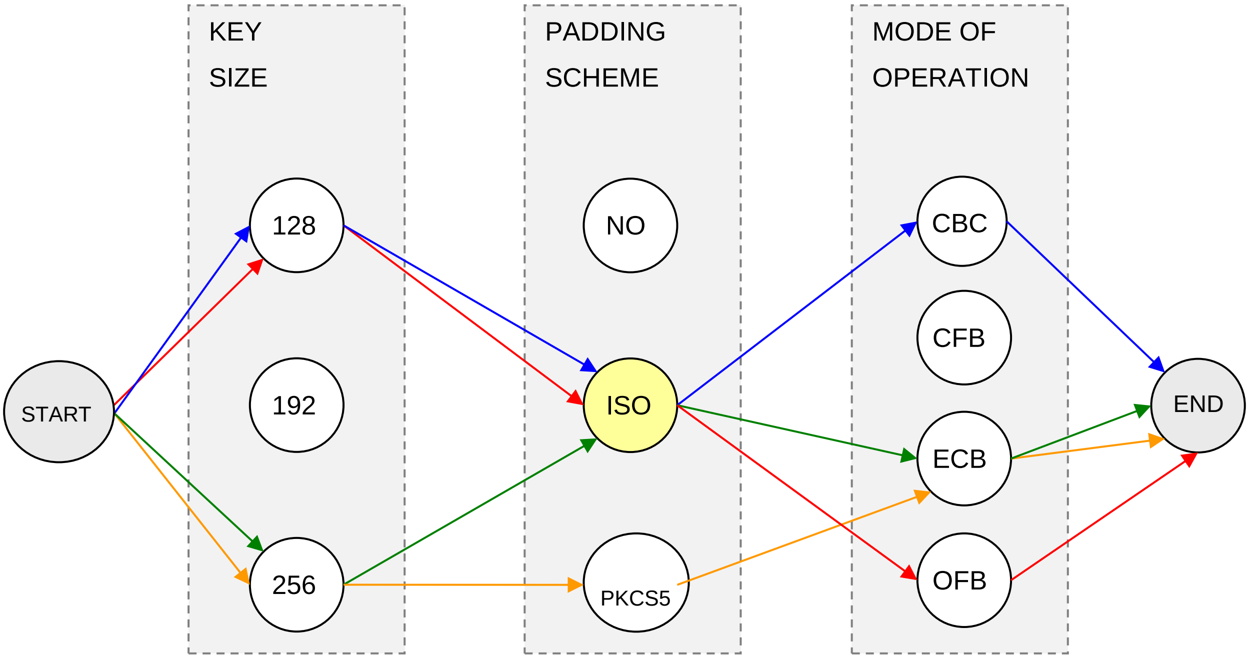

Figure 7.5: Shortest paths and most central nodes: R = 0.3.

Following, the reliability level is set equal to 30%. Figure 7.5 depicts the shortest paths of this service, where the ISO padding scheme appears to be the most significant node of the graph. Key size 128 and 256, as well as ECB mode of operation are traversed by shortest paths in 50% of the cases. The nodes that are not present in any of the shortest paths, and therefore are not connected to any other node, have zero centrality. As in the previous scenario, key size 256 has 50% betweenness centrality, which means that 50% of the shortest paths pass through this node. For the same reason that was explained in the previous scenario and because reliability is relatively high (30%), the shortest paths that consume less energy have been eliminated due to the reliability requirement.

As shown in Table 7.2, key size 128 bit and key size 256 share an equal significance of $BC_{128} = BC_{256} = 0.5$, while key size 192 bit with $BC_{192} = 0$ is not present in any of the shortest paths of this service. ISO padding was present in 75% of the shortest paths with $BC_{ISO} = 0.75$, while PKCS5 was involved in 25% of the cases with $BC_{PKCS5} = 0.25$ and NoPadding in none of the shortest paths with $BC_{NO} = 0$. Regarding the modes of operation, ECB has a relatively high centrality $BC_{ECB} = 0.5$ compared to CBC and OFB with $BC_{CBC} = BC_{OFB} = 0.25$ and CFB with $BC_{CFB} = 0$.

Table 7.2: Betweenness centrality: R = 0.3.

Parameter	Betweenness
128	0.5
192	0
256	0.5
No Padding	0
ISO	0.75
PKCS5	0.25
CBC	0.25
CFB	0
ECB	0.5
OFB	0.25

In the next scenario, the reliability level is decreased to 20%. Figure 7.6 illustrates the connectivity of the nodes that are involved in the shortest paths of the encryption. 128 bit key size, No padding and ECB mode of operation are the most central nodes. The results are summarized in Table 7.3.

The key size with the highest centrality was found to be the 128 bit key with BC_{128} = 0.57, followed by key size 192 with BC_{192} = 0.43. Key size 256 is not involved in the shortest paths and therefore BC_{256} = 0. NoPadding showed a relatively high centrality of BC_{NO} = 0.57, followed by PKCS5 with BC_{PKSC} = 0.28 and BC_{ISO} = 0.14. Finally, ECB was the most significant mode of operation with BC_{ECB} = 0.57, followed by the equal centralities of the rest modes with BC_{CBC} = BC_{CFB} = BC_{OFB} = 0.14.

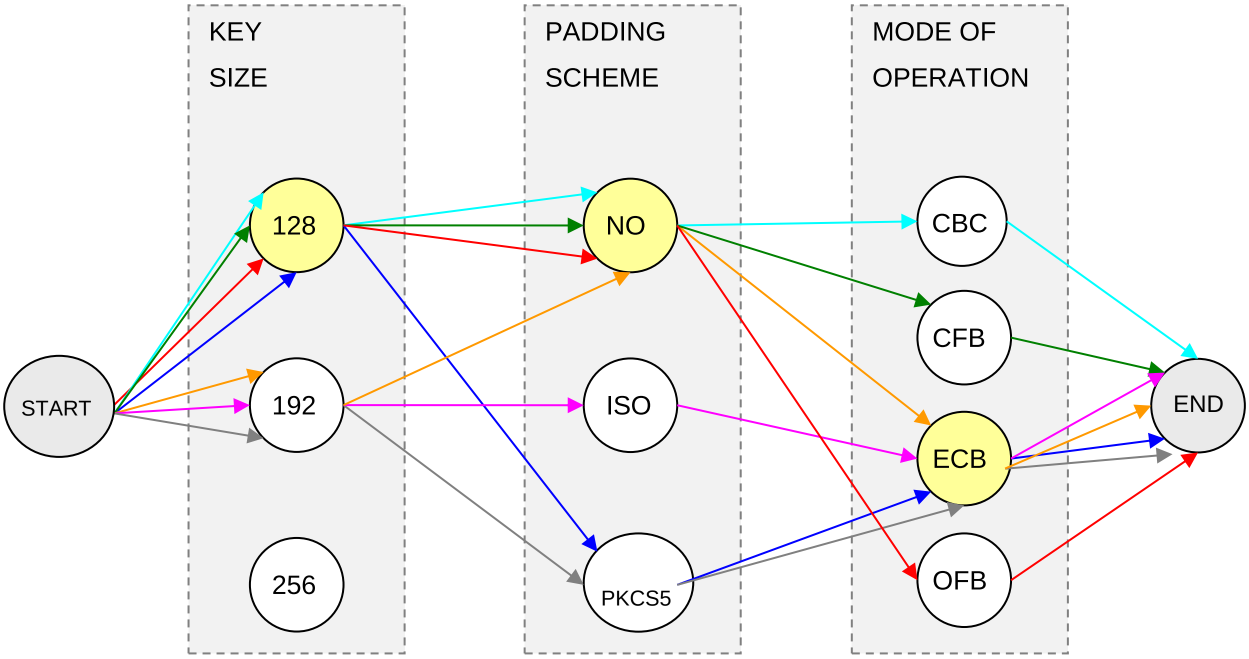

Figure 7.6: Shortest paths and most central nodes: R = 0.2.

Table 7.3: Betweenness centrality: R = 0.2.

Parameter	Betweenness
128	0.57
192	0.43
256	0
No Padding	0.57
ISO	0.14
PKCS5	0.28
CBC	0.14
CFB	0.14
ECB	0.57
OFB	0.14

Finally, the test is repeated for a desired reliability level of 10%. As shown in Figure 7.7, key size 128 and ECB mode of operation are traversed by all shortest paths. The

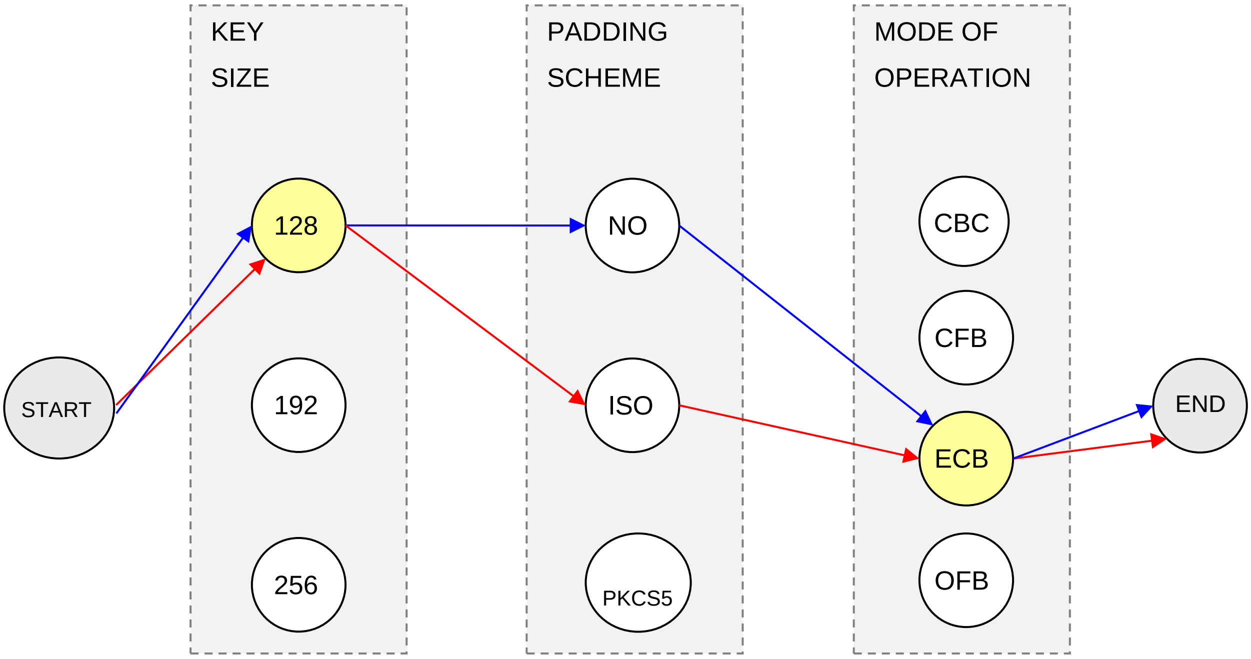

Figure 7.7: Shortest paths and most central nodes: R = 0.1.

Table 7.4: Betweenness centrality for R = 0.1.

Parameter	Betweenness
128	1
192	0
256	0
No Padding	0.5
ISO	0.5
PKCS5	0
CBC	0
CFB	0
ECB	1
OFB	0

rest key sizes and modes of operations, as well as PKCS5 padding, have no significance in the graph, as there is no shortest path that passes through these nodes.

In the last variation of the reliability level, it was found that key size 128 was involved in all shortest paths with $BC_{128} = 1$, while none of the rest keys was present in a shortest path, $BC_{192} = BC_{256} = 0$. NoPadding and ISO share a centrality of $BC_{NO} = BC_{ISO} = 0.5$, meaning that PKCS5 was not involved in any shortest path with $BC_{PKCS5} = 0$. ECB with $BC_{ECB} = 1$ revealed its presence in all shortest paths for this service and that the rest of the modes were not significant in the graph, as $BC_{CBC} = BC_{CFB} = BC_{OFB} = 0$. Table 7.4 depicts the results for 10% reliability.

7.3. Discussion

To conclude, Table 7.5 depicts the most central encryption components of the case study.

Table 7.5: Most significant parameters

Reliability	Key size	Padding scheme	Mode of operation
0.4	128	PKCS5	CFB
0.3	128, 256	ISO	ECB
0.2	128	NO	ECB
0.1	128	NO, ISO	ECB

The results suggest that key size 128 is always present in the shortest paths of this example, with a centrality varying between 50% and 100%. Similarly, the centrality results regarding the modes of operation indicate that ECB traverses the shortest paths 75% of the cases. Specifically, for a reliability level in the interval of 10% and 30%, ECB is the most central mode of operation with centrality varying between 50% and 100%. In addition, the results indicate that there is a negative relationship between ECB and reliability, as the centrality of ECB increases as the reliability level decreases, meaning that the higher the probability that the encryption will have finish prior to the threshold, the stronger the betweenness of ECB will be. Unlike key size and mode of operation, there is no evidence that centrality can be used as an indicator of the most significant padding scheme for this example.

Figure 7.8 shows the relationship between the reliability level and the betweenness centrality metric for the encryption parameters. The observations of the results reveal that the impact of the energy consumption is in line with the conclusions made in Chapter 6. Specifically, in the regression model analysis, it was found that the key size is the most significant parameter, followed by the mode of operation, while the padding scheme has the lowest impact on the energy consumption of the encryption procedure.

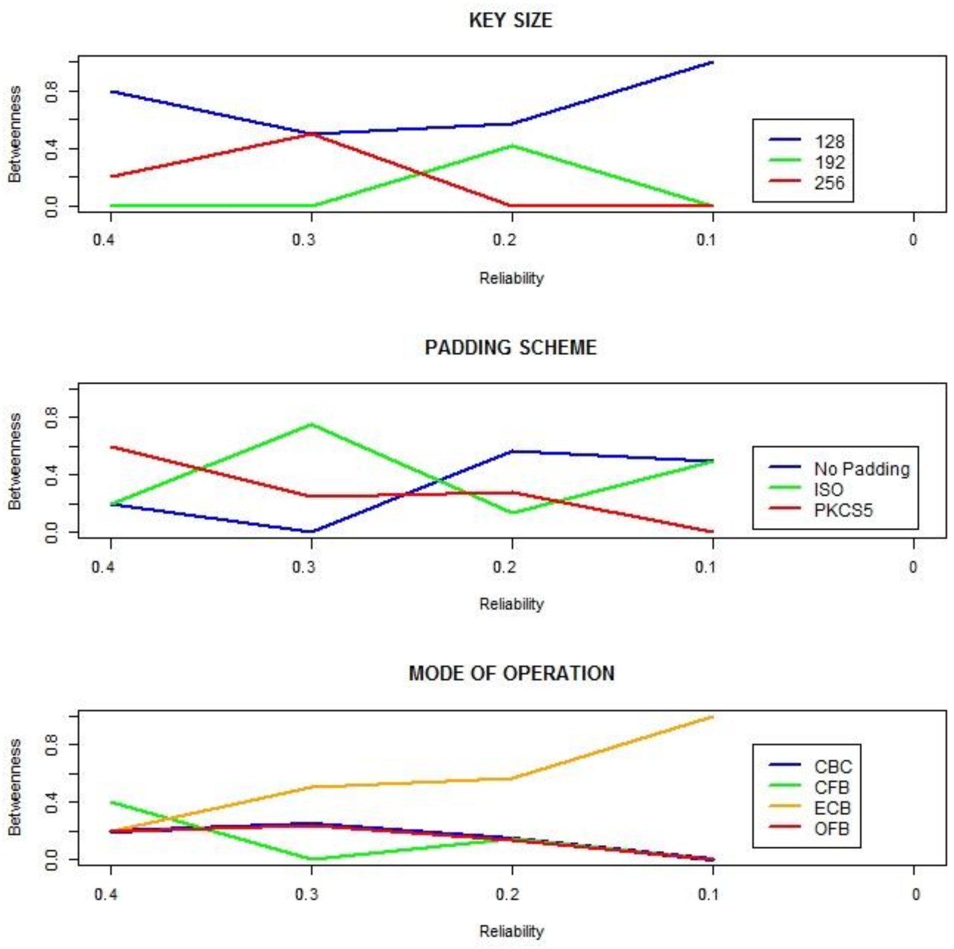

Figure 7.8: Betweenness vs Reliability.

Table 7.6 summarizes the impact of the predictor variable (encryption parameters) on the response variable (energy), as resulted from the regression analysis. As mentioned in Chapter 6, the higher the coefficient of a predictor variable, the highest the impact of the encryption parameter on the energy consumption and therefore the more energy is consumed during encryption. Therefore, for a specific encryption parameter, when the rest of the predictor variables are fixed, by applying the regression equation to predict the energy consumption, and by adjusting the encryption parameter of interest, based on the coefficient, the model will result in different values of energy.

Table 7.6: Regression coefficients

Encryption parameter	**Regression coefficient**
Key size 1	0
Key size 2	0.059
Key size 3	0.063
No Padding	0
ISO	0.01
PKCS5	0.005
CBC	0
CFB	0.007
ECB	-0.038
OFB	0.007

As everything apart from the variable of interest is fixed in the regression equation, the energy will vary based on the coefficients of the specific variable. Therefore, the higher the coefficient, the higher will be the value of the estimated energy. Thus, the higher the coefficient, the highest the impact of the corresponding variable on the energy consumption will be. As shown in Table 7.6, Key size 1 (128) has the lowest impact on the energy consumption, whilst Key size 3 (256) has the highest one. Regarding the Padding scheme, NoPadding is the most efficient one, followed by PKCS5, whilst ISO padding has the highest impact on the response variable and therefore consumes the highest energy during encryption. Finally, ECB mode of operation has the lowest impact on the energy consumption, followed by CBC, whilst OFB and CFB have the highest impact on the energy consumed during encryption. Table 7.5 summarises the most significant parameters; the ones with the highest betweenness centralities. As mentioned earlier, the higher the betwenness centrality, the lowest the impact on energy consumption. From Table 7.5, it is shown that key size 1 (128) is the one with the lowest impact on energy consumption. Similarly, for the mode of operation, ECB is the mode with the lowest impact on energy in most of the cases. Regarding the padding scheme, NoPadding and ISO are the ones that appear in the paths with the lowest energy consumption in most of the cases. Taking into consideration the centralities as resulted from the betweenness analysis for R = 0.1, as for higher reliability there is uncertainty in terms of completion, there is a consistency between the betweenness centrality results and those resulted from the regression analysis. Specifically, according to the centrality results, key size 128 appears to be the most central node in the shortest paths, which is in line with the regression model as key size 1 (128) has the lowest impact on the energy consumption. Key sizes 192 and 256 resulted in 0 centrality, conforming to the regression coefficients of 0.059 and 0.063 respectively. Similarly, No Padding is one of the most central nodes in the shortest paths for low reliability levels, which is in agreement with the zero impact of the padding scheme on the energy consumption as per the regression model. Finally, regarding the mode of operation, ECB is by far the node with the highest centrality, conforming to the regression results where ECB has the lowest impact on the energy consumption with a coefficient equal to -0.038.

In this chapter, the problem of low energy encryption was studied to identify the most efficient security mode. The encryption flow was modelled using ideas from graph theory, and a novel approach was developed, based on the betweenness centrality measure. The goal was to transform the impact of each encryption parameter on the probability that the system will finish the encryption prior to a threshold into a knowledge asset that can be used for the decision making, regarding the selection of the most efficient security mode. The results demonstrated that betweenness centrality is a useful measure that facilitates the study of the security mode selection taking into consideration the energy consumption. Even though in some cases the results might seem anomalous, considering that reliability is fixed and also because of the combination of the high energy consuming parameters with the low consuming ones, this concern can be justified.

Chapter 8 - Conclusions and future work

Over the preceding chapters it has been shown that the use of the proposed adaptive security framework when energy efficient encryption is necessary, may be one possible beneficial method.

The motivation for the development of this framework including all variations presented through different approaches was the identification of a gap in the area of energy efficient encryption systems. This framework addresses the global impact of the encryption parameters on energy levels, which has not previously been addressed by traditional mechanisms. This thesis has not aimed at focusing on specific algorithms or security threats, but has rather presented an approach for the selection of the most efficient security mode. The framework presented in this work enables a novel decision making procedure, using several stochastic and statistical methods, as well as ideas from graph theory.

In this chapter, the thesis contributions are summarized and the prospective directions of future work are introduced.

Chapter 1 presented the motivation of this work. The problem identification was outlined and the rationalization for addressing the energy implication on encryption was introduced. Chapter 2 introduced notions of energy and security, as well as mathematical methods that have been used for the development of the proposed framework and are further analyzed in the following chapters. Existing efforts in the area of low energy encryption were also highlighted and limitations in terms of universal deployment and adaptability were discussed.

In Chapter 3, the concept of how a global metric for energy performance evaluation allows for optimal security mode selection with respect to energy consumption was presented, followed by an implementation overview where the simulation and data analysis techniques are discussed. In total five ciphers were simulated and the variation of four encryption parameters resulting in 576 possible security modes. These modes are further analyzed in terms of performance in the subsequent chapters.

Chapter 4 provided a detailed analysis of the reliability model. A framework was developed for the determination of the encryption performance for all security modes. This allowed for relationships to be formed based upon the probability that the encryption will be completed prior to a time or energy threshold. The model was also optimized to provide the optimum threshold that is essential when the security mode has to be adjusted according to the severity of the requested service. The results showed how the threshold can affect the reliability metric and vice versa. Finally, the analysis of the derived asymptotic distributions of distinct, as well as compound encryption policies, provided a probabilistic framework for predicting, ranking and comparing the energy consumption induced by different policies, allowing for customization according to the security requirements. Overall, the benefits of deploying the reliability model for the decision making for the selection of the most effective security mode were highlighted. As shown in Chapter 4, the investigation of the probability that a single encryption will be completed prior to a given threshold is feasible using the ECDF derived from the simulations. This acts as a first indicator of performance for the corresponding security mode. Naturally, the question that arises concerns the performance evaluation of n encryptions i.e. What happens when n cases need to be encrypted using a specific security mode? What is the probability that the overall encryption time will not exceed a threshold specified by the user? In Chapter 4, a Normal approximation has been presented under the mathematical assumption that $\boldsymbol{n} \to \infty$. To answer the question above, knowledge about the distribution is needed. By applying CLT, the distribution was found to be Normal. The flexibility of CLT is that it requires independent random variables with finite variance and $\boldsymbol{n} \to \infty$. This allowed for the investigation of the compound encryption scenario. In section 4.5 the compound scenario has been presented, that can be used as a decision making policy for the allocation of k encryptions to a specific security mode and *n*-*k* to another one. This policy could be further extended and include more security modes. The distribution of the overall encryption time of the *n* encryptions would be Normal and the corresponding $\boldsymbol{\mu}$ and $\boldsymbol{\sigma^2}$ values could be used to derive the compound mode distribution from Equation (4.15).

CLT is very useful and it is widely used for decision making and inference on the sum or mean of *n* statistics. However, the assumption that $\boldsymbol{n} \to \infty$ sometimes is not fully stated or it is assumed that as *n* is large, it approaches to infinity. Most of the results in probability theory are asymptotic, i.e. assumption that $\boldsymbol{n} \to \infty$. Since this work considers a real world application (the overall encryption time or the mean encryption time of *n* encryptions), it is considered that this asymptotic assumption is not realistic, as *n* is finite (on a scale of hundreds or thousands of encryptions). In order to relax this assumption and maintain the inference in a mathematically rigorous way, Chernoff type bounds can be used, as presented in Chapter 5. In Chapter 5, a second probabilistic approach was presented, that bounds the tail of the distribution of both the overall and the mean execution time of *n* encryptions. Several inequalities have been presented, for both upper and two-sided bounds, resulting in a framework that allows for optimal policies to be identified, under certain user constraints. The advantage of this framework is that is mathematically rigorous and also flexible, since knowledge of distribution is not required. Finally, this approach relaxes the CLT's assumption made in Chapter 4, as a number of executions *n* can be considered finite. Thus, the trade-off between the number of encryptions, level of accuracy and sharpness of predictions can be achieved.

In Chapter 6 a statistical analysis was presented that identified the impact of the encryption parameters on the overall energy consumption and that can be used to predict the energy consumption for certain encryption parameter values. The results showed that the regression model fits well for cases that the energy consumption levels lay around the statistical mean, but also showed some possible drawbacks for extreme cases. The validity of the regression model has been confirmed by assessing the principal assumptions – linearity, normality, independence, homoscedasticity and non multi-collinearity – about the predictor variables, the response variable and their relationship. R^2, which is an indicator for goodness of fit, was found to be 0.634. This means that 63% of the variation in energy is explained by the model and indicates a relatively good fit. A complicated and risky situation (interpolation) may exist when prediction outside the range of the data set that has been used to derive the regression model is attempted. In this case, the model would be statistically unreliable and the results would not be trustworthy. However, in this work, this situation has been taken into consideration, as no attempt has been made to make predictions using predictor values outside the range of the data set that has been used for the regression model. In addition, as discussed in section 6.2, small variations in the input data would not affect the model, as there is no multi-collinearity. Therefore, variations of the predictors would not violate the statistical accuracy of the model, as long as the risk of interpolation is taken into consideration i.e. predictors' values vary between the minimum and maximum observed values. As shown in Chapter 6, regression analysis can also provide useful information about the influence of each encryption parameter on energy. Overall, the regression model provided reasonable results, which were in line with the betweenness centrality ones, presented in Chapter 7.

Betweeness centrality, presented in Chapter 7, was used to measure the significance of each encryption parameter within the encryption process. In this approach, the encryption system was treated as a graph and the components with the highest impact on the energy consumption were identified. It was found that the key size showed the highest betweenness centrality, followed by the mode of operation, while the padding scheme had the lowest impact on the energy consumption. It was observed that the results were in line with the statistical analysis in Chapter 6. One of the drawbacks of this approach is that in order to make a reasonable comparison of the security modes, the reliability requirement had to be fixed; otherwise it would not be reasonable to compare different levels of reliability i.e. < 0.3 because in that case, many high energy consuming nodes would appear in the shortest paths, as they would meet the reliability requirement and therefore the centrality comparison would not be fair. Thus, for a fixed reliability, when the latter is high, in some cases high energy consuming nodes will be included in the resulting shortest paths. However, these are usually combined with low energy consuming parameters; therefore the results are sensible.

This study found that the adaptability of an encryption system can indeed be based on the energy consumption of the encryption algorithms. By adjusting the encryption parameters and therefore the level of security, unnecessary energy consumption can be avoided. Therefore, a system that adapts to the specified security requirements can be used to achieve the minimum energy consumption possible. The combination of the statistical and stochastic methods can facilitate the development of the energy conscious adaptive security scheme, as it allows for a thorough investigation of the behaviour of the encryption algorithms and the corresponding parameters. In addition, depending on the needs of the system, the analysis can be made based on the specific purpose. For example, when the scheme is aimed for low risk applications, regression analysis might provide adequate information for the development of the system. However, for cases where extreme scenarios need to be considered, probabilistic bounds should be used in order to provide more meaningful results that should be taken into consideration for the implementation of the scheme. The combination of the techniques presented in this work could therefore provide an integrated solution. This would allow for effective decision making in terms of the most efficient selection of the encryption algorithm and the encryption parameters. A formal decision making framework that incorporates statistical and stochastic approaches and provides improvement strategies (such as bounds) and verification methods (such as regression analysis and betweenness centrality) can therefore be applied to energy conscious adaptive security schemes regardless of the available algorithms and primitives.

Overall, the original aims of this thesis have been fulfilled. It has been demonstrated that the proposed techniques can be used to optimise the decision making with respect to the most efficient security mode selection. The proposed methods are not tied to any specific security primitives and thus provide a flexible solution that may find application in any type of communication system. Therefore, the proposed methods could be considered as a security framework for encryption systems where the energy cost is of prime importance.

The ideas presented in this work can be further developed in several ways. One direction of future work is to test and evaluate the proposed framework in a real world system. It would be reasonable to perform a comparison of simulation methods and real world encryption systems. The aim would be to compare encryption time and energy consumption for identical models on real and simulated data. Specifically, the area of WSNs is currently investigated so that a real world application of the proposed energy conscious adaptive security scheme will be realised. The work, which is at its initial stage, is currently attempted using real equipment and it is expected that it will successfully provide useful results.

In addition, regarding the regression analysis, although the generalisation of algorithms and modes was one of the intentions of this work, it would be interesting to investigate further the behaviour of the algorithms and the encryption parameters' variation for specific standards. It would be interesting to fit a regression model based on a specific algorithm and compare the results (coefficients, R^2, response variable) to the generalised ones. This could lead to more accurate predictions and a clearer view of the impact of the encryption parameters on energy.

Another possibility of future work would be to include more encryption algorithms in the system. The resulting scheme would allow for the optimum selection among a plethora of options and hence provide a more general purpose decision making tool. Moreover, examining algorithms other than those used in this work, would provide an interesting point of comparison.

Furthermore, although in the presented approach security is considered as a requirement set by the user or the system administrator, this could be further extended so that the system can automatically rank the available security modes based on their strength. This could be achieved by assigning a security metric to each security mode i.e. by performing several attacks and evaluating the algorithms' resistance or the probability of being compromised within a certain time threshold. Although this is a rather challenging task, as it is not easy to consider all possible attacks, such a system would be very useful, especially for cases where the main concern is the security strength, rather than the energy consumption.

Although it is beyond the scope of this thesis, another direction of work would be to investigate when the encryption scheme is considered adequate, in terms of security, by the user. Allowing a probability that a single encryption may be intercepted, the aim would be to find the probability that a stream of *n* encryptions can also be intercepted. Modern security schemes are designed to ensure secure communication between two parties, even for cases where the attacker has information about the encrypted message. The investigation of the entropy could possibly contribute to the optimisation of the policy that will eliminate the attacker's knowledge to a desired threshold of information. It is expected that the framework proposed in this thesis could be adapted to this direction. To this end, future research plans include the investigation of the entropy and its adaptation to the energy conscious adaptive security proposed in this thesis.

The investigation of the global impact of the encryption parameters on energy levels offers the potential for significant steps forward in the area of green cryptography. The concept of reusing ideas from existing algorithms and primitives in the development of new security protocols or algorithms could be deployed. Examining the parameters' variation in terms of energy consumption could be used to identify emerging areas of low energy encryption, through the proposed methods, to identify potential new low energy security protocols.

References

[1] D. S. A. Elminaam, H. M. A. Kader, and M. M. Hadhoud, "Energy Efficiency of Encryption Schemes for Wireless Devices," *International Journal of Computer theory and Engineering,* vol. 1, pp. 302-309, 2009.

[2] R. Chandramouli, S. Bapatla, K. Subbalakshmi, and R. Uma, "Battery power-aware encryption," *ACM Transactions on Information and System Security (TISSEC),* vol. 9, pp. 162-180, 2006.

[3] N. Jorstad and T. Landgrave, "Cryptographic algorithm metrics," in *20th National Information Systems Security Conference*, 1997.

[4] C. Taramonli, R. J. Green, and M. S. Leeson, "Energy conscious adaptive security scheme for optical wireless," in *14th International Conference on Transparent Optical Networks (ICTON)*, 2012, pp. 1-4.

[5] D. Shackleford, "Regulations and Standards: Where Encryption Applies," *Best Practices for Data Privacy Compliance, SANS Institute Reading Room,* November 2007 2007.

[6] N. R. Potlapally, S. Ravi, A. Raghunathan, and N. K. Jha, "A study of the energy consumption characteristics of cryptographic algorithms and security protocols," *IEEE Transactions on Mobile Computing,* vol. 5, pp. 128-143, 2006.

[7] A. Simmonds, P. Sandilands, and L. Ekert, "An Ontology for Network Security Attacks," in *Applied Computing*. vol. 3285, S. Manandhar, J. Austin, U. Desai, Y. Oyanagi, and A. Talukder, Eds., ed: Springer Berlin Heidelberg, 2004, pp. 317-323.

[8] N. Sklavos and X. Zhang, *Wireless Security and Cryptography: Specifications and Implementations*: CRC Press, Inc., 2007.

[9] L. Trevisan, "Cryptography Lecture Notes," in *Computer Science*, S. University, Ed., ed, 2009.

[10] H. Imai, M. G. Rahman, and K. Kobara, *Wireless communications security*: Artech House, 2006.

[11] C. T. Hager, "Context aware and adaptive security for wireless networks," Virginia Polytechnic Institute and State University, 2004.

[12] A. J. Menezes, P. C. Van Oorschot, and S. A. Vanstone, *Handbook of applied cryptography*: CRC press, 2010.

[13] S. Seys, "Cryptographic Algorithms and Protocols for Security and Privacy in Wireless Ad Hoc Networks," PhD, Engineering, Katholieke Universiteit Leuven, Leuven-Heverlee, 2006.

[14] B. Schneier, "Applied cryptography: protocols, algorithms, and source code in C," ed: John Wiley, 1996.

[15] M. Dworkin, "Recommendation for block cipher modes of operation. methods and techniques," DTIC Document2001.

[16] IBM, "Data Encryption Standard," *Federal Information Processing Standards Publication,* 1977.

[17] M. Simpson, K. Backman, and J. Corley, *Hands-On Ethical Hacking and Network Defense*: Cengage Learning, 2010.

[18] J. Daemen and V. Rijmen, "AES Proposal: Rijndael," 1998.

[19] J. Daemen and V. Rijmen, "Advanced encryption standard," *Federal Information Processing Standard, FIPS-197,* p. 12, 2001.

[20] L. R. Knudsen, V. Rijmen, R. L. Rivest, and M. J. Robshaw, "On the design and security of RC2," in *Fast Software Encryption*, 1998, pp. 206-221.

[21] C. T. Hager, S. F. Midkiff, J.-M. Park, and T. L. Martin, "Performance and energy efficiency of block ciphers in personal digital assistants," in *Pervasive Computing and Communications, 2005. PerCom 2005. Third IEEE International Conference on*, 2005, pp. 127-136.

[22] P. Prasithsangaree and P. Krishnamurthy, "On a framework for energy-efficient security protocols in wireless networks," *Computer Communications,* vol. 27, pp. 1716-1729, 2004.

[23] J. Rittinghouse and W. M. Hancock, *Cybersecurity operations handbook*: Access Online via Elsevier, 2003.

[24] P. J. Havinga and G. J. Smit, "Energy-efficient wireless networking for multimedia applications," *Wireless communications and mobile computing,* vol. 1, pp. 165-184, 2001.

[25] C. E. Jones, K. M. Sivalingam, P. Agrawal, and J. C. Chen, "A survey of energy efficient network protocols for wireless networks," *wireless networks,* vol. 7, pp. 343-358, 2001.

[26] K. Naik and D. S. L. Wei, "Software implementation strategies for power-conscious systems," *Mob. Netw. Appl.,* vol. 6, pp. 291-305, 2001.

[27] A. Elkhodary and J. Whittle, "A survey of approaches to adaptive application security," in *Proceedings of the 2007 International Workshop on Software Engineering for Adaptive and Self-Managing Systems*, 2007, p. 16.

[28] J. Ma, C. Wang, and Z. Ma, "Adaptive Security Policy," in *Security Access in Wireless Local Area Networks*, ed: Springer, 2009, pp. 295-329.

[29] P. H. Lamanna, "Adaptive security policies enforced by software dynamic translation," Citeseer, 2002.

[30] C. Guo, H. J. Wang, and W. Zhu, "Smart-phone attacks and defenses," in *HotNets III*, 2004.

[31] P. Prasithsangaree and P. Krishnamurthy, "Analysis of energy consumption of RC4 and AES algorithms in wireless LANs," in *Global Telecommunications Conference, 2003. GLOBECOM'03. IEEE*, 2003, pp. 1445-1449.

[32] K. Lahiri, S. Dey, D. Panigrahi, and A. Raghunathan, "Battery-driven system design: A new frontier in low power design," in *Proceedings of the 2002 Asia and South Pacific Design Automation Conference*, 2002, p. 261.

[33] J. Troutman and V. Rijmen, "Green Cryptography: Cleaner Engineering through Recycling," *Security & Privacy, IEEE,* vol. 7, pp. 71-73, 2009.

[34] P. Rogaway and J. Steinberger, "Constructing cryptographic hash functions from fixed-key blockciphers," in *Advances in Cryptology–CRYPTO 2008*, ed: Springer, 2008, pp. 433-450.

[35] B. Preneel, R. Govaerts, and J. Vandewalle, "Hash functions based on block ciphers: A synthetic approach," in *Advances in Cryptology—CRYPTO'93*, 1994, pp. 368-378.

[36] S. Panasenko and S. Smagin, "Lightweight Cryptography: Underlying Principles and Approaches," *International Journal of Computer Theory and Engineering,* vol. 3, 2011.

[37] G. Leander, C. Paar, A. Poschmann, and K. Schramm, "New lightweight DES variants," in *Fast Software Encryption*, 2007, pp. 196-210.

[38] A. Bogdanov, L. R. Knudsen, G. Leander, C. Paar, A. Poschmann, M. J. Robshaw*, et al.*, *PRESENT: An ultra-lightweight block cipher*: Springer, 2007.

[39] Z. Gong, S. Nikova, and Y. W. Law, "KLEIN: a new family of lightweight block ciphers," in *RFID. Security and Privacy*, ed: Springer, 2012, pp. 1-18.

[40] C. De Canniere, O. Dunkelman, and M. Knežević, "KATAN and KTANTAN—a family of small and efficient hardware-oriented block ciphers," in

Cryptographic Hardware and Embedded Systems-CHES 2009, ed: Springer, 2009, pp. 272-288.

[41] S. P. Panasenko and S. A. Smagin, "Energy-efficient cryptography: Application of KATAN," in *Software, Telecommunications and Computer Networks (SoftCOM), 2011 19th International Conference on*, 2011, pp. 1-5.

[42] S. Panasenko and S. Smagin, "On Use of Lightweight Cryptography in Routing Protocols."

[43] O. Dunkelman, N. Keller, and A. Shamir, "Minimalism in cryptography: the even-mansour scheme revisited," in *Advances in Cryptology–EUROCRYPT 2012*, ed: Springer, 2012, pp. 336-354.

[44] H. C. Van Tilborg and S. Jajodia, *Encyclopedia of cryptography and security* vol. 2: Springer, 2011.

[45] S. Even and Y. Mansour, "A construction of a cipher from a single pseudorandom permutation," in *Advances in Cryptology—ASIACRYPT'91*, 1993, pp. 210-224.

[46] C. Lamprecht, A. Van Moorsel, P. Tomlinson, and N. Thomas, "Investigating the efficiency of cryptographic algorithms in online transactions," *International Journal of Simulation: Systems, Science & Technology*, vol. 7, pp. 63-75, 2006.

[47] Z. Guo, W. Jiang, N. Sang, and Y. Ma, "Energy measurement and analysis of security algorithms for embedded systems," in *Proceedings of the 2011 IEEE/ACM International Conference on Green Computing and Communications*, 2011, pp. 194-199.

[48] S. P. Singh and R. Maini, "Comparison of data encryption algorithms," *International Journal of Computer Science and Communication*, vol. 2, pp. 125-127, 2011.

[49] A. Shnitko, "Pratical and theoretical issues on adaptive security," in *Proceedings of FCS'04 Workshop on Foundations of Computer Security*, 2004, pp. 267-282.

[50] A. Shnitko, "Adaptive security in complex information systems," in *Science and Technology, 2003. Proceedings KORUS 2003. The 7th Korea-Russia International Symposium on*, 2003, pp. 206-210.

[51] Y. Xu, L. Korba, L. Wang, Q. Hao, W. Shen, and S. Lang, "A security framework for collaborative distributed system control at the device-level," in *Industrial Informatics, 2003. INDIN 2003. Proceedings. IEEE International Conference on*, 2003, pp. 192-198.

[52] S. P. Alampalayam and A. Kumar, "An adaptive security model for mobile agents in wireless networks," in *Global Telecommunications Conference, 2003. GLOBECOM'03. IEEE*, 2003, pp. 1516-1521.

[53] C. Chigan, L. Li, and Y. Ye, "Resource-aware self-adaptive security provisioning in mobile ad hoc networks," in *Wireless Communications and Networking Conference, 2005 IEEE*, 2005, pp. 2118-2124.

[54] S. H. Son, R. Zimmerman, and J. Hansson, "An adaptable security manager for real-time transactions," in *Real-Time Systems, 2000. Euromicro RTS 2000. 12th Euromicro Conference on*, 2000, pp. 63-70.

[55] J. Zou, K. Lu, and Z. Jin, "Architecture and fuzzy adaptive security algorithm in intelligent firewall," in *MILCOM 2002. Proceedings*, 2002, pp. 1145-1149.

[56] B. Gnedenko, I. V. Pavlov, and I. A. Ushakov, *Statistical reliability engineering*: Wiley-Interscience, 1999.

[57] B. M. Ayyub and R. McCuen, *Probability, statistics, and reliability for engineers and scientists*: Taylor & Francis US, 2011.

[58] S. Boucheron, G. Lugosi, and O. Bousquet, "Concentration inequalities," in *Advanced Lectures on Machine Learning*, ed: Springer, 2004, pp. 208-240.

[59] P. Massart, "Concentration inequalities and model selection," 2007.

[60] G. Lugosi, "Concentration-of-measure inequalities," 2004.

[61] G. Bennett, "Probability inequalities for the sum of independent random variables," *Journal of the American Statistical Association*, vol. 57, pp. 33-45, 1962.

[62] S. Bernstein, "The theory of Probabilities," ed: Gastehizdat Publishing House, Moscow, 1946.

[63] D. G. Kleinbaum, *Applied regression analysis and multivariable methods*: CengageBrain. com, 2007.

[64] C. Croarkin and P. Tobias, "Nist/sematech e-handbook of statistical methods," *NIST/SEMATECH, July. Available online: http://www.itl.nist.gov/div898/handbook*, 2006.

[65] M. S. L. Beck, *Applied regression: An introduction* vol. 22: SAGE Publications, Incorporated, 1980.

[66] Q. Wu, X. Qi, E. Fuller, and C.-Q. Zhang, "“Follow the Leader”: A Centrality Guided Clustering and Its Application to Social Network Analysis," *The Scientific World Journal*, vol. 2013, 2013.

[67] M.-J. Lee, J. Lee, J. Y. Park, R. H. Choi, and C.-W. Chung, "QUBE: a Quick algorithm for Updating BEtweenness centrality," in *Proceedings of the 21st international conference on World Wide Web*, 2012, pp. 351-360.

[68] F. Salvetti and S. Srinivasan, "Local flow betweenness centrality for clustering community graphs," in *Internet and Network Economics*, ed: Springer, 2005, pp. 531-544.

[69] L. C. Freeman, "A set of measures of centrality based on betweenness," *Sociometry,* pp. 35-41, 1977.

[70] J. Daemen and V. Rijmen, "AES proposal: Rijndael," *First Advanced Encryption Standard (AES) Conference,* 1998.

[71] D. Flanagan, *Java in a Nutshell*: o'reilly, 2013.

[72] J. Knudsen, *Java cryptography*: O'Reilly, 2010.

[73] Oracle, "Java Cryptography Architecture (JCA) Reference Guide for Java Platform Standard Edition 6," 2011.

[74] P. Kumar, *J2ee™ security for servlets, ejbs and web services: applying theory and standards to practice*: Prentice Hall Press, 2003.

[75] O. Jones, R. Maillardet, and A. Robinson, *Introduction to scientific programming and simulation using R*: CRC Press, 2009.

[76] L. A. Kirkpatrick and B. C. Feeney, *A simple guide to SPSS: For version 16.0*: CengageBrain. com, 2009.

[77] D. S. A. Elminaam, H. M. Abdual-Kader, and M. M. Hadhoud, "Evaluating The Performance of Symmetric Encryption Algorithms," *IJ Network Security,* vol. 10, pp. 216-222, 2010.

[78] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, 3 ed. New York: McGraw-Hill, 1991.

[79] J. S. Milton, Jesse C. Arnold, *Introduction to Probability and Statistics:Principles and Applications for Engineering and Computing Science*, 2 ed. New York: McGraw-Hill, 1990.

[80] C. Taramonli, R. J. Green, and M. S. Leeson, "Energy Conscious Ada[tive Security Scheme: A Reliability-based Stochastic Approach," *Performance Evaluation,* submitted.

[81] T. T. Soong, *Fundamentals of probability and statistics for engineers*: Wiley.com, 2004.

[82] W. Hoeffding, "Probability inequalities for sums of bounded random variables," *Journal of the American statistical association,* vol. 58, pp. 13-30, 1963.

[83] W. Feller, *An introduction to probability theory and its applications* vol. 2. New York: John Wiley & Sons, 1971.

[84] J. Shynk, *Probability, Random Variables, and Random Processes: Theory and Signal Processing Applications*. New Jersey: John Wiley & Sons, 2013.

[85] W. E. Franck, and D. L. Hanson, "Some results giving rates of convergence in the law of large numbers for weighted sums of independent random variables," *Bulletin of the American Mathematical Society,* vol. 72, pp. 266-268, 1966.

[86] S. Greenland, "Modeling and variable selection in epidemiologic analysis," *American Journal of Public Health,* vol. 79, pp. 340-349, 1989.

[87] J. Osborne and E. Waters, "Four assumptions of multiple regression that researchers should always test," *Practical Assessment, Research & Evaluation,* vol. 8, pp. 1-9, 2002.

[88] R. J. Freund, W. J. Wilson, and P. Sa, *Regression analysis*: Academic Press, 2006.

[89] O. D. M. Concepts, "11g Release 1 (11.1)," *Oracle Corp,* vol. 2007, 2005.

[90] H. J. Seltman, "Experimental design and analysis," *Online at: http://www.stat.cmu.edu/hseltman/309/Book/Book.pdf,* 2012.

[91] M. C. Newman, "Regression analysis of log-transformed data: Statistical bias and its correction," *Environmental Toxicology and Chemistry,* vol. 12, pp. 1129-1133, 1993.

[92] S. L. Weinberg and S. K. Abramowitz, *Statistics using SPSS: an integrative approach*: Cambridge University Press, 2008.

[93] R. G. Lomax and D. L. Hahs-Vaughn, *An introduction to statistical concepts*: Routledge New York, NY, 2012.

[94] S. C. Albright, W. L. Winston, and C. J. Zappe, *Data Analysis and Decision Making with Microsoft® Excel*: CengageBrain. com, 2009.

[95] D. S. Moore, *The basic practice of statistics*: Palgrave Macmillan, 2010.

[96] J. Durbin and G. S. Watson, "Testing for serial correlation in least squares regression. I," *Biometrika,* vol. 37, pp. 409-428, 1950.

[97] G. D. Hutcheson and N. Sofroniou, *The multivariate social scientist: Introductory statistics using generalized linear models*: Sage, 1999.

[98] M. J. Norusis, *IBM SPSS statistics 19 statistical procedures companion*: Prentice Hall, 2012.

[99] M. A. Schroeder, J. Lander, and S. Levine-Silverman, "Diagnosing and dealing with multicollinearity," *Western journal of nursing research*, vol. 12, pp. 175-187, 1990.

[100] S. L. Jackson, *Research Methods and Statistics: A Critical Thinking Approach: A Critical Thinking Approach*: Cengage Learning, 2011.

[101] J. K. Lindsey, *Applying generalized linear models*: Springer, 1997.

[102] J. Fox, *Regression diagnostics: An introduction* vol. 79: Sage, 1991.

[103] A. Field, *Discovering statistics using SPSS for Windows: Advanced techniques for beginners (Introducing Statistical Methods series)*: Sage, 2000.

APPENDIX A – Encryption schemes’ summary statistics

CASE	SCHEME	MODE	KEY	DATA	PADDING	Min	Max	Mean	Variance	Median	1st Quartile	3rd Quartile
1	AES	CBC	128	16	NoPadding	100000	304200	130500	1150647256	123300	111000	133000
2	AES	CBC	192	16	NoPadding	109800	1041000	143500	9965921223	125300	114500	134000
3	AES	CBC	256	16	NoPadding	100000	329900	122100	1041722756	112000	107500	121700
4	AES	CBC	128	2048	NoPadding	266800	450900	293100	661472724	283600	279100	297000
5	AES	CBC	192	2048	NoPadding	279900	876100	317400	4407720451	300600	291700	319500
6	AES	CBC	256	2048	NoPadding	296100	547600	324100	1056765518	314100	306700	328100
7	AES	CBC	128	4096	NoPadding	442800	982000	480200	3146667005	469900	461200	478100
8	AES	CBC	192	4096	NoPadding	474100	729700	503700	1030441118	495300	490200	502300
9	AES	CBC	256	4096	NoPadding	494800	1024000	534500	3072120189	521600	514900	540000
10	AES	CFB	128	16	NoPadding	87440	661000	129800	3943247964	112000	106700	124900
11	AES	CFB	192	16	NoPadding	100000	268500	120800	969624054	109000	105500	121300
12	AES	CFB	256	16	NoPadding	100900	267900	120400	652919085	111600	106900	121000
13	AES	CFB	128	2048	NoPadding	266200	540600	300600	1677312851	290000	284900	299000
14	AES	CFB	192	2048	NoPadding	286100	831700	324100	6376218216	303100	298000	317100
15	AES	CFB	256	2048	NoPadding	297500	511200	331200	1080600766	322500	314200	333800
16	AES	CFB	128	4096	NoPadding	456800	709900	493900	1811391030	481100	472500	497800
17	AES	CFB	192	4096	NoPadding	484100	828600	522200	1986281871	508400	502200	527200
18	AES	CFB	256	4096	NoPadding	509600	742600	547500	1070556514	536900	528800	560500
19	AES	ECB	128	16	NoPadding	87440	274300	112900	911539166	101400	95750	119200
20	AES	ECB	192	16	NoPadding	86320	268200	109900	1049264931	96520	92330	114100
21	AES	ECB	256	16	NoPadding	88000	246100	112900	649289121	105300	97150	119400
22	AES	ECB	128	2048	NoPadding	240300	392500	264800	680479145	256000	248900	268500
23	AES	ECB	192	2048	NoPadding	248900	483600	286300	1417235821	274500	263900	296300
24	AES	ECB	256	2048	NoPadding	262300	438300	295600	701566146	290000	276200	309000
25	AES	ECB	128	4096	NoPadding	398700	672700	433600	1108952453	425600	414800	446500
26	AES	ECB	192	4096	NoPadding	425200	593900	458200	779226064	450900	441400	464700
27	AES	ECB	256	4096	NoPadding	448100	654000	482000	1154747526	472400	463400	487800
28	AES	OFB	128	16	NoPadding	97780	298400	120200	847303671	109900	106200	123300
29	AES	OFB	192	16	NoPadding	98340	309800	120200	1231366065	107700	103900	118200
30	AES	OFB	256	16	NoPadding	98340	327100	119100	1154752087	109200	105900	118700
31	AES	OFB	128	2048	NoPadding	268200	498900	295800	1278323750	285800	280400	295000
32	AES	OFB	192	2048	NoPadding	277100	462900	310200	965214361	300600	294100	315500
33	AES	OFB	256	2048	NoPadding	294200	880300	342700	6684871957	318900	311100	342500
34	AES	OFB	128	4096	NoPadding	449200	1037000	490600	3974727421	474800	466300	494800
35	AES	OFB	192	4096	NoPadding	478000	1072000	530500	9327494283	500300	495500	527700
36	AES	OFB	256	4096	NoPadding	505100	1042000	551300	6200415098	528300	523400	555100
37	AES	CBC	128	16	ISO	98340	312600	125700	1001563116	114000	109800	129100
38	AES	CBC	192	16	ISO	98060	286900	123400	893774518	112900	108100	125300
39	AES	CBC	256	16	ISO	102500	386100	130600	1788575398	116600	111300	129200
40	AES	CBC	128	2048	ISO	265700	798100	303600	3643393110	288000	281500	302000
41	AES	CBC	192	2048	ISO	279400	592500	315100	2321450831	300700	292400	312800
42	AES	CBC	256	2048	ISO	291900	795100	333700	3924530220	315000	308100	334700

43	AES	CBC	128	4096	ISO	453100	1034000	506700	10384780070	478300	469300	498500
44	AES	CBC	192	4096	ISO	472100	908200	521700	3882736897	501200	490800	528800
45	AES	CBC	256	4096	ISO	501500	1022000	553300	5993719679	529300	521300	559200
46	AES	CFB	128	16	ISO	96940	340300	126500	1257197933	111700	108400	131100
47	AES	CFB	192	16	ISO	98060	292200	122100	837094002	110800	107200	120700
48	AES	CFB	256	16	ISO	102200	300900	126500	969288169	113800	110200	128600
49	AES	CFB	128	2048	ISO	275700	807100	311900	3663724059	293500	288800	311100
50	AES	CFB	192	2048	ISO	293100	880600	334800	7758128197	311800	302300	326600
51	AES	CFB	256	2048	ISO	303700	555400	340500	1821316089	326200	317600	344500
52	AES	CFB	128	4096	ISO	470500	750700	500800	1234337076	490300	484300	502400
53	AES	CFB	192	4096	ISO	488900	727500	526900	1144124139	515300	509700	531700
54	AES	CFB	256	4096	ISO	524100	1015000	563600	3162328342	547800	540900	571400
55	AES	ECB	128	16	ISO	83810	240300	113000	582205897	107400	99380	116600
56	AES	ECB	192	16	ISO	89120	393900	112700	1365767242	101400	95540	115400
57	AES	ECB	256	16	ISO	90790	641100	127800	8657179499	103800	97360	118400
58	AES	ECB	128	2048	ISO	246400	763500	281100	5520392231	260600	252700	280300
59	AES	ECB	192	2048	ISO	251400	551500	288500	1981127142	273600	267100	294900
60	AES	ECB	256	2048	ISO	267400	882000	302500	3989151306	293200	278800	305600
61	AES	ECB	128	4096	ISO	405100	625800	440500	1075005587	431800	420900	448100
62	AES	ECB	192	4096	ISO	431600	688900	467700	1344200062	457700	448600	478600
63	AES	ECB	256	4096	ISO	450600	1031000	501800	7028946350	479900	475200	497500
64	AES	OFB	128	16	ISO	88000	273500	124300	768133066	113800	110000	128400
65	AES	OFB	192	16	ISO	90240	314600	124000	1451790420	111600	106700	123100
66	AES	OFB	256	16	ISO	88000	351400	123800	1183348559	112700	109000	124000
67	AES	OFB	128	2048	ISO	271500	886400	310700	7097050950	291500	284800	300900
68	AES	OFB	192	2048	ISO	272900	565700	316400	1431144250	303900	298400	325700
69	AES	OFB	256	2048	ISO	296700	812100	336700	3340143308	322000	313700	341500
70	AES	OFB	128	4096	ISO	463500	1309000	510200	9687721170	487900	478300	507700
71	AES	OFB	192	4096	ISO	484400	939800	524900	2556542050	512500	504400	534600
72	AES	OFB	256	4096	ISO	518500	1091000	557600	4166165143	537800	532700	566600
73	AES	CBC	128	16	PKCS5	103900	323500	130700	883455454	121500	116400	132300
74	AES	CBC	192	16	PKCS5	104800	330500	131400	1413597202	117500	113100	136400
75	AES	CBC	256	16	PKCS5	103400	323800	130400	1008091260	120400	113900	132800
76	AES	CBC	128	2048	PKCS5	279100	473000	305400	928431300	292400	285400	316700
77	AES	CBC	192	2048	PKCS5	282700	550600	320900	1913200576	307400	298400	324500
78	AES	CBC	256	2048	PKCS5	299500	604500	336400	1823424411	324200	314800	342100
79	AES	CBC	128	4096	PKCS5	458200	990900	498200	3807091118	482500	472700	501600
80	AES	CBC	192	4096	PKCS5	484400	1103000	537600	9245805618	510800	501700	538300
81	AES	CBC	256	4096	PKCS5	511500	1093000	555200	7153584056	532300	526500	546900
82	AES	CFB	128	16	PKCS5	101700	354500	129000	1301802405	115200	111500	135600
83	AES	CFB	192	16	PKCS5	102000	364000	124800	1153356377	114400	110600	122700
84	AES	CFB	256	16	PKCS5	98060	228500	124300	491784738	116100	112800	124800
85	AES	CFB	128	2048	PKCS5	281900	448400	307900	787414767	298100	292500	312300
86	AES	CFB	192	2048	PKCS5	291400	416500	325200	767628307	313700	305800	337200
87	AES	CFB	256	2048	PKCS5	311800	510400	339200	1177836529	328000	320700	344900
88	AES	CFB	128	4096	PKCS5	471300	1469000	518700	11857464064	492500	486100	510800
89	AES	CFB	192	4096	PKCS5	501200	661000	534700	1214420069	521900	514800	540400

90	AES	CFB	256	4096	PKCS5	513200	720500	561200	1270367666	548300	540800	572000
91	AES	ECB	128	16	PKCS5	84930	270100	115800	900092033	107000	101100	113800
92	AES	ECB	192	16	PKCS5	91070	334100	112000	975486228	102400	96940	115200
93	AES	ECB	256	16	PKCS5	94150	247000	114900	650224981	103600	100400	116600
94	AES	ECB	128	2048	PKCS5	246100	398100	274400	846297013	263000	257400	278500
95	AES	ECB	192	2048	PKCS5	254500	756200	294800	3252355723	280100	270400	299300
96	AES	ECB	256	2048	PKCS5	268800	408700	303300	927717942	293100	283300	310900
97	AES	ECB	128	4096	PKCS5	407900	602000	443000	861705633	433600	426500	451200
98	AES	ECB	192	4096	PKCS5	433300	692500	476500	1862030794	461000	454100	490700
99	AES	ECB	256	4096	PKCS5	461800	1024000	505200	4100658742	487800	481300	507400
100	AES	OFB	128	16	PKCS5	106200	338900	132200	1178402752	120000	114300	133700
101	AES	OFB	192	16	PKCS5	105300	387200	129700	1350792726	115700	112200	131500
102	AES	OFB	256	16	PKCS5	86880	375700	128400	1612997933	114700	109700	127300
103	AES	OFB	128	2048	PKCS5	282200	536100	305800	1285031282	295000	288900	306500
104	AES	OFB	192	2048	PKCS5	288300	847300	327300	4681769076	308700	303900	318100
105	AES	OFB	256	2048	PKCS5	303900	770200	343800	3187549449	328000	319800	346100
106	AES	OFB	128	4096	PKCS5	467100	662100	497500	794874017	488200	481800	506800
107	AES	OFB	192	4096	PKCS5	486900	1048000	533100	4242360822	517000	507200	533600
108	AES	OFB	256	4096	PKCS5	514300	730800	558200	1163283178	544600	538700	567500
109	DES	CBC	56	16	NoPadding	102000	320400	123200	1210623929	110600	107500	120800
110	DES	CBC	56	2048	NoPadding	410400	586900	437500	677831149	431600	422300	440900
111	DES	CBC	56	4096	NoPadding	726900	987600	761300	1638048388	749300	741800	763400
112	DES	CFB	56	16	NoPadding	104500	440800	135000	2292881243	122100	113400	136500
113	DES	CFB	56	2048	NoPadding	412900	982500	446500	4448949098	433400	423400	441400
114	DES	CFB	56	4096	NoPadding	731900	1305000	777300	4333585643	759700	752100	783000
115	DES	ECB	56	16	NoPadding	85210	266800	116800	779554803	110100	98900	123200
116	DES	ECB	56	2048	NoPadding	375500	580800	406700	858764639	399200	391000	412700
117	DES	ECB	56	4096	NoPadding	674900	1243000	714400	3649713719	701900	692300	718800
118	DES	OFB	56	16	NoPadding	105000	230200	130900	623082294	124900	114200	134900
119	DES	OFB	56	2048	NoPadding	410700	635600	444300	1211457006	435300	426700	447400
120	DES	OFB	56	4096	NoPadding	731400	1054000	769700	1914708991	757200	750000	774100
121	DES	CBC	56	16	ISO	103400	318200	134400	1294000653	123800	119200	132100
122	DES	CBC	56	2048	ISO	407600	698400	447900	2259599600	436400	425900	447000
123	DES	CBC	56	4096	ISO	730800	995400	770200	1959618575	755500	748000	772300
124	DES	CFB	56	16	ISO	102000	637000	134600	3261267815	125400	114800	132000
125	DES	CFB	56	2048	ISO	411200	736100	448100	1947392014	439000	429400	449100
126	DES	CFB	56	4096	ISO	736700	989500	776000	1491628940	763200	757800	781400
127	DES	ECB	56	16	ISO	93030	282400	120700	951227765	112300	107300	122700
128	DES	ECB	56	2048	ISO	385500	624400	418400	1337958732	409400	401700	417900
129	DES	ECB	56	4096	ISO	681100	889500	719100	1295830967	708500	701400	725200
130	DES	OFB	56	16	ISO	107600	412300	139400	1700826097	127400	121200	142100
131	DES	OFB	56	2048	ISO	409300	916600	452100	3881735078	438200	429900	446500
132	DES	OFB	56	4096	ISO	731400	1279000	780800	6596441815	760200	755500	770100
133	DES	CBC	56	16	PKCS5	102200	1003000	143400	8654544069	128400	118900	137200
134	DES	CBC	56	2048	PKCS5	409000	690600	445500	1543568138	435700	424900	444900
135	DES	CBC	56	4096	PKCS5	729100	1299000	772500	4039375770	758600	751000	771300
136	DES	CFB	56	16	PKCS5	106200	262900	139700	740519982	131900	119000	153400

137	DES	CFB	56	2048	PKCS5	416800	965200	454500	3373226653	443600	431100	458500
138	DES	CFB	56	4096	PKCS5	741200	919100	790700	943482725	782500	771700	799600
139	DES	ECB	56	16	PKCS5	94150	284700	118700	604377919	112900	103600	126300
140	DES	ECB	56	2048	PKCS5	381600	585000	413300	940407857	407500	397700	416700
141	DES	ECB	56	4096	PKCS5	684700	1226000	726900	4040043999	713800	705300	727600
142	DES	OFB	56	16	PKCS5	105300	647600	145700	5767101290	129100	121400	138400
143	DES	OFB	56	2048	PKCS5	413500	690900	452600	1294428856	445900	438400	452900
144	DES	OFB	56	4096	PKCS5	738100	1245000	779500	3529871358	765200	757100	776900
145	TDES	CBC	112	16	NoPadding	116500	334100	146800	1193255286	137400	131900	148100
146	TDES	CBC	168	16	NoPadding	116500	358400	144900	1395986577	136200	126100	144200
147	TDES	CBC	112	2048	NoPadding	816300	1371000	865500	7092467541	845500	839400	860700
148	TDES	CBC	168	2048	NoPadding	816900	1017000	857900	764845512	850800	843100	867000
149	TDES	CBC	112	4096	NoPadding	1544000	2050000	1585000	2845183146	1575000	1567000	1583000
150	TDES	CBC	168	4096	NoPadding	1540000	2102000	1593000	4266895936	1576000	1569000	1596000
151	TDES	CFB	112	16	NoPadding	119300	474600	149600	1888455553	138800	135400	150200
152	TDES	CFB	168	16	NoPadding	117900	494800	148500	2143060334	138300	134800	148100
153	TDES	CFB	112	2048	NoPadding	823600	1375000	859900	3945071900	846800	835500	859700
154	TDES	CFB	168	2048	NoPadding	819400	1304000	864200	4935656700	847600	839100	858800
155	TDES	CFB	112	4096	NoPadding	1550000	2070000	1596000	5489458433	1577000	1569000	1592000
156	TDES	CFB	168	4096	NoPadding	1547000	2645000	1602000	14451419971	1578000	1571000	1594000
157	TDES	ECB	112	16	NoPadding	105300	640300	138700	4134394975	124600	115900	135100
158	TDES	ECB	168	16	NoPadding	106400	339200	133600	1541574492	123200	114300	133000
159	TDES	ECB	112	2048	NoPadding	791400	1004000	821100	992186043	813700	806200	822500
160	TDES	ECB	168	2048	NoPadding	790000	1012000	825500	901507008	818700	811300	828900
161	TDES	ECB	112	4096	NoPadding	1493000	2085000	1560000	14607089855	1523000	1514000	1541000
162	TDES	ECB	168	4096	NoPadding	1494000	1968000	1542000	4668360018	1523000	1517000	1545000
163	TDES	OFB	112	16	NoPadding	120100	414600	146000	1831733980	137200	125500	146600
164	TDES	OFB	168	16	NoPadding	118700	384700	143700	1397990237	137000	124000	144300
165	TDES	OFB	112	2048	NoPadding	820500	1276000	859400	2826258385	850900	838200	858800
166	TDES	OFB	168	2048	NoPadding	819700	1193000	857100	2141707878	846600	836500	861700
167	TDES	OFB	112	4096	NoPadding	1541000	2142000	1586000	4452608972	1572000	1563000	1587000
168	TDES	OFB	168	4096	NoPadding	1542000	1738000	1584000	1261897288	1574000	1563000	1593000
169	TDES	CBC	112	16	ISO	108400	654300	152500	4273254413	138400	127600	148700
170	TDES	CBC	168	16	ISO	113400	408700	145100	1672058543	138600	127000	143900
171	TDES	CBC	112	2048	ISO	823000	1051000	851300	978746301	845500	834300	853200
172	TDES	CBC	168	2048	ISO	823600	1321000	860600	3066260368	850100	841100	860700
173	TDES	CBC	112	4096	ISO	1551000	2139000	1606000	10673766529	1581000	1571000	1590000
174	TDES	CBC	168	4096	ISO	1545000	2126000	1607000	8569237207	1584000	1575000	1606000
175	TDES	CFB	112	16	ISO	121500	331300	148200	870105237	141500	134400	150200
176	TDES	CFB	168	16	ISO	119800	395900	147900	1241467318	141400	130500	148700
177	TDES	CFB	112	2048	ISO	823600	1419000	861200	3764272460	851600	839900	862100
178	TDES	CFB	168	2048	ISO	824400	1116000	864300	1585942790	854900	845300	872900
179	TDES	CFB	112	4096	ISO	1554000	2122000	1606000	7984338275	1586000	1578000	1601000
180	TDES	CFB	168	4096	ISO	1551000	1776000	1595000	1024020679	1586000	1579000	1603000
181	TDES	ECB	112	16	ISO	87440	246400	133600	519594214	127000	122200	138100
182	TDES	ECB	168	16	ISO	108400	632200	139700	3345460831	128500	118400	137000
183	TDES	ECB	112	2048	ISO	793700	1338000	831900	3467054835	820800	813400	827200

184	TDES	ECB	168	2048	ISO	799800	1289000	842500	6687378711	822700	813700	836600
185	TDES	ECB	112	4096	ISO	1510000	2045000	1564000	9114846903	1539000	1532000	1553000
186	TDES	ECB	168	4096	ISO	1511000	2048000	1552000	3647878310	1537000	1531000	1560000
187	TDES	OFB	112	16	ISO	122600	340500	151500	1235826938	143900	134000	153500
188	TDES	OFB	168	16	ISO	121500	395900	151900	1810300250	141500	135800	149000
189	TDES	OFB	112	2048	ISO	825500	1414000	870500	6865130270	852200	845000	864000
190	TDES	OFB	168	2048	ISO	824100	1014000	860600	833959713	853200	848200	862700
191	TDES	OFB	112	4096	ISO	1554000	2166000	1608000	7799881850	1584000	1576000	1608000
192	TDES	OFB	168	4096	ISO	1551000	2131000	1619000	11260991265	1590000	1580000	1612000
193	TDES	CBC	112	16	PKCS5	119300	669600	156200	3981869015	142600	130300	156200
194	TDES	CBC	168	16	PKCS5	109500	342200	148100	745524065	144400	131800	150400
195	TDES	CBC	112	2048	PKCS5	824100	1302000	861900	2677850162	851600	841800	864100
196	TDES	CBC	168	2048	PKCS5	826400	1337000	873600	5253777272	857500	847500	871100
197	TDES	CBC	112	4096	PKCS5	1551000	2068000	1601000	5399518835	1584000	1575000	1607000
198	TDES	CBC	168	4096	PKCS5	1552000	2119000	1601000	5252927380	1587000	1578000	1596000
199	TDES	CFB	112	16	PKCS5	126600	412600	157400	1992169893	146700	139300	159600
200	TDES	CFB	168	16	PKCS5	123500	383600	151400	906613347	145700	141500	154300
201	TDES	CFB	112	2048	PKCS5	835600	940600	863100	426571692	858300	852000	866500
202	TDES	CFB	168	2048	PKCS5	825500	1000000	861400	828215031	854900	848400	861900
203	TDES	CFB	112	4096	PKCS5	1557000	2028000	1601000	3384459006	1589000	1580000	1603000
204	TDES	CFB	168	4096	PKCS5	1566000	2128000	1613000	5449410291	1593000	1584000	1613000
205	TDES	ECB	112	16	PKCS5	109200	652000	143200	3213101716	131600	124500	145100
206	TDES	ECB	168	16	PKCS5	114800	367900	140600	1135722588	134500	120900	141800
207	TDES	ECB	112	2048	PKCS5	798100	1372000	846700	11166740471	818300	807900	833100
208	TDES	ECB	168	2048	PKCS5	796700	1366000	835500	4099117189	822900	812900	835900
209	TDES	ECB	112	4096	PKCS5	1510000	2092000	1560000	6535513514	1545000	1531000	1556000
210	TDES	ECB	168	4096	PKCS5	1499000	2096000	1564000	10768602627	1538000	1528000	1559000
211	TDES	OFB	112	16	PKCS5	125400	497000	155200	2176465192	146200	134600	154100
212	TDES	OFB	168	16	PKCS5	123800	520500	150700	2106223154	143900	131000	150900
213	TDES	OFB	112	2048	PKCS5	831100	1368000	875500	7588171220	856000	845000	867200
214	TDES	OFB	168	2048	PKCS5	831400	1024000	864800	891832170	857200	847900	868900
215	TDES	OFB	112	4096	PKCS5	1556000	1895000	1596000	1993245612	1586000	1575000	1604000
216	TDES	OFB	168	4096	PKCS5	1563000	2127000	1615000	7850168401	1590000	1584000	1616000
217	BF	CBC	56	16	NoPadding	238600	507300	266400	986091932	261100	252900	267700
218	BF	CBC	112	16	NoPadding	231000	382700	252100	513095242	244900	239700	256500
219	BF	CBC	256	16	NoPadding	231900	457300	256500	1161713564	245800	240000	258500
220	BF	CBC	56	2048	NoPadding	410700	599800	441700	489593450	435500	432200	444300
221	BF	CBC	112	2048	NoPadding	412900	678300	444800	985851758	434700	429400	449300
222	BF	CBC	256	2048	NoPadding	412100	985000	448000	3782431390	434700	429100	446800
223	BF	CBC	56	4096	NoPadding	616000	846800	647100	1090149389	636700	630000	654800
224	BF	CBC	112	4096	NoPadding	611500	1136000	651400	4008834534	637200	629900	649800
225	BF	CBC	256	4096	NoPadding	611200	829200	649300	959365322	642000	630500	659100
226	BF	CFB	56	16	NoPadding	220400	379400	257700	625774317	248500	242100	264600
227	BF	CFB	112	16	NoPadding	230800	458200	256500	940341786	247400	240700	260000
228	BF	CFB	256	16	NoPadding	234400	441700	254700	615936354	247500	242900	255700
229	BF	CFB	56	2048	NoPadding	416500	738400	447000	1363131847	438700	432900	446400
230	BF	CFB	112	2048	NoPadding	413500	652000	446600	1174436302	436600	431000	451600

231	BF	CFB	256	2048	NoPadding	418500	592800	447700	640556709	437800	432700	456000
232	BF	CFB	56	4096	NoPadding	621000	740300	644600	497271289	637000	630800	648300
233	BF	CFB	112	4096	NoPadding	623300	794800	649500	775237270	639600	634200	655000
234	BF	CFB	256	4096	NoPadding	615200	901000	652300	1641768631	638900	634200	663000
235	BF	ECB	56	16	NoPadding	219600	411200	240900	873359252	233100	228200	242600
236	BF	ECB	112	16	NoPadding	216800	392800	241100	535361507	233500	229200	242600
237	BF	ECB	256	16	NoPadding	218700	373500	243100	526078193	237000	229900	246500
238	BF	ECB	56	2048	NoPadding	386100	942900	415700	3192482564	404000	399100	417100
239	BF	ECB	112	2048	NoPadding	386600	600600	412000	999932528	403100	395900	418200
240	BF	ECB	256	2048	NoPadding	383600	611200	414700	1029708107	404900	400000	415300
241	BF	ECB	56	4096	NoPadding	565400	1087000	600600	3617448919	587200	580800	596200
242	BF	ECB	112	4096	NoPadding	561800	883900	600100	1268263952	588600	581600	609400
243	BF	ECB	256	4096	NoPadding	568800	827800	600500	995631144	590700	584700	608800
244	BF	OFB	56	16	NoPadding	236100	520700	260000	1580761583	246700	242500	261000
245	BF	OFB	112	16	NoPadding	233300	774400	267200	4986728220	247200	241900	268300
246	BF	OFB	256	16	NoPadding	236100	476600	258300	821993511	249200	243300	262300
247	BF	OFB	56	2048	NoPadding	416500	747000	449700	1865404158	439400	432500	452300
248	BF	OFB	112	2048	NoPadding	423500	661800	448200	1129668511	437600	431800	453000
249	BF	OFB	256	2048	NoPadding	422700	996800	461900	6825457299	439900	432200	460500
250	BF	OFB	56	4096	NoPadding	617100	1142000	653400	3349261585	638800	631600	655800
251	BF	OFB	112	4096	NoPadding	619600	933600	656400	2292087778	641100	635100	662000
252	BF	OFB	256	4096	NoPadding	619600	786400	651000	697471457	642400	635300	661000
253	BF	CBC	56	16	ISO	233800	535300	264100	2416950200	248600	243000	261600
254	BF	CBC	112	16	ISO	226300	783300	268100	5053343531	249200	241900	265400
255	BF	CBC	256	16	ISO	234900	438600	255900	656449128	247800	243300	257200
256	BF	CBC	56	2048	ISO	417400	654000	447100	1119444994	438300	432700	449000
257	BF	CBC	112	2048	ISO	417700	630500	451100	1159229956	440700	434100	452300
258	BF	CBC	256	2048	ISO	418500	678000	454400	1169484863	444900	436600	455200
259	BF	CBC	56	4096	ISO	618000	807900	649600	818338653	640200	634400	655700
260	BF	CBC	112	4096	ISO	615200	826100	649800	932482596	639900	634300	651800
261	BF	CBC	256	4096	ISO	614300	1212000	660100	4266912451	642100	636400	662200
262	BF	CFB	56	16	ISO	234900	448700	256200	679297408	247900	244200	258500
263	BF	CFB	112	16	ISO	234400	778600	261500	3781604033	247100	242800	258600
264	BF	CFB	256	16	ISO	232400	778000	269500	5547632589	250600	244400	271300
265	BF	CFB	56	2048	ISO	423000	690900	450200	1151978542	439700	435200	455500
266	BF	CFB	112	2048	ISO	420200	919900	453500	3265442751	437900	433800	454200
267	BF	CFB	256	2048	ISO	419000	660100	453900	1348577989	441500	434300	459700
268	BF	CFB	56	4096	ISO	618000	1120000	654500	2911912573	643700	637000	656700
269	BF	CFB	112	4096	ISO	620500	819100	647200	859272359	638800	631600	647800
270	BF	CFB	256	4096	ISO	617100	875500	652700	1317666509	642800	636700	656300
271	BF	ECB	56	16	ISO	221500	421300	242900	623839455	235800	231200	245400
272	BF	ECB	112	16	ISO	222400	417400	245100	1016325625	236500	230500	244000
273	BF	ECB	256	16	ISO	223200	445300	244700	961509669	235600	231200	245100
274	BF	ECB	56	2048	ISO	390000	912100	427100	3467302819	412500	403000	433600
275	BF	ECB	112	2048	ISO	385800	576600	419200	1025942055	408400	400300	430900
276	BF	ECB	256	2048	ISO	387200	912100	424700	2993796103	414000	403800	427400
277	BF	ECB	56	4096	ISO	566800	1115000	601600	3163112877	590000	581400	607100

278	BF	ECB	112	4096	ISO	562100	1087000	607200	4085893894	590600	583000	608300
279	BF	ECB	256	4096	ISO	570200	1143000	609000	6185437938	591700	583700	604000
280	BF	OFB	56	16	ISO	237700	438900	256300	837614339	248500	243800	259000
281	BF	OFB	112	16	ISO	230800	754000	263200	3512033134	248200	242700	258100
282	BF	OFB	256	16	ISO	236300	777200	265200	3455247117	251300	243900	266900
283	BF	OFB	56	2048	ISO	423800	924700	455500	2900132568	443800	436900	453800
284	BF	OFB	112	2048	ISO	425800	642300	450500	887389150	440700	433900	455200
285	BF	OFB	256	2048	ISO	427400	950100	455800	3228320566	441400	435000	455700
286	BF	OFB	56	4096	ISO	623300	1156000	665900	5627815226	645200	640000	668700
287	BF	OFB	112	4096	ISO	623500	876900	655000	1072445004	644100	638600	662000
288	BF	OFB	256	4096	ISO	621000	1108000	660500	2933078981	644400	637400	666300
289	BF	CBC	56	16	PKCS5	240500	378300	260500	554425191	251400	246700	263900
290	BF	CBC	112	16	PKCS5	225400	466800	257700	928339099	248900	244200	258800
291	BF	CBC	256	16	PKCS5	237700	361500	261400	529209410	251300	248400	264400
292	BF	CBC	56	2048	PKCS5	426600	945900	456600	3252018905	445000	436900	457500
293	BF	CBC	112	2048	PKCS5	425800	613200	457400	1066830504	445300	438900	464400
294	BF	CBC	256	2048	PKCS5	428000	627700	458200	982350181	448900	440400	465600
295	BF	CBC	56	4096	PKCS5	624400	1138000	661200	3954645589	645500	639700	656500
296	BF	CBC	112	4096	PKCS5	621900	1279000	665900	4638282578	651500	640600	676300
297	BF	CBC	256	4096	PKCS5	616300	869700	657900	1227525006	647600	639700	669000
298	BF	CFB	56	16	PKCS5	242800	481600	267300	986187355	255500	249100	273600
299	BF	CFB	112	16	PKCS5	240300	366500	260800	497747916	253000	248300	263800
300	BF	CFB	256	16	PKCS5	241400	390300	262900	703956487	253500	248100	265300
301	BF	CFB	56	2048	PKCS5	427400	952600	467300	4923940544	448500	442700	464000
302	BF	CFB	112	2048	PKCS5	427400	955100	472500	8838089732	447000	438800	461700
303	BF	CFB	256	2048	PKCS5	432200	647300	461700	1122884231	450600	444500	461000
304	BF	CFB	56	4096	PKCS5	628800	806500	657200	869767264	645900	639700	663600
305	BF	CFB	112	4096	PKCS5	623500	1122000	658600	3073475792	644100	639100	655700
306	BF	CFB	256	4096	PKCS5	625500	1196000	667300	4411535231	649800	643600	667400
307	BF	ECB	56	16	PKCS5	225400	569300	257400	2084645289	242500	234900	264300
308	BF	ECB	112	16	PKCS5	228200	450600	256900	1541229527	243900	237900	259000
309	BF	ECB	256	16	PKCS5	227700	403100	252400	817734310	242900	236500	258900
310	BF	ECB	56	2048	PKCS5	393100	616600	422500	891638099	413300	408100	425800
311	BF	ECB	112	2048	PKCS5	392200	620200	424700	1314678098	414200	407600	424600
312	BF	ECB	256	2048	PKCS5	397300	899600	427200	3127468067	413500	407600	428000
313	BF	ECB	56	4096	PKCS5	575200	1143000	610600	3742790413	596700	589700	613100
314	BF	ECB	112	4096	PKCS5	570500	1089000	609800	3417671515	595200	585200	616100
315	BF	ECB	256	4096	PKCS5	572400	1113000	605800	3543879694	592800	587200	604700
316	BF	OFB	56	16	PKCS5	241400	413500	263000	808396005	252100	248900	265800
317	BF	OFB	112	16	PKCS5	238600	766600	269000	3592627396	252500	246400	274000
318	BF	OFB	256	16	PKCS5	241400	774100	269800	3526988549	254500	248900	265500
319	BF	OFB	56	2048	PKCS5	428300	528000	450400	398583576	444100	436900	456300
320	BF	OFB	112	2048	PKCS5	428000	659900	455700	1125477899	444100	436900	460900
321	BF	OFB	256	2048	PKCS5	426300	959300	466600	3843464204	448800	443000	462600
322	BF	OFB	56	4096	PKCS5	619900	1181000	659000	4039863210	643000	637200	655400
323	BF	OFB	112	4096	PKCS5	620500	1181000	662700	5528548145	644500	637400	656000
324	BF	OFB	256	4096	PKCS5	620700	1160000	664300	3433393500	647600	638900	673700

325	RC2	CBC	40	16	NoPadding	94150	339100	117000	1168874774	106600	102500	117900
326	RC2	CBC	64	16	NoPadding	90790	324100	111600	1294061340	100300	95470	110300
327	RC2	CBC	128	16	NoPadding	91910	287500	111800	1020083362	100300	95750	109200
328	RC2	CBC	40	2048	NoPadding	318800	843700	355800	5232061584	337300	330400	348200
329	RC2	CBC	64	2048	NoPadding	312600	569900	348500	1279547266	339800	330400	356300
330	RC2	CBC	128	2048	NoPadding	318500	824400	352300	3123175030	338600	333800	352800
331	RC2	CBC	40	4096	NoPadding	559300	747900	590700	812160016	584000	576000	596900
332	RC2	CBC	64	4096	NoPadding	554300	1198000	610200	9755115576	586200	577100	610000
333	RC2	CBC	128	4096	NoPadding	558700	1068000	599800	3167203510	586900	578100	603000
334	RC2	CFB	40	16	NoPadding	92190	334100	115400	969438772	105000	101900	114100
335	RC2	CFB	64	16	NoPadding	91910	376900	112100	1208523314	103600	98830	111500
336	RC2	CFB	128	16	NoPadding	94980	254500	113900	525313887	106600	100700	118500
337	RC2	CFB	40	2048	NoPadding	341900	752300	375900	2195814967	362500	356400	375000
338	RC2	CFB	64	2048	NoPadding	340000	632800	379400	1769476073	366400	360500	381600
339	RC2	CFB	128	2048	NoPadding	342500	674400	379600	1846555856	367500	360900	384500
340	RC2	CFB	40	4096	NoPadding	610400	1077000	647800	2560248452	635700	628900	651300
341	RC2	CFB	64	4096	NoPadding	607600	1147000	653500	5718563768	634200	627900	649200
342	RC2	CFB	128	4096	NoPadding	613800	1157000	652100	5736645067	636100	626800	650700
343	RC2	ECB	40	16	NoPadding	77660	163700	97980	344585485	91630	86320	103500
344	RC2	ECB	64	16	NoPadding	77380	268500	101300	711483962	92050	84930	111600
345	RC2	ECB	128	16	NoPadding	74030	267600	98340	667041623	91770	86600	97640
346	RC2	ECB	40	2048	NoPadding	287700	390000	312700	407379632	306000	299400	317400
347	RC2	ECB	64	2048	NoPadding	286100	390800	310300	371157882	303700	297700	316100
348	RC2	ECB	128	2048	NoPadding	283800	368800	312400	347118087	305800	299200	320800
349	RC2	ECB	40	4096	NoPadding	502300	975000	537100	2364307211	526700	521600	539300
350	RC2	ECB	64	4096	NoPadding	505100	1093000	554100	8595888337	529500	521200	552600
351	RC2	ECB	128	4096	NoPadding	508400	784700	538600	1094067441	531400	522300	546300
352	RC2	OFB	40	16	NoPadding	96100	369000	118300	1153636862	106900	103900	114700
353	RC2	OFB	64	16	NoPadding	94980	307900	113500	857569823	102400	98620	115400
354	RC2	OFB	128	16	NoPadding	94150	296100	114400	1052307356	103600	98900	112700
355	RC2	OFB	40	2048	NoPadding	340000	847900	371600	3106763617	358400	352500	371800
356	RC2	OFB	64	2048	NoPadding	341900	897600	384600	7843201765	363900	355100	376200
357	RC2	OFB	128	2048	NoPadding	345600	530500	372400	760205011	363500	356700	379200
358	RC2	OFB	40	4096	NoPadding	601800	778000	635000	917946274	626800	618000	640700
359	RC2	OFB	64	4096	NoPadding	601500	721900	633200	503465986	625900	619800	643800
360	RC2	OFB	128	4096	NoPadding	597000	1135000	647100	5990410759	627200	619500	645500
361	RC2	CBC	40	16	ISO	93870	181600	114800	430402253	107400	103400	117400
362	RC2	CBC	64	16	ISO	93590	261800	112500	777206830	103400	99170	109600
363	RC2	CBC	128	16	ISO	94430	286100	115200	974119463	105000	99450	116000
364	RC2	CBC	40	2048	ISO	321000	910500	355600	4000402453	343100	335700	353800
365	RC2	CBC	64	2048	ISO	317900	582500	354600	1253134060	343200	337500	360100
366	RC2	CBC	128	2048	ISO	321800	858200	358400	3721755912	343800	338200	356600
367	RC2	CBC	40	4096	ISO	563200	1077000	603200	3428133500	587500	581000	604300
368	RC2	CBC	64	4096	ISO	561500	773300	597100	931874181	587400	581900	605600
369	RC2	CBC	128	4096	ISO	568500	1109000	615100	7323082208	589700	584400	608100
370	RC2	CFB	40	16	ISO	83250	274300	115800	581188134	108300	104800	116800
371	RC2	CFB	64	16	ISO	93310	346400	116400	1080299427	105300	101700	116400

372	RC2	CFB	128	16	ISO	96940	219000	117000	524104728	107600	103400	120100
373	RC2	CFB	40	2048	ISO	347300	453100	374400	412364671	368900	363500	379700
374	RC2	CFB	64	2048	ISO	349800	595600	379800	1397310813	371000	363000	380400
375	RC2	CFB	128	2048	ISO	346400	920200	385200	4161521790	369500	362500	381900
376	RC2	CFB	40	4096	ISO	614300	834200	646900	920572709	639900	631600	652500
377	RC2	CFB	64	4096	ISO	614300	912700	650800	1196277334	642000	634100	656200
378	RC2	CFB	128	4096	ISO	614300	861600	652900	1227455807	641600	635500	657600
379	RC2	ECB	40	16	ISO	77940	184900	98840	392738791	91490	87580	102200
380	RC2	ECB	64	16	ISO	78780	260900	102700	630099821	95260	88770	105600
381	RC2	ECB	128	16	ISO	77380	346700	106600	1350064265	96240	91280	111700
382	RC2	ECB	40	2048	ISO	291900	400300	314000	342258606	309500	301900	321100
383	RC2	ECB	64	2048	ISO	285500	528600	317200	1003527602	309100	301100	322700
384	RC2	ECB	128	2048	ISO	288900	459800	313300	549914004	307300	301400	317000
385	RC2	ECB	40	4096	ISO	510400	656200	537900	540220195	531400	525600	544800
386	RC2	ECB	64	4096	ISO	511500	1057000	546800	3645003954	534800	526500	546800
387	RC2	ECB	128	4096	ISO	514600	730500	548500	790698383	541300	533000	552900
388	RC2	OFB	40	16	ISO	99180	347800	122800	1318834999	110800	106400	125200
389	RC2	OFB	64	16	ISO	95820	301700	117500	959562364	106200	101400	123800
390	RC2	OFB	128	16	ISO	96660	214000	112700	430119944	105000	101100	116600
391	RC2	OFB	40	2048	ISO	343600	485800	373900	655326478	366500	357900	379100
392	RC2	OFB	64	2048	ISO	347500	526900	376400	1018075566	365800	360500	379900
393	RC2	OFB	128	2048	ISO	343900	536700	376700	894360615	367200	361200	380100
394	RC2	OFB	40	4096	ISO	609900	794200	639500	666959086	632600	625100	649900
395	RC2	OFB	64	4096	ISO	607300	1214000	658000	9019546736	633000	625400	646700
396	RC2	OFB	128	4096	ISO	611300	819100	648800	1195513630	637800	629400	654300
397	RC2	CBC	40	16	PKCS5	97220	276600	117600	644657089	108700	104200	121100
398	RC2	CBC	64	16	PKCS5	97500	283600	118100	918277039	107300	103200	117300
399	RC2	CBC	128	16	PKCS5	95260	207000	116400	594750964	106000	102200	121200
400	RC2	CBC	40	2048	PKCS5	326000	883400	357100	3196223871	346000	340500	359100
401	RC2	CBC	64	2048	PKCS5	323200	481300	352300	519292144	346800	339600	358600
402	RC2	CBC	128	2048	PKCS5	322700	448400	354600	519431893	347400	341200	363000
403	RC2	CBC	40	4096	PKCS5	570200	1091000	607800	3726600433	593200	587500	608900
404	RC2	CBC	64	4096	PKCS5	564600	793700	599600	992029100	589900	582100	609600
405	RC2	CBC	128	4096	PKCS5	563800	1132000	613800	4446990131	595500	588000	621700
406	RC2	CFB	40	16	PKCS5	98340	290500	121100	786834989	112600	105300	122500
407	RC2	CFB	64	16	PKCS5	96100	373200	120900	1312170390	108300	105000	120800
408	RC2	CFB	128	16	PKCS5	98620	256500	118700	676559755	109900	106100	119100
409	RC2	CFB	40	2048	PKCS5	350300	1297000	386600	9123433238	368900	363300	385000
410	RC2	CFB	64	2048	PKCS5	351200	539700	382500	916680983	370600	365900	391900
411	RC2	CFB	128	2048	PKCS5	351400	877500	392200	5180859701	373200	367000	394100
412	RC2	CFB	40	4096	PKCS5	615400	749500	646200	605383471	638100	630200	651200
413	RC2	CFB	64	4096	PKCS5	618800	825500	651600	1035705635	640700	632800	658700
414	RC2	CFB	128	4096	PKCS5	619600	906800	655400	1330685581	644200	637800	663500
415	RC2	ECB	40	16	PKCS5	84930	264000	106600	572071965	98760	93450	109600
416	RC2	ECB	64	16	PKCS5	74310	193300	99300	466318965	90790	86880	101800
417	RC2	ECB	128	16	PKCS5	79900	301400	104000	708363485	95960	90100	109300
418	RC2	ECB	40	2048	PKCS5	297200	398900	319900	448239531	312500	307800	325100

419	RC2	ECB	64	2048	PKCS5	291700	630800	317200	1299551732	308300	303800	319100
420	RC2	ECB	128	2048	PKCS5	294500	462600	324100	983658508	311900	307200	327000
421	RC2	ECB	40	4096	PKCS5	510700	638100	541200	458834494	535000	528800	547000
422	RC2	ECB	64	4096	PKCS5	515200	1069000	549500	3543643793	535800	530800	549000
423	RC2	ECB	128	4096	PKCS5	514600	624900	545300	573209241	537500	530400	555900
424	RC2	OFB	40	16	PKCS5	96380	248400	120400	639761043	111900	107600	120100
425	RC2	OFB	64	16	PKCS5	98900	308400	120300	818502704	112000	107000	118800
426	RC2	OFB	128	16	PKCS5	97220	322700	117000	899100751	107300	103900	115600
427	RC2	OFB	40	2048	PKCS5	345000	572100	376300	855627739	367600	362300	381100
428	RC2	OFB	64	2048	PKCS5	350300	442800	377000	445339216	369600	362800	384100
429	RC2	OFB	128	2048	PKCS5	349500	510700	378800	724898016	369300	364300	385500
430	RC2	OFB	40	4096	PKCS5	612600	826600	647100	1183789747	635100	629200	652500
431	RC2	OFB	64	4096	PKCS5	610100	1195000	648700	3763445769	634300	626000	657100
432	RC2	OFB	128	4096	PKCS5	615400	1198000	659600	9003309195	634000	628300	653400
433	AES	CBC	128	1024	NoPadding	183300	448100	217000	977424044	206700	201100	218700
434	AES	CBC	192	1024	NoPadding	190500	404500	227200	1405319179	214000	206600	236300
435	AES	CBC	256	1024	NoPadding	186600	484400	218900	1274249182	209000	200700	226000
436	AES	CFB	128	1024	NoPadding	177400	475200	203400	1204904863	194900	187000	208300
437	AES	CFB	192	1024	NoPadding	182700	517900	210400	1430006239	201100	192800	210300
438	AES	CFB	256	1024	NoPadding	191400	333800	218500	611200515	210100	202200	220800
439	AES	ECB	128	1024	NoPadding	155000	258700	177500	415862448	172500	163400	184200
440	AES	ECB	192	1024	NoPadding	164300	310700	187000	557113058	180200	173200	191100
441	AES	ECB	256	1024	NoPadding	173200	700400	203600	2910799020	193200	184900	207100
442	AES	OFB	128	1024	NoPadding	175200	719900	207600	3270009752	197900	184300	212500
443	AES	OFB	192	1024	NoPadding	179600	738900	214300	3367887924	202800	191600	219000
444	AES	OFB	256	1024	NoPadding	188600	325500	215900	611663060	208100	200700	218700
445	AES	CBC	128	1024	ISO	177700	359800	206200	840899795	198500	184700	216600
446	AES	CBC	192	1024	ISO	181900	317900	212300	804761642	202000	195800	215600
447	AES	CBC	256	1024	ISO	191900	310700	221600	697535802	212900	204700	228400
448	AES	CFB	128	1024	ISO	180700	325700	210800	894514812	201300	192400	216700
449	AES	CFB	192	1024	ISO	191400	787200	221100	3865460754	206000	201600	219800
450	AES	CFB	256	1024	ISO	198900	341100	224500	687120045	216800	210000	226400
451	AES	ECB	128	1024	ISO	152500	741400	190600	3708002944	177300	169300	193200
452	AES	ECB	192	1024	ISO	165700	291700	191300	527279123	185400	174900	195900
453	AES	ECB	256	1024	ISO	175700	330800	200300	731106671	193600	185400	205100
454	AES	OFB	128	1024	ISO	178200	655400	211300	2798447922	197500	191100	211400
455	AES	OFB	192	1024	ISO	183800	331300	213400	809365356	203200	196300	217800
456	AES	OFB	256	1024	ISO	192200	371600	222500	923709366	214300	204200	228700
457	AES	CBC	128	1024	PKCS5	181300	335800	211700	773757562	203700	193900	222700
458	AES	CBC	192	1024	PKCS5	187200	392000	222000	1111790203	209500	198700	237900
459	AES	CBC	256	1024	PKCS5	194400	317900	225300	694652239	217300	207700	236100
460	AES	CFB	128	1024	PKCS5	182400	308400	212800	702520302	205600	193300	224700
461	AES	CFB	192	1024	PKCS5	187500	298600	216900	576344378	208400	202300	224900
462	AES	CFB	256	1024	PKCS5	198600	322100	226800	665863696	219000	208700	235400
463	AES	ECB	128	1024	PKCS5	160600	300600	190000	780012571	181600	172900	192900
464	AES	ECB	192	1024	PKCS5	169300	276600	194000	534419136	185900	177900	198700
465	AES	ECB	256	1024	PKCS5	175400	686700	209200	3240055576	196700	186300	209000

466	AES	OFB	128	1024	PKCS5	177400	685600	213200	3699022991	198100	190200	210600
467	AES	OFB	192	1024	PKCS5	190200	722700	225100	3736165950	209200	200200	224100
468	AES	OFB	256	1024	PKCS5	197200	384700	226500	834136234	218200	210900	231000
469	DES	CBC	56	1024	NoPadding	244700	757100	272900	3197149176	258400	252200	269500
470	DES	CFB	56	1024	NoPadding	242500	476900	279000	1121292240	269300	263700	277700
471	DES	ECB	56	1024	NoPadding	223500	449500	255300	1073002453	247100	241400	253900
472	DES	OFB	56	1024	NoPadding	252800	495300	281100	1163643670	268200	261700	288400
473	DES	CBC	56	1024	ISO	239400	470700	280900	1000695220	269200	265100	284500
474	DES	CFB	56	1024	ISO	257300	845400	287500	4024451186	271700	265700	287000
475	DES	ECB	56	1024	ISO	229600	804600	264600	3850512946	249600	243500	267300
476	DES	OFB	56	1024	ISO	252800	596200	282600	1539043386	270100	264800	284600
477	DES	CBC	56	1024	PKCS5	248600	460700	278000	880728867	267900	262300	278500
478	DES	CFB	56	1024	PKCS5	252800	449800	286600	955167021	274200	268200	296000
479	DES	ECB	56	1024	PKCS5	237500	432500	260900	855886954	251800	246100	260600
480	DES	OFB	56	1024	PKCS5	258400	494800	285400	1306622773	272900	268400	281000
481	TDES	CBC	112	1024	NoPadding	459300	701800	484800	1417410917	469200	465600	488100
482	TDES	CBC	168	1024	NoPadding	452900	947300	489200	3931357186	469900	464800	479600
483	TDES	CFB	112	1024	NoPadding	458200	980900	489600	3881717265	472000	465400	484200
484	TDES	CFB	168	1024	NoPadding	456500	976400	490400	3754665235	472500	465700	487400
485	TDES	ECB	112	1024	NoPadding	436600	966300	466000	3768944891	449800	444100	459000
486	TDES	ECB	168	1024	NoPadding	438900	1117000	468200	5338666330	451200	447200	462200
487	TDES	OFB	112	1024	NoPadding	455400	1152000	494800	7570046064	471800	466000	483700
488	TDES	OFB	168	1024	NoPadding	455900	650100	486900	1282234562	472300	467100	491800
489	TDES	CBC	112	1024	ISO	459800	983400	493500	3468899984	478300	470200	496800
490	TDES	CBC	168	1024	ISO	463500	992000	501300	5979983329	479300	473700	494500
491	TDES	CFB	112	1024	ISO	464300	707100	489300	1218323585	476200	471200	490600
492	TDES	CFB	168	1024	ISO	465700	987600	493800	3853369675	476600	471300	488500
493	TDES	ECB	112	1024	ISO	443900	999300	472200	3701461080	455600	450600	469500
494	TDES	ECB	168	1024	ISO	441100	667700	472000	1336618978	459300	453400	468300
495	TDES	OFB	112	1024	ISO	464000	1084000	497500	4993666422	478800	472700	490700
496	TDES	OFB	168	1024	ISO	462100	988700	505100	6685688551	480400	474300	495000
497	TDES	CBC	112	1024	PKCS5	463200	681100	492900	1255948264	480500	473500	494800
498	TDES	CBC	168	1024	PKCS5	464600	686700	494200	1437868233	480600	474900	493100
499	TDES	CFB	112	1024	PKCS5	454200	701500	493200	1192657212	481800	476800	493600
500	TDES	CFB	168	1024	PKCS5	467700	715500	493400	1434738169	480600	476800	492200
501	TDES	ECB	112	1024	PKCS5	445300	985600	473500	3471330421	460400	455400	467200
502	TDES	ECB	168	1024	PKCS5	445300	684400	468600	1349100226	457600	452400	465500
503	TDES	OFB	112	1024	PKCS5	464300	857100	492300	2139528337	480600	471800	490000
504	TDES	OFB	168	1024	PKCS5	462900	769400	497700	1729253212	484800	478800	496600
505	BF	CBC	56	1024	NoPadding	325500	829400	366200	3733333890	352000	346300	361400
506	BF	CBC	112	1024	NoPadding	314000	949300	352200	4909370859	336900	330500	352900
507	BF	CBC	256	1024	NoPadding	317600	893700	350900	4352731777	336800	329300	351300
508	BF	CFB	56	1024	NoPadding	314600	573300	343800	1030641866	336200	326800	347900
509	BF	CFB	112	1024	NoPadding	317100	450100	344600	684601127	336100	328500	348400
510	BF	CFB	256	1024	NoPadding	321000	804900	357100	4664348763	342500	332100	355200
511	BF	ECB	56	1024	NoPadding	297500	848400	324300	3160966639	314300	306400	321900
512	BF	ECB	112	1024	NoPadding	297500	508400	321500	673956306	314700	307700	326900

513	BF	ECB	256	1024	NoPadding	298900	816900	327200	2919963929	316400	307200	327600
514	BF	OFB	56	1024	NoPadding	315700	857400	346000	3210615476	335000	325900	344300
515	BF	OFB	112	1024	NoPadding	316000	868800	346700	3578512463	333700	327100	345200
516	BF	OFB	256	1024	NoPadding	317600	851200	348800	3536488917	334800	325500	344600
517	BF	CBC	56	1024	ISO	317900	904000	347700	4089592298	336800	325500	344400
518	BF	CBC	112	1024	ISO	319600	845100	348800	3619210453	335900	330300	346100
519	BF	CBC	256	1024	ISO	319000	573800	345100	906352493	337600	331000	346100
520	BF	CFB	56	1024	ISO	317600	878300	355300	6325065526	338900	329200	350400
521	BF	CFB	112	1024	ISO	317400	662100	350900	1577361594	340300	331500	354000
522	BF	CFB	256	1024	ISO	321800	849300	352000	3124953547	340400	331300	351800
523	BF	ECB	56	1024	ISO	300300	542000	327000	979640077	317400	308800	334400
524	BF	ECB	112	1024	ISO	300900	430500	322600	391920480	317100	308000	326600
525	BF	ECB	256	1024	ISO	300000	579100	329700	1066536740	321300	315000	334900
526	BF	OFB	56	1024	ISO	309300	429100	346400	484247181	338200	332000	357500
527	BF	OFB	112	1024	ISO	319000	882500	352200	3767023471	338000	333200	348900
528	BF	OFB	256	1024	ISO	325500	528300	349400	821569449	341200	334000	350700
529	BF	CBC	56	1024	PKCS5	324100	561000	352500	1637677505	337900	332400	349700
530	BF	CBC	112	1024	PKCS5	326900	528600	351000	898530981	341800	334600	352100
531	BF	CBC	256	1024	PKCS5	322100	880600	353900	3909985851	340800	332400	348400
532	BF	CFB	56	1024	PKCS5	320400	855100	352500	3255853956	340000	330400	350900
533	BF	CFB	112	1024	PKCS5	323800	536400	352100	869216279	344700	335200	358200
534	BF	CFB	256	1024	PKCS5	320700	424100	346800	512142615	341700	333300	347600
535	BF	ECB	56	1024	PKCS5	304200	481300	327300	606883308	321500	312300	329000
536	BF	ECB	112	1024	PKCS5	306700	415400	325900	360849749	321800	312000	331300
537	BF	ECB	256	1024	PKCS5	305300	533300	329100	774235945	321100	313900	333800
538	BF	OFB	56	1024	PKCS5	321500	883900	358200	3407384933	345000	336600	359000
539	BF	OFB	112	1024	PKCS5	319600	534400	350400	775906984	342200	333900	359600
540	BF	OFB	256	1024	PKCS5	319000	978100	366900	9373951686	344600	336100	360200
541	RC2	CBC	40	1024	NoPadding	200900	378800	227100	650641653	218300	210600	235100
542	RC2	CBC	64	1024	NoPadding	197800	313400	225100	626352470	217900	207600	230100
543	RC2	CBC	128	1024	NoPadding	197500	378500	224700	964752286	214300	206900	232500
544	RC2	CFB	40	1024	NoPadding	211500	768800	242600	3488323431	231200	221000	243700
545	RC2	CFB	64	1024	NoPadding	214300	457900	242500	1212064288	232300	223200	248200
546	RC2	CFB	128	1024	NoPadding	211500	359500	237500	541855687	230500	223400	241700
547	RC2	ECB	40	1024	NoPadding	176600	1148000	208400	9625559789	193600	185900	204300
548	RC2	ECB	64	1024	NoPadding	179100	720800	216200	3603334446	199500	189500	226200
549	RC2	ECB	128	1024	NoPadding	180700	778000	224300	3858684097	217100	197000	235400
550	RC2	OFB	40	1024	NoPadding	215100	357300	249900	956383617	238400	229200	261200
551	RC2	OFB	64	1024	NoPadding	209000	726400	241000	2988782762	229100	220300	241400
552	RC2	OFB	128	1024	NoPadding	209800	424600	234700	878623488	223800	218700	236300
553	RC2	CBC	40	1024	ISO	197500	1207000	235400	10176256164	218500	210000	236300
554	RC2	CBC	64	1024	ISO	200900	363500	228400	668249864	220000	210100	239700
555	RC2	CBC	128	1024	ISO	201100	444500	227400	988629820	216900	211100	231100
556	RC2	CFB	40	1024	ISO	213200	487800	239600	1013734224	231700	224000	243700
557	RC2	CFB	64	1024	ISO	210400	500300	242700	1430009701	231200	223300	246300
558	RC2	CFB	128	1024	ISO	213400	464900	239900	905808602	231700	223800	245700
559	RC2	ECB	40	1024	ISO	177400	327100	202400	524006571	195300	189900	206200

560	RC2	ECB	64	1024	ISO	177400	725800	211400	6034166605	193500	188400	202500
561	RC2	ECB	128	1024	ISO	176600	320700	202700	601070177	195000	189800	207400
562	RC2	OFB	40	1024	ISO	212300	381900	239000	858049152	230100	223500	240000
563	RC2	OFB	64	1024	ISO	204500	710700	243800	3239155155	229800	220300	244500
564	RC2	OFB	128	1024	ISO	212300	402000	238900	882589144	229400	222200	245000
565	RC2	CBC	40	1024	PKCS5	202000	361200	227400	766261381	218300	211100	231500
566	RC2	CBC	64	1024	PKCS5	203100	861300	229800	4421298071	218200	210600	227800
567	RC2	CBC	128	1024	PKCS5	203400	785600	231300	3606120608	218600	212600	231200
568	RC2	CFB	40	1024	PKCS5	210400	699500	244500	2726832528	234700	225400	248100
569	RC2	CFB	64	1024	PKCS5	214800	374600	241100	508247478	235100	226800	245400
570	RC2	CFB	128	1024	PKCS5	214600	1154000	250400	8796685864	233700	225700	255100
571	RC2	ECB	40	1024	PKCS5	179100	295300	201200	323252128	195800	190100	205800
572	RC2	ECB	64	1024	PKCS5	180200	270700	204000	285467850	198600	191600	211000
573	RC2	ECB	128	1024	PKCS5	179600	269900	201700	282921488	198900	190700	204600
574	RC2	OFB	40	1024	PKCS5	213200	316800	240300	433824682	232400	226800	243300
575	RC2	OFB	64	1024	PKCS5	209000	708700	242500	2710760214	230100	222700	248500
576	RC2	OFB	128	1024	PKCS5	210900	353700	238200	589062180	229600	224800	241000