2013

Achievable secrecy enchancement through joint encryption and privacy amplification

Yahya Sowti Khiabani
Louisiana State University and Agricultural and Mechanical College, ysowti1@lsu.edu

ACHIEVABLE SECRECY ENHANCEMENT
THROUGH JOINT ENCRYPTION AND PRIVACY AMPLIFICATION

A Dissertation

Submitted to the Graduate Faculty of the
Louisiana State University and
Agricultural and Mechanical College
in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

in

The School of Electrical Engineering and Computer Science

by
Yahya Sowti Khiabani
B.Sc., University of Tabriz, 2003
M.Sc., University of Tabriz, 2007
August 2013

To my beloved wife, Rogaye,

and my wonderful parents, Naimeh and Khalil.

Acknowledgements

First and foremost, I would like to express my sincerest gratitude to my advisor, Dr. Shuangqing Wei for his invaluable and insightful guidance throughout the course of my PhD study at the Louisiana State University. I am so grateful for his kind support and guidance that helped me to succeed in my research career. He gave me the main idea of this work, and without his continuous encouragement next to his deep knowledge of communication and mathematics, this dissertation could not have been accomplished. I learned from him how to scientifically approach difficult problems and develop the true spirit of a diligent researcher.

I want to express my special gratitude to my dissertation committee, Dr. Morteza Naraghi-Pour, Dr. Xuebin Liang, Dr. Guoxiang Gu, and Dr. Suzanne D. Pawlowski for patiently attending my general and final exams and providing me with their constructive comments and suggestions. I would also like to thank Dr. Morteza Naraghi-Pour for his courses, Wireless Communication Networks, Error Control Coding and Random Process-II, and Dr. Xuebin Liang for his course, MIMO Systems. Your valuable classes had great impact on my knowledge and research insight.

I would also like to give my special thanks to Dr. Jian Yuan, Dr. Jian Wang and Dr. George T. Amariucai for providing insights into some of the problems encountered along the way.

Finally, I am obliged to many of my colleagues who supported me during my Ph.D. years. I thank Mahdi Orooji, Erfan Soltanmohammadi, Ali Haddadpour, Ahsan-Abbas Ali, Phuoc Doan Huu Vu, Shuhang Wu, Arash Karimpour and Kamran Ghavami for their warm friendship and support.

Table of Contents

List of Tables

List of Figures

Abstract

In this dissertation we try to achieve secrecy enhancement in communications by resorting to both cryptographic and information theoretic secrecy tools and metrics. Our objective is to unify tools and measures from cryptography community with techniques and metrics from information theory community that are utilized to provide privacy and confidentiality in communication systems. For this purpose we adopt encryption techniques accompanied with privacy amplification tools in order to achieve secrecy goals that are determined based on information theoretic and cryptographic metrics.

Every secrecy scheme relies on a certain advantage for legitimate users over adversaries viewed as an asymmetry in the system to deliver the required security for data transmission. In all of the proposed schemes in this dissertation, we resort to either inherently existing asymmetry in the system or proactively created advantage for legitimate users over a passive eavesdropper to further enhance secrecy of the communications. This advantage is manipulated by means of privacy amplification and encryption tools to achieve secrecy goals for the system evaluated based on information theoretic and cryptographic metrics.

In our first work discussed in Chapter 2 and the third work explained in Chapter 4, we rely on a proactively established advantage for legitimate users based on eavesdropper's lack of knowledge about a shared source of data. Unlike these works that assume an error-free physical channel, in the second work discussed in Chapter 3 correlated erasure wiretap channel model is considered. This work relies on a passive and internally existing advantage for legitimate users that is built upon statistical and partial independence of eavesdropper's channel errors from the errors in the main channel. We arrive at this secrecy advantage for legitimate users by exploitation of an authenticated but insecure feedback channel.

From the perspective of the utilized tools, the first work discussed in Chapter 2 considers a specific scenario where secrecy enhancement of a particular block cipher called Data Encryption standard (DES) operating in cipher feedback mode (CFB) is studied. This secrecy enhancement is achieved by means of deliberate noise injection and wiretap channel encoding as a technique for privacy amplification against a resource constrained eavesdropper. Compared to the first work, the third work considers a more general framework in terms of both metrics and secrecy tools. This work studies secrecy enhancement of a general cipher based on universal hashing as a privacy amplification technique against an unbounded adversary. In this work we also reach to the goal of exponential secrecy where information leakage to adversary, that is assessed in terms of mutual information as an information theoretic measure and Eve's distinguishability as a cryptographic metric, decays at an exponential rate. In the second work generally encrypted data frames are transmitted through Automatic Repeat reQuest (ARQ) protocol to generate a common random source between legitimate users that later on is transformed into information theoretically secure keys for encryption by means of privacy amplification based on universal hashing.

In chapter 5, we discuss possible future works as an extension of the accomplished research in this dissertation. Proofs of some Lemmas and Theorems are presented in Chapter 6 as Appendix.

Chapter 1
Introduction

In introduction we first discuss our main objectives in this dissertation and then present a brief preview of some concepts that are required as the basis of discussion in next chapters. First of all, a big picture that demonstrates the overall theme of this dissertation is given in Section 1.1. In this section two major common themes threading throughout the dissertation are discussed and then similarities and differences between different works presented in next chapters are analyzed concentrating on these unifying ideas.

Section 1.2 provides a brief preview of some information theoretic concepts like entropy and statistical distance that will be used throughout the dissertation. An introduction to physical layer secrecy and wiretap channel encoder based on secrecy capacity metric is presented in Section 1.3. We describe symmetric-key encryption and block ciphered systems in Section 1.4 that also presents a background for known plaintext attack and in particular linear cryptanalysis. In our first work discussed in Chapter 2, wiretap channel encoding is adopted as a privacy amplification technique along with a block ciphered system against eavesdropper. In this work secrecy capacity is utilized to evaluate security of the whole scheme, and linear cryptanalysis is considered as the basis of Eve's performance analysis.

Section 1.5 explains the principle of secret key agreement with emphasis on physical layer based secret key sharing. Section 1.6 provides a required background for privacy amplification as the main component in most information theoretic key generation algorithms. In particular, it discusses universal class of hash functions and randomness extractors as two major techniques for privacy amplification. Universal hashing is utilized as the main approach for privacy amplification in our works in Chapters 3 and 4. Note that the proposed method for key agreement in Chapter 3 lies in the category of physical layer based key sharing algoritms. The concept of randomness extractors is discussed in Chapter 4 where we try to define a new notion for extracting randomness.

Section 1.7 describes some secrecy metrics including mutual information, attacker's error probability and Eve's distinguishability that are adopted for secrecy analysis in next chapters. We also comparatively explain advantages and weaknesses of each one of these metrics. Mutual information as an information theoretic secrecy metric is used in all works in this dissertation to measure secrecy of the proposed schemes. Attacker's error probability or success rate is another metric that is adopted to evaluate secrecy of the ciphers in the works presented in Chapters 2 and 4. Eve's distinguishability is a universally composable metric used by cryptography community whereby performance of the proposed secrecy scheme in Chapter 4 is evaluated.

1.1 The Big Picture

The research on physical layer secrecy, which was initiated by Wyner's seminal work [1], has extended to many new problems and channel models including single-antenna, multiple-antenna, broadcast, interference, multiple-access channels etc, and mainly involves information theory (IT) community. On the other hand, researchers who work on ciphers, authenticated encryption etc, and rely on provable security are from theoretical computer science (CS) and crypto community. Both communities try to enhance secrecy, but from two different angles. The first one focuses on physical layer based techniques and protocols and relies on information theoretic metrics to reach perfect secrecy against unbounded adversary, but these techniques mostly require some knowledge or assumptions about eavesdropper's physical channel. The latter one utilizes cryptographic tools in higher layers, and relies on complexity based metrics against resource constrained adversary. They mainly discuss how much computationally hard it is for an adversary to mount the cryptanalysis and obtain some knowledge about system secrets.

Although IT-based security aims at reaching a stronger notion of secrecy than cryptographic based secrecy, its main disadvantage is that it relies on physical channel conditions that is not realistic in most cases. Moreover, physical layer based secrecy does not conform with end-to-end or link-wise secrecy, used mainly in internet or network security, where one may have no information about utilized communication channel or available techniques in physical layer. There have been some works that attempted to merge techniques and notions from both communities and achieve a stronger notion of secrecy but with lesser assumptions. In [2], Maurer exploited privacy amplification as a cryptographic tool with correlated randomness created based on public discussion and two-way communication model. The main advantage of Maurer's work was that it did not require this strong assumption that the adversary channel is noisier than the receiver one. Nevertheless, works in this line of research still require some knowledge of physical channel.

From crypto community, Bellare et.al in [3, 4] related IT-based secrecy metrics to provable security based on eavesdropper's advantage. They argued that IT-based metric which amounts to Eve's information in terms of mutual information is a relatively weak notion of secrecy since it resorts to the assumption that the input source has uniform distribution that is not necessarily true in realistic scenarios. Thus, they developed a new notion of mutual information security that requires Eve's information to be negligible for any possible input distribution. However, in this work privacy amplification is utilized in the context of wiretap channel which still requires some knowledge about physical channel condition. Recently some researchers from crypto community adopted information theoretic and statistical metrics like variational distance to measure secrecy enhancement against adversaries with unbounded computational power [4–8].

In all security schemes, confidentiality or authentication is built upon establishing some certain advantages or asymmetry of what legitimate users share over an adversary. For instance, in symmetric encryption Alice and Bob share a secret key that is unknown to eavesdropper where to measure secrecy we need to quantify how much effort with what success probability it takes for a bounded adversary to obtain the correct key. Physical layer secrecy requires a wiretapper channel that is degraded compared to the main channel either directly or through public discussion [1, 2, 9]. Some key extracting approaches assume that there already exists some random source of data shared between Alice and Bob that is partially secure from adversary rendering the required asymmetry to the system from which a highly secure key can be derived through privacy amplification [6, 10]. Reconciliation is an important step in most key agreement algorithms to generate this correlated randomness partially unknown to Eve that presents asymmetry in the system. This common randomness in the system can be established by the aid of a third party which supplies additional correlated information [11], existence of a public discussion and feedback communication [12, 13] or through existing extractable randomness in physical channel [14].

The overall theme of this dissertation is to bridge the gap between two communities: IT community and crypto community. In particular, our objective is to combine cryptographic tools and metrics together with secrecy enhancement techniques and measures in IT-based security into a single framework. Indeed, we try to leverage strength of both approaches to enhance secrecy in communications. In all of these works we attempt to take advantage of the existing or created asymmetry in favor of legitimate users over Eve by utilizing privacy amplification accompanied with encryption.

In Chapter 2 based on our work in [15], this asymmetry is provided through cipher keys shared between Alice and Bob about which Eve has no information. As a special case we use DES block cipher operating in CFB mode for encryption. We add adjustable noise into generated ciphertexts, thereby creating further difficulty for a resource constrained adversary who mounts linear cryptanalysis against this cipher. This additional hardness is manifested in reduced success probability of linear attack that consequently in multiple frames turns Eve's channel into a degraded version of the main channel. It provides additional secrecy in terms of secrecy capacity, as an information theoretic measure, that can be manipulated by a secrecy encoder applied over multiple frames to intensify Eve's uncertainty and deliver highly secure transmission. Therefore, secrecy encoder can be considered as a special class of privacy amplification techniques. In this work all procedure is performed in application layer in the context of end-to-end secrecy without any assumption on physical channel condition.

In Chapter 3 based on the work in [16], a two-way communication scheme based on ARQ mechanism exploits the authenticated but insecure feedback channel between targeted recipient and the transmitter, to create advantage over a passive adversary whose channel is partially independent from the main channel. We adopted a general packet erasure channel model that mostly conforms with link-wise communication. This work relies on statistical independence of Eve's channel from the main channel to provide advantage for Alice and Bob when a feedback channel is utilized to create a correlated randomness that is guaranteed to have a sufficient uncertainty on Eve's side. This random set shared between Alice and Bob is created over a data frame containing a large number of packets that are encrypted using the same symmetric key. By application of privacy amplification based on universal hashing over this generated randomness we obtain information theoretically secure keys that can be utilized for encryption of the next data frame. As a result, this work can be viewed as exploitation of cryptographic and information theoretic secrecy means to further enhance security based on the created advantage through two-way communication.

The third work presented in Chapter 4 based on [17] has more similarities with our first work since both, unlike the second work, assume an error free physical channel. In both of them encryption is used as a baseline to deliver primitive security over which applying privacy amplification enables highly confidential message transmission. Inverse universal hashing technique adopted in this work has similar properties with the wiretap channel encoding of the first work since both of them cause further confusion for Eve by injecting some fresh randomness into transmitted data for the purpose of privacy amplification.

The third work takes into account a more generalized setting than the first one in terms of metrics and secrecy approaches. In this work unlike the first one we do not specify the type of cipher instead consider a general cipher that generates key streams from cipher keys and combines them with plaintexts. Moreover, compared to the first work that assumes there already exists a key scheduling scheme that generates uniformly distributed keys, here we only require an initial key source shared between Alice and Bob that is partially known to Eve. We tailor to this weak source of randomness as the main asymmetry in favor of legitimate users and design a key generating scheme to extract nearly uniform keys out of it, later on being used as cipher keys. In this work we adopt variational distance or Eve's distinuishability as a universally composable metric from crypto community and information leakage in terms of mutual information as a strong secrecy metric from IT community. In terms of commonality with the second work both of them assume a computationally unbounded passive eavesdropper distinguishing them from the first work with a resource constrained adversary.

In summary, a common theme threading in all of these works is to establish and manipulate secrecy advantage using cryptographic and information theoretic tools in order to achieve a higher level of secrecy in communication evaluated on the basis of metrics from both communities.

1.2 Review of Basics on Information Theory

Information theory provides measures to quantify uncertainty of random variables [18]. Let X, Y be two random variables, with $\mathcal{X}$, $\mathcal{Y}$ as their sets of values, and $P_X(x) = Pr[X = x]$ and $P_Y(y) = Pr[Y = y]$ as their probability distributions. The entropy of X is defined as: $H(X) = -\sum_{x\in\mathcal{X}} P_X(x)\log P_X(x)$. When the logs are in base 2, entropy measures uncertainty of its outcome in bits. $H(X)$ takes the maximum value $\log|X|$ when X has uniform distribution, that presents the highest randomness.

Conditional entropy is defined as: $H(X|Y) = -\sum_{y\in\mathcal{Y}} P_Y(y)\sum_{x\in\mathcal{X}} P_{X|Y}(x|y)\log P_{X|Y}(x|y)$ which measures the remaining uncertainty or randomness in X when Y is known. The mutual information $I(X;Y)$ between two random variables X and Y is defined as $I(X;Y) = H(X) - H(X|Y)$ that measures the information known about X provided that Y is observed. Due to its symmetric property $I(X;Y) = I(Y;X)$. Note that both conditional entropy and mutual information take the maximum value of $H(X)$ when X and Y are totally independent.

Let $\tilde{P}_X$ be a uniform distribution on $\mathcal{X}$. L_1 distance of P_X from uniform distribution is

$$d_1(P_X, \tilde{P}_X) \triangleq \sum_x |P_X(x) - \tilde{P}_X(x)|. \tag{1.1}$$

Statistical distance between two distributions is half the L_1 distance between them. When statistical distance of X from uniform distribution is at most ε, it is called that X is ε-close to uniform. When the distribution P_Y of the random variable Y and the joint distribution of $P_{X,Y}$ are given, we get

$$\begin{aligned} d_1(P_{X,Y}, \tilde{P}_X \times P_Y) &= \sum_{x,y} |P_{X,Y}(x,y) - \tilde{P}_X(x)P_Y(y)| \\ &= \sum_y P_Y(y) \sum_x |P_{X|Y}(x|y) - \tilde{P}_X(x)| \\ &= \sum_y P_Y(y) d_1(P_{X|Y=y}, \tilde{P}_X). \end{aligned} \tag{1.2}$$

Rényi entropy of order $1+\alpha$ for $\alpha > 0$ is defined as [19, 20]:

$$H_{1+\alpha}(X) = -\frac{1}{\alpha} \log \sum_x P_X(x)^{1+\alpha}. \tag{1.3}$$

Then, we can define the conditional Rényi entropy as

$$H_{1+\alpha}(X|Y) = -\frac{1}{\alpha} \sum_y P_Y(y) \log \sum_x P_{X|Y}(x|y)^{1+\alpha}. \tag{1.4}$$

As another measure of difference between two distributions, we can obtain distance of P_X from uniform distribution in terms of Rényi divergence of order $1+\alpha$, for $\alpha > 0$ [19]

$$D_{1+\alpha}(P_X || \tilde{P}_X) = \frac{1}{\alpha} \log \sum_x \frac{P_X(x)^{1+\alpha}}{\tilde{P}_X(x)^{\alpha}}. \tag{1.5}$$

For $\alpha = 0$, KL-divergence is defined as

$$D(P_X || \tilde{P}_X) = \sum_x P_X(x) \log \frac{P_X(x)}{\tilde{P}_X(x)}. \tag{1.6}$$

The following inequality in [21] characterizes the relationship between KL-divergence and L_1 distance

$$-\log d_1(P_X, \tilde{P}_X) \geq -\frac{1}{2} \log D(P_X || \tilde{P}_X). \tag{1.7}$$

We can define min-entropy of random variable X as

$$H_\infty(X) = \min_{x\in\mathcal{X}} [-\log P_X(x)] = -\log\left(\max_{x\in\mathcal{X}} P_X(x)\right). \tag{1.8}$$

We say that X is a k-source if $H_\infty(x) \geq k$, i.e. if for all $x \in \mathcal{X}$, $P_X(x) \leq 2^{-k}$.

1.3 Wiretap Channel Secrecy

The notion of perfect secrecy in information theoretic terms was introduced by Shannon [22]. Suppose that Alice tries to securely transmit a k-bit packet **M** to a legitimate receiver Bob across a public channel. If **M** is encoded by Alice into a transmitted n-bit codeword **X**, perfect secrecy is said to be achieved if **I(M;X)=0**; meaning that the mutual information between **M** and **X** has to be zero. Shannon showed that in order to achieve this goal, Alice and Bob need to share k bits of the secret keys. This requires existence of a one-time pad which is an additive cipher that Xors message with a shared secure key of the same length of the message to generate a ciphertext. One-time pad is a theoretical cipher that is practically impossible to implement.

In his pioneering work [1] in 1975, Wyner introduced an alternative notion of communication known as wiretap channel coding with the general model shown in Fig. 1.1. In this model, the legitimate parties Alice and Bob are separated by a channel called main channel, and Eve observes information transmitted by Alice through a channel called wiretapper's channel, where these two channels are supposed to be discrete memoryless channels (DMCs). k-bit message **M** is encoded by Alice into an n-bit codeword **X**, but what Bob and Eve observe across two different channels are denoted by **Y** and **Z**, respectively. $\hat{M}$ is the decoder output at the receiver end. Alice does encoding such that not only can **Y** be decoded into **M** with arbitrarily small error probability, but also **Z** should not reveal any valuable information about **M** beyond what is available a priori. The first goal known as **reliability requirement** can be formulated as $\lim_{k\to\infty} Pr[M \neq \hat{M}] = 0$, and the second objective known as the **security requirement** can be formulated by $I(M;Z)/n \to 0$ as $n \to \infty$, meaning that for a large number of channel uses the average mutual information rate between Eve's knowledge and the secure message has to be negligible. Wyner showed that we can

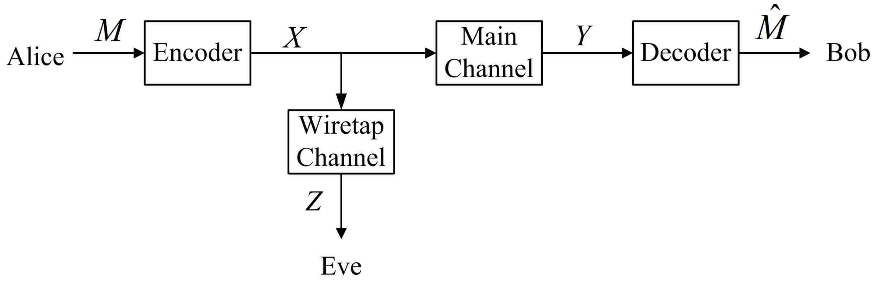


FIGURE 1.1. Wiretap channel

achieve both objectives by forward coding without any need for secret key sharing when these two channels satisfy the required conditions.

We should note that both reliability and security constraints are information theoretic rather than computational with the assumption that adversary is computationally unbounded. In other words, unlike cryptographic approaches that rely on computational hardness of the attack for adversary, Wyner's information theoretic secrecy does not depend on any assumptions on the wiretapper's resources and capabilities of any kind. Namely, physical layer secrecy provides a stronger notion of secrecy than complexity based cryptographic approaches. The results of Wyner's work have been extended to many contexts, most notably Gaussian channels [23] and broadcast channels with confidential messages [9].

The largest k/n for which both objectives of reliability and security are achievable in communication is called secrecy capacity which is a function of both main and wiretapper's channels. In [1] Wyner showed that if the wiretapper's channel is a concatenation of the main channel and another DMC, meaning that it is a degraded version of the main channel, the secrecy capacity will be positive. Csiszar et.al in [9] proved that when the main channel is *less noisy* than the wiretapper's channel, the secrecy capacity is positive. In other words, they showed that when the capacity of the main channel is higher than that of the wiretapper's, we would intuitively expect a positive secrecy capacity. When two channels are arbitrary, computation of secrecy capacity in general is an open problem. Suppose that X has distribution of $P_X(x)$. Let $I(X;Y)$ and $I(X;Z)$ denote the mutual information between Alice-Bob and Alice-Eve, respectively. It is proven in [24] by Van Dijk that if

$I(X;Y)$ and $I(X;Z)$ are individually maximized by the same input distribution of $P_X(x)$, and the main channel $(X \to Y)$ is less noisy than the wire-tap channel $(X \to Z)$, The secrecy capacity for the wire-tap channel will be

$$C_s = \text{Capacity}(X \to Y) - \text{Capacity}(X \to Z). \tag{1.9}$$

In Wyner-type encoder redundancy will be added to correct errors that occur across the main channel, and randomness is added for keeping Eve ignorant across the wiretap channel. If an encoder with information transmission rate of $R_s = k/n$ satisfies the reliability and security requirements for a given wiretap channel with secrecy capacity of C_s, such that $R_s = C_s$, then it is said that such an encoder achieves the secrecy capacity. Most of the work in the context of wiretap channel encoding rely on non-constructive random-coding framework with an argument that when the secrecy capacity is positive there exist codes that achieve secrecy capacity.

It should be pointed out that a general wiretap channel encoding is based on coset-coding or syndrom-coding that goes back to the works by Wyner, [1, 25]. This approach is further generalized and extended in [26, 27]. Coset-coding technique utilizes two binary linear codes: an inner code and an outer code. The inner code is a subset of outer code, assuming that the difference in their dimension is k, the outer code can be divided into 2^k cosets of the inner code. Each message corresponds to a linearly chosen coset, but what is transmitted by Alice is a uniform randomly selected codeword in that coset. Indeed, the outer code provides error correction across the main channel and therefore guarantees reliability, but the inner code based on which the choice of the codeword is randomized ensures secrecy. As a result, the problem is to construct inner and outer codes that satisfy both reliability and secrecy constraints and achieve the secrecy capacity. However, it is still a challenge to design an outer code that can be decoded across the main channel. So far, the constructive solution for the wiretap channel problem is only available in some special cases. For instance, when the wiretap channel is binary erasure channel (BEC) and the main channel is noiseless, a special class of LDPC codes are proposed in [28, 29], that are proven to achieve secrecy capacity. Recently, in [30], Mahdavifar et.al. used polar codes to construct coding schemes that can achieve secrecy capacity for a wider range of channel models that are symmetric with binary inputs.

1.4 Basics on Symmetric Encryption and Cryptanalysis

End-to-end cryptography is the most common technique used to ensure security in communication systems [31]. In all of these algorithms, the transmitter Alice tries to transmit a message to the receiver Bob, but meanwhile eavesdropper, called Eve, tries to obtain any knowledge about the message. The cryptographic algorithm with an encryption key determines a number of mathematical operations applied over the original message called plaintext, to generate a ciphertext (also called cryptogram). Since Bob is aware of the utilized key, he will be able to decipher and obtain the plaintext. Although it is often assumed that Eve knows the algorithm, decryption of the ciphertext to obtain the original message without knowing the key is computationally infeasible for her. Basically, there are two concepts of symmetric and asymmetric encryption in cryptography field. In this Section we give an introduction about symmetric key encryption, and asymmetric cryptography will be briefly discussed in Section 1.5.

1.4.1 Symmetric Encryption

A class of cryptography algorithms that use the same key to encipher the plaintext and decipher the ciphertext are called symmetric key encryption. This secret key, denoted by k, used for enciphering is indeed shared between Alice and Bob to keep a private information link. Alice enciphers the plaintext m using the encryption algorithm denoted by E and generates the ciphertext c where $c = E(k, m)$, and then sends c to Bob. Bob deciphers the received c using the shared key k to recover the plaintext $m = D(k, c)$. It is assumed that encryption and decryption functions E and D are publicly known [32]. The encryption algorithm is designed in the sense that without knowing the key k, it is computationally infeasible to apply the decryption function D over c and obtain the message. Symmetric encryption algorithms have the fastest hardware and software implementation among all encryption techniques. It makes them well suited for encryption of a large amount of data. However, their main disadvantage is that they require a secure channel to exchange and share secret keys. This can be resolved by using public key encryption, that belongs to asymmetric encryption class and are less efficient, or other existing secret key agreement algorithms that will be discussed in Section 1.5. That is the main reason why symmetric and asymmetric encryptions together provide a complete cryptosystem [32].

Symmetric encryption algorithms are divided into two major categories: stream ciphers and block ciphers. Unlike block ciphers that use a deterministic function to encrypt fixed length of plaintext, stream ciphers operate the encryption over a stream of plaintext, and processes it character by character while the encryption transformation and the length of the strings to be encrypted vary by time. In hardware implementation, they are faster than block ciphers and have less complexity [33]. Moreover, in situations that transmission noise is highly likely, stream ciphers are more appropriate since they cause much less or no error propagation compared to block ciphers. Due to these advantages stream ciphers are widely used in today's cipher systems. Some important stream ciphers include Linear Feedback Shift Register (LFSRs) based stream ciphers, SEAL (Software-optimized Encryption Algorithms) and RC4. Since in our work in [15] we used a specific block cipher for encryption and analysis, we dedicate a separate Section for block ciphers.

1.4.2 Block Ciphers and Cipher Feedback Mode of Operation

A block cipher is a symmetric key encryption algorithm with $M = C = \{0,1\}^n$, where M is the message space, C is the ciphertext space and with key space $K = \{0,1\}^r$:

$$E : \{0,1\}^r \times \{0,1\}^n \longrightarrow \{0,1\}^n, \quad (k,m) \longmapsto E(k,m). \tag{1.10}$$

The encryption algorithm E encrypts the plaintext block with a fixed length of n-bit by using the secret key k, to generate ciphertext blocks c with the same length of n, where n is called the block length, and r is called the key length [32].

The modern block ciphers are designed based on the notion of iterated product cipher. In his seminal work [22], Shannon suggested to use product ciphers in which simple operations like substitutions and permutations are combined as a tool to effectively enhance security of the message. In product ciphers, encryption is carried out in multiple rounds, where there exists an original key out of which different sub-keys for each round are derived. The structure of Data Encryption standard (DES) as the most well-known symmetric block cipher, is based on iterated product cipher in which each round involves a Feistel scheme [34]. This Feistel function includes expansion, key mixing, substitution and permutation. DES and some recent realizations of block ciphers like Advanced Encryption Standard (AES) belong to the class of substitution-permutation (SP) networks. SP network is a product cipher which consists of multiple stages each involving substitutions and permutations [33].

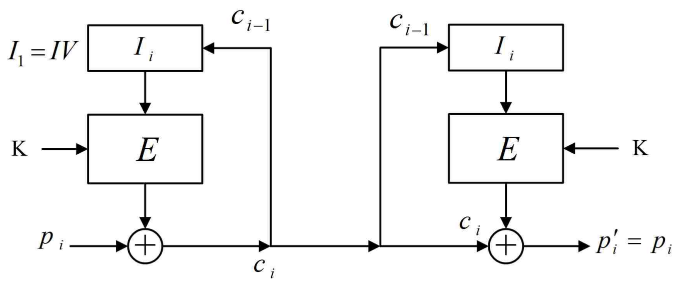
In DES cipher the substitution process in Feistel scheme is performed by 8 substitution boxes (S-boxes) that apply a non-linear transformation to their input bits based on a look-up table. S-boxes are the only non-linear mapping in DES that provide the core security for this cipher without which it would be easily breakable. S-boxes in block ciphers like DES and AES are designed in a way that the required “Confusion and Diffusion" introduced by Shannon in 1940’s [22] is satisfied. It requires that when one bit of the key or the input to the cipher is altered, decryption output will have a burst of errors. This property is called avalanche effect [35].

DES was published by the U.S. National Bureau of Standards (now National Institute of Standards and Technology, NIST) in 1977 as the first commercial-grade modern cryptographic algorithm. It has block length of 64-bit and key length of 64-bit where only 56-bit of these are actually used by the cipher as key bits. Due to the short key length that DES has, it is now considered to be insecure. In January 1999 it was publicly broken in a collaborative work done by distributed.net and Deep Crack within 22 hours and 15 minutes. Furthermore, because of some analytical weaknesses that it has, it was withdrawn by NIST as a standard and now is superseded by AES [34]. Although DES is replaced by AES, it is still being used and studied in some applications [36, 37] mostly in the form of Triple DES with three independent keys (168-bit key and 112-bit security) which is believed to be secure in practice. Moreover, many general attacks against block ciphers like differential and linear cryptanalysis were developed based on the studies on DES cipher making it the most studied and analyzed cipher. That is why we chose to use DES to investigate whether our proposed secrecy system in [15] can remain sufficiently secure when it is used to enhance secrecy of DES cipher which has well-known secrecy weaknesses.

After a 5 year competition, AES was selected to replace DES as a Federal standard and then was adopted by NIST. It was submitted by two Belgian cryptographers, Joan Daemen and Vincent Rijmen with the name of *Rijndael, Daemen*. AES algorithm [38, 39] is a symmetric block cipher that can encrypt blocks of size 128-bits, by using cipher keys with different lengths of 128, 192 and 256-bits. This cipher operates on a 4×4 column major order matrix bytes, called the state.

By a block cipher, one can encrypt a single block of data with cipher's block length. Using block ciphers with modes of operation allows us to utilize them in a secure and repeated way. Modes of operation are designed to derive a key stream from block ciphers like DES or AES. To encrypt a variable length message, it must be divided into separate cipher blocks, such that the last block must be extended to have the same length as the cipher block length by using a padding scheme. Each of these blocks will be processed within the chain structure of the operating mode which applies randomization by using an initialization vector (IV) as an additional input [33]. The five most common modes of operation for block ciphers are Electronic Codebook (ECB), Cipher-Block Chaining (CBC), Cipher Feedback (CFB), Output Feedback (OFB) and Counter Mode (CTR) [40].

In Chapter 2 we use CFB mode with block cipher of DES to analyze performance of a cipher in our scheme. CFB mode is one of the operational modes that can be used to transform a block cipher like DES into a stream cipher which is widely used in many applications [41, 42]. We consider a simple case when the plaintext is partitioned into blocks with the size of cipher's block.



(i) Encipherment (ii) Decipherment

FIGURE 1.2. Cipher Feedback mode

Its structure is illustrated in Fig. 1.2. The operation in CFB can be divided into three steps:

$$\begin{aligned} &\text{intialization:} \quad I_1 = IV. \\ &\text{Encryption:} \quad I_i = c_{i-1}, \quad c_i = E_k(I_i) \oplus p_i, \\ &\text{Decryption:} \quad p_i' = E_k(I_i) \oplus c_i, \end{aligned} \tag{1.11}$$

Where p_i' is the stored plaintext or decryption output at time i. In this structure the ciphertext at time i is used as input to the cipher at time $i+1$, implying that the currently generated ciphertext depends on both the current input and the previous ciphertext and consequently the preceding plaintexts. When the block cipher is operated in CFB mode, it acts like a self synchronizing stream cipher, meaning that when a block or a number of blocks are lost, after the same number of blocks it can resynchronize itself and avoid further errors in decryption. Another property of CFB is that encryption function E is used both for encipherment and decipherment [33]. CFB causes error propagation when a received ciphertext is noisy, that will be later discussed and analyzed in Chapter 2.

1.4.3 Known Plaintext Attack and Linear Cryptanalysis

When the attacker has both samples of the plaintexts and their corresponding ciphertexts, the attack model for cryptanalysis is called the known-plaintext attack (KPA). Linear cryptanalysis is a KPA which was first proposed by Matsui in [43] to attack DES. It is one of the most widely used attacks on block ciphers. This cryptanalysis approach exploits a linear equation with the probability of

$p \neq \frac{1}{2}$ which involves some input and output bits of the DES cipher and is used to obtain some key bits [43]. The quantity $\varepsilon = |p - \frac{1}{2}|$, which is called bias, measures the correlation among plaintext, ciphertext and the key bits, and can be used as a criterion to distinguish the right key. Before attack, Eve has to gather a large number of plaintext-ciphertext pairs by querying an oracle, and then for each possible key value count the number of pairs that satisfy the linear equation. Since the bias obtained by the right key will be considerably larger than the bias of a random key, attacker takes the key value that maximizes the bias as the right key.

If we refer to m as the number of attacked key bits in linear cryptanalysis, the number of subkey candidates would be 2^m that has to be sorted from rank 1 to 2^m based on their corresponding probability bias. It should be noted that it is not necessarily always true that the right key ranks the highest, but it will be surely among high ranked candidates. Assume that adversary only checks top 2^{m-a} candidates during exhaustive search, and since each subkey candidate gets checked with all possible combinations of $56 - m$ remaining unattacked bits, Eve has to run exhaustive search with at most 2^{56-m} encryptions for each candidate. As a result, the total number of 56 key bits examined in linear attack with bit advantage a is 2^{56-a}. In [44], A. Selçuk showed that when the number of attacked key bits m and the total number of gathered plaintext-ciphertext pairs N are large enough, the probability of success P_s, defined as the probability that the right key is among 2^{56-a} top candidates, can be derived as

$$P_s = \Phi(2\sqrt{N}\varepsilon - \Phi^{-1}(1 - 2^{-a-1})), \tag{1.12}$$

where a is the bit advantage of the attack, ε is the bias of the used linear approximation and Φ is defined as $\Phi(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty \exp(-u^2/2) du$.

1.5 Secret Key Agreement

The main difficulty in symmetric encryption is that it requires identical keys to be transmitted and shared between Alice, as a transmitter, and Bob, as a receiver, in a secure way before communication. in order to solve this problem Whitefiled Diffie and Martin Hellman in 1976 proposed the principles of asymmetric encryption in [45] which is also called public-key encryption. In this work they invented a key exchange protocol and revolutionized cryptography techniques. The first real cryptographic algorithm for public key encryption was designed by L. Rivest, K. Shamir, and L. Adleman at MIT, and was named RSA as the initial letters of its three inventors [46].

In public-key-encryption, the same key is not used for encryption and decryption, and therefore there is no need to share the same key between them. It works in this way that initially each user generates a pair of keys, one as a public encryption-key which is widely distributed and every user is aware of it, and one as a private-decryption-key that is known only to the recipient. The transmitter and every user can encrypt the message using the public key of the recipient while only the intended recipient can decrypt the ciphertext with his private key. These keys are mathematically related but are designed such that finding the private key from the known public key is computationally infeasible.

Since most of the public-key-encryption algorithms require randomly generating large prime numbers, which is computationally inefficient and slow, they are mainly used to communicate secret keys, and then Alice and Bob can use the shared keys in their fast computable symmetric encryption algorithms for secure communication. However, the existing computational as well as power constraints in some applications like wireless devices make public-key-encryption unfavorable for them. As a result, there is a need to present low complexity schemes that can handle key management problem. We try to deal with the problem in Chapter 3.

1.5.1 Physical Layer Based Secret key Sharing

The idea that physical channel characteristics can be utilized to enhance secrecy goes back to Wyner's work in [1] that was discussed in Section 1.3. However, Wyner's degraded wiretap channel was described to be unrealistic by Maurrer in his seminal work [2]. In this paper he presented an information theoretic based key agreement scheme with a two-way channel model that is publicly observable in the presence of a passive eavesdropper. His strategy is based on correlated randomness and public discussion. Its key elements are outcomes of the information reconciliation and privacy amplification procedures.

In reconciliation step both terminals come up with a randomly generated body of data that is identical between them through exploiting public discussion channel. This common random sequence that is agreed by two parties will be used to extract secret keys by privacy amplification. Although the correlated randomness may be partially known by Eve, privacy amplification reduces it into a shorter length sequence, that has a uniformly random distribution given Eve's information implying that she can gain almost no knowledge about it. Advantage of this scheme over Wyner's is that in certain cases it works even if Eve has a less noisy channel. Due to importance of privacy amplification we dedicate the whole Section 1.6 to discuss about it.

Maurrer defined secret key rate as the maximal achievable rate at which secret key can be generated by legitimate partners about which an eavesdropper has virtually no knowledge. In other words, secret key rate is the maximal rate at which Alice and Bob, by communicating over an authentic but insecure public channel, can generate secret keys in a way that Eve obtains knowledge about the shared key at an arbitrarily small rate [2].

Since Maurrer's scheme does not involve any complex computations of prime number generation, as does public-key-encryption, it offers a more efficient solution to secret key sharing problems. In a related work by Ahlswede and Csizar [47], the problem of secret-key sharing based on the generated common randomness is studied, and the concept of key capacity is defined. In [48], Csiszar and Narayan derived secret key capacity when a helper supplies additional correlated information for Alice and Bob. They characterized single-letter key-based capacities when there exist arbitrary number of terminals in [49]. The problem of physical layer secret key sharing studied in [2] was extended by Maurrer and Wolf in [13] to active adversary scenario when adversary, in addition to just eavesdropping, can actively interact with legitimate parties or even tamper with legitimate communications. In this work, they showed that secret key can be agreed at the same rate as the passive eavesdropper scenario or such a secret key sharing protocol is infeasible.

The next evolution in physical-layer-based secret key sharing was exploitation of inherent common randomness in wireless communication channels. One of the first examples of such techniques was proposed by Koorapaty et.al [14], in which based on the independence of channels of Alice/Bob and Alice/Eve, the secret key was extracted from the phase of fading coefficients. Since then there have been numerous techniques that utilized the randomness inherent in wireless channels for key generation [50], [51], [52]. There are also some other techniques like [53], [54] that utilized the well known ARQ protocol to facilitate exchange of secret keys between Alice and Bob. Our proposed scheme in [16], to be discussed in Chapter 3, is another application of ARQ mechanism to establish secret keys.

1.6 Privacy Amplification

Privacy amplification is a technique to distill highly secret shared information, from a large body of common information which is only partially secure. Suppose that legitimate users share a string $X = Y = S$, about which, however adversary has possibly some information. Privacy amplification is the art of transforming this partially secret string into a virtually secret key $\hat{S}$ about which Eve can only obtain arbitrary little information.

First described in the context of quantum key agreement, privacy amplification was generalized by Bennett et.al in [55] to probabilistic information about S. They showed that when from Eve's perspective the length of $\hat{S}$ is approximately equal to the Rényi entropy of S, we can make sure that she can only attain a negligible knowledge about $\hat{S}$. This privacy amplification technique is based on universal hashing. Another technique that is currently used for privacy amplification is based on random extractors [56], [57].

1.6.1 Universal Class of Hash Functions

Among all techniques for privacy amplification, universal hashing is a well-known approach against deterministic eavesdropping [55]. A class of hash functions that maps an n-bit binary string into a r-bit string is universal if the collision probability for two distinct inputs is 2^{-r} [55]. Universal hashing has this property that by uniform-randomly choosing a function from a universal class of hash functions, regardless of what distribution the actual input has, for sufficiently short output, the expected hash output will have a distribution close to uniform which results in the maximum entropy.

Let X be a random variable with distribution of P_X. The collision probability $P_c(X)$ is the probability that X takes the same value in two independent experiments:

$$P_c(X) = \sum_{x \in \mathcal{X}} P_X(x)^2 \tag{1.13}$$

The Rènyi entropy of order 2 of X is also called collision entropy since it can be written as the negative logarithm of the collision probability:

$$H_2(X) = -\log_2 P_c(X) \tag{1.14}$$

The conditional collision entropy on a random variable Y, $H_2(X|Y)$ can be computed by

$$H_2(X|Y) = \sum_{y} P_Y(y) H_2(X|Y=y) \tag{1.15}$$

The following bound provided by Bennett et.al in [55] (as Corollary 4), describes how Eve's collision entropy about a created randomness W conditioned on her observed data v limits her knowledge regarding output of universal hashing function applied over W:

Lemma 1. *Let P_{VW} denote an arbitrary probability distribution where v is a realization of random variable V observed by Eve. Let G be uniform randomly chosen function from a universal class of hash functions from $\mathcal{W}$ to $\{0,1\}^r$, when Alice and Bob choose $K = G(W)$ as their secure key, If Eve's collision entropy $H_2(W|V=v)$ about W is lower-bounded by c, we will have*

$$H(K|G, V=v) \geq r - \log_2(1 + 2^{r-c}) \geq r - \frac{2^{r-c}}{\ln 2}$$

Thus, when $r < c$, Eve's uncertainty about the secret key K is close to maximum value r as the entropy of the uniform distribution, and her information about this key K will be arbitrarily small.

For our analysis in Chapter 3 we use the following lemma, proven in [55] as Corollary 4 which is derived from Lemma 1:

Lemma 2. *Let W denote a random $n-bit$ string with uniform distribution over $\{0,1\}^n$, for an arbitrary eavesdropping function $e:\{0,1\}^n \longrightarrow \{0,1\}^t$, and let $V=e(W)$. let $s<n-t$ be a positive safety parameter, for some $t<n$, and let $r=n-t-s$. If Alice and Bob select $K=G(W)$ as their secret key where G is chosen randomly from a universal class of hash functions from $\{0,1\}^n$ to $\{0,1\}^r$, then Eve's expected knowledge about the secret key given G and V, satisfies $I(K,GV)\leq 2^{-s}/\ln 2$.*

Roughly speaking universal hashing extracts the minimum collision entropy of the weakly random source W into the secret string K, in the sense that the knowledge of Eve about K would be upper-bounded by $2^{-s}/\ln 2$ where s is the security parameter.

Hayashi in [58] showed that when input has sufficient entropy in terms of Rényi entropy of order $1+\alpha$, for $\alpha>0$, after application of universal hash function, Eve's information about the generated random variable decreases at an exponential rate that can be lower-bounded. The bound provided by Hayashi is more generalized and in some cases even tighter than the bound obtained by Bennett in [55]. It is the basis of our analysis in Chapter 4.

1.6.2 Randomness Extractors

Although universal classes are more economic compared to other functions, one limitation that these functions have is that they require a description as long as the string that forms their input. Extractors, as another technique for privacy amplification, allow us to more efficiently extract the randomness of some weakly random source into entirely random data, by using a small additional number of perfectly random bits called catalyst or seed, in a sense that these bits reappear as a part of the almost uniformly generated output [56]. Due to their efficiency, extractors have attracted lots of attention and intensely studies in recent years [57], [59], [60]. A formal definition of a randomness extractor according to [56] is as follows:

Definition 1. *A function* $Ext : \{0,1\}^N \times \{0,1\}^d \longrightarrow \{0,1\}^r$ *is called a* (δ, ε) *extractor if for any random variable* X *with range* $\mathcal{X} \subseteq \{0,1\}^N$*, and min-entropy* $H_\infty(X) \geq \delta N$*, the variational distance of the distribution of* $[S, Ext(X,S)]$ *to the uniform distribution over* $\{0,1\}^{d+r}$ *is at most* ε *when* S *is independent of* X *and uniformly distributed in* $\{0,1\}^d$*.*

As stated in their definition, extractors distill virtually all the min-entropy out of a weakly-random source $\mathcal{X}$, thereby requiring only a small number of truly-random bits S from the set $\{0,1\}^d$. This definition not only requires that the length of the extractor output is approximately equal to the min-entropy of the source plus the number of random bits, but that these bits even reappear as a part of the output. The following theorem proven in [56] measures the entropy of the adversary given her knowledge about the random bits S when an extractor is used for distilling randomness.

Theorem 1. *Let* $\delta, \Delta_1, \Delta_2 > 0$ *be constants. Then, there exists for all sufficiently large* N*, a function* $Ext : \{0,1\}^N \times \{0,1\}^d \longrightarrow \{0,1\}^r$*, where* $d \leq \Delta_1 N$ *and* $r \geq (\delta - \Delta_2)N$*, such that for all random variables* X *with* $\mathcal{X} \subseteq \{0,1\}^N$ *and* $H_\infty(X) > \delta N$ *we have*

$$H(Ext(X,S)|S) \geq r - 2^{-N^{1/2-o(1)}} \tag{1.16}$$

As Eq. (1.16) implies, when min-entropy H_∞ is at least a fraction of the length of the source X, for sufficiently large N, the second term in upper bound of the entropy goes to zero, thereby maximizing Eve's uncertainty about the extracted output given her knowledge.

Most of the recent research on extractors has focused on the extraction of secure keys from discrete noisy sources [59]. Fuzzy extractor is the basic primitive resulted from this work that allows to extract a secure cryptographic key from a noisy source. It consists basically of two phases. In the first phase that is called enrollment, by using probabilistic procedures a secure key and a helper string will be extracted from an original random source. The helper string is truly random and has to be considered as publicly available and hence attacker can observe it. The receiver receives a noisy version of the original source whose distance from it is less than a determined threshold. In the second phase which is called reconstruction phase the receiver will recover the original source from noisy data by using the helper and then extracts the secure key out of it. In other words, the reconstruction phase takes input as the noisy received data and the helper to reconstruct the original key.

Fuzzy extractor combines two functionalities, information reconciliation (also called error correction) and privacy amplification (ensuring that eavesdropper has negligible knowledge about the key). It was noted in [59] that a fuzzy extractor can in general be built upon two primitives: a secure sketch and a strong extractor. The secure sketch part makes it possible to exactly reconstruct the original source from the public helper string and the noisy received data. The strong extractor extracts the secret key from the reconstructed source. In [60], Ishai et.al. introduces the notion of correlation extractors as a generalization of randomness extraction and related the notion of privacy amplification to the case of two correlated sources. Correlation extractors extract nearly perfect instances of a given joint distribution from imperfect, or leaky, instances of the same distribution.

In Chapter 4, we introduce a new notion of Rényi entropy extractors that extracts randomness in terms of Rényi entropy which is generalization of the current extractors that measure randomness on the basis of min-entropy and statistical distance.

1.7 Secrecy Metrics

Consider a security function (like an encryption) $\mathcal{E} : \{0,1\}^m \to \{0,1\}^c$ that is applied over input M to generate the output $\mathcal{E}(M)$. What Eve receives through her channel ChA $: \{0,1\}^c \to \{0,1\}^d$ is $Z = \text{ChA}(\mathcal{E}(M)$. The security that this function provides with respect to an adversary can be measured using secrecy metrics. Secrecy metric xs denoted by $\textbf{Adv}^{xs}(\mathcal{E}, \text{ChA})$ measures the amount of information about message M that is present in Z. The smaller this number, the more security $\mathcal{E}$ is able to deliver. Some examples of secrecy metrics are semantic security, distinguishing security, mutual information and adversary success rate or decryptment error probability. In this chapter we discuss the last three that are relevant to the accomplished works in this dissertation.

1.7.1 Mutual Information Security

The secrecy condition that Wyner used in [1] was that $\lim_{n\to\infty} I(M;\text{ChA}(\mathcal{E}(M)))/m = 0$ where n can be considered as the number of channel uses or a parameter that both m and c are functions of it. Wyner assumes that message M is uniformly distributed over $\{0,1\}^m$. It was criticized by Maurer in [61] who proposed a stronger notion of secrecy condition which is $\lim_{n\to\infty} I(M;\text{ChA}(\mathcal{E}(M))) = 0$ with the remaining assumption of having a message with uniform distribution. This secrecy condition put forth by Maurer is equivalent to saying that $H(M|Z)$ also called equivocation moves to $H(M)$ for large n. Namely, knowing Z does not reduce Eve's uncertainty about the message. Since then mutual information between Alice and Eve's variables has been adopted as a measure of information leakage and secrecy criterion in many works by information theory community [1, 9, 47].

Bellare et.al. in [3] named this secrecy metric that was adopted by Wyner and Maurer as mutual information for random messages (Mis-r). He denoted it by $\mathbf{Adv}^{mis-r}(\mathcal{E},\text{ChA}) = \mathbf{I}(M;\text{ChA}(\mathcal{E}(M)))$ for it is defined for uniformly random message M. However, from cryptography point of view this metric is weak since we know that real messages are not uniformly distributed. They maybe English text, votes, scores of an exam that are not necessarily uniform messages. Namely, Mis-r can not ensure security for these types of data that have rise in applications of cryptography. In cryptography community the independence from message distribution has been viewed important for a good definition. Although it is argued in information theory community that message can be compressed before encryption, we should note that lossless compression is a deterministic operation that does not change entropy. Moreover, no universally source independent compression exists for finitely long messages [62].

In [3, 4], Bellare suggests using a stronger notion of secrecy called mutual informations security (MIS) defined as

$$\mathbf{Adv}^{mis}(\mathcal{E};\text{ChA}) = \max_{M} \mathbf{I}(M;\text{ChA}(\mathcal{E}(M))), \tag{1.17}$$

with maximization over all distributions of M over $\{0,1\}^m$. When $\mathbf{Adv}^{\text{mis}}$ is negligible, it ensures that the required security will be achieved regardless of how messages are distributed. It provides the required message distribution independence for security. In Chapter 4 we use mutual information between Eve's information and the message source as the secrecy criterion and require it to be negligible for any given distribution of the input. Thus, our defined secrecy metric can also capture distributions that arise in cryptographic applications.

1.7.2 Eve's distinguishability

Although MIS is a strong secrecy metric, its underlying intuition is somewhat obscure for cryptographers. They have very different approaches and intuition. Bellare in [3, 4] defines two other metrics that more conform with cryptographic approaches that are Semantic Security (SS) and Distinguishing Security (DS). Here we discuss DS.

Advantage of Eve's distinguishability or distinguishing security (DS) is defined as

$$\begin{aligned}\mathbf{Adv}^{ds}(\mathcal{E};\text{ChA}) &= \max_{\mathcal{A},M_0,M_1} 2Pr[\mathcal{A}(M_0,M_1,\text{ChA}(\mathcal{E}(M_b))) = b] - 1\\ &= \max_{M_0,M_1} \mathbf{SD}(\text{ChA}(\mathcal{E}(M_0));\text{ChA}(\mathcal{E}(M_1))), \end{aligned} \tag{1.18}$$

where b is a random variable uniformly distributed on $\{0,1\}$ and **SD** denotes the statistical distance which is half of the L_1 distance. The maximization is over all messages M_0, M_1 as strings in $\{0,1\}^m$ and all adversaries $\mathcal{A}$. $Pr[\mathcal{A}(M_0,M_1,\text{ChA}(\mathcal{E}(M_b))) = b]$ is the probability that adversary $\mathcal{A}$, given m-bit messages M_0, M_1, and the ciphertext resulted from M_b, is able to correctly identify the random challenge bit b. The advantage is defined as how success probability of adversary in distinguishing bit b differs from a priori success probability of $\frac{1}{2}$. As Eq. (1.18) indicates this advantage is equivalent to statistical distance between random variables ChA$(\mathcal{E}(M_0))$ and ChA$(\mathcal{E}(M_1))$.

Distinguishability security based on statistical distance was also adopted by Canetti in [7] as a universally composable secrecy metric. In seeking for a general methodology for expressing security requirements in any protocol environment, they proposed using variational distance or Eve's distinguishability to evaluate information leakage based on half of L_1 norm distance. Hayashi in [63] presents secrecy exponent analysis based on this metric. Variational distance is a metric that given Eve's knowledge measures statistical distance of the secret message from a uniformly generated random message over the same alphabet set. This metric will be analyzed in Chapter 4 where we compare it with the metric based on mutual information. Our analysis shows that mutual information is a slightly stronger metric compared to variational distance. However, variational distance is more compatible with cryptographic approaches and can be used to evaluate secrecy of any cryptographic protocol that brings about its main advantage which is universal composability.

1.7.3 Attack Success Rate or Eve's Error Probability

Cryptanalysis success probability has been used as a widespread secrecy criterion in cryptography community [43, 44, 64, 65]. In traditional and more strict definition of success rate it refers to the probability that the right candidate is found as the first key among sorted ones in the first phase of cryptanalysis [64, 65]. Selçuk in [44] proposed a new definition for success in attack in the sense that the correct key is found not necessarily as the highest-ranking candidate but among a set of high-ranking ones. In this analysis he provided formulas for direct calculation of the success probability of linear and differential cryptanalysis. We discussed his approach of analysis for linear attack in more details in Section 1.4.

If we consider a more general model for cipher where its type and the approach for cryptanalysis are not specified, success rate of the attack can be interpreted as the probability of correct decryptment [66, 67] or attack error probability [68]. Works in [66–68] consider Additive-Like Instantaneous Block (ALIB) cipher that has an additive-like function as a combiner of the message input and the key stream. In fact, they view cipher as a communication system encoder, whose code rate is the rate of key stream generation from a cipher key, and the cryptanalyst as the communication decoder. They try to choose a key rate and a cipher to make it improbable for the attacker to deduce any significant portion of the message. In other words, their objective is to design a bad code acting like a good cipher that results in a high probability of error for the decoder which is adversary [68].

Consider a sequence of n message blocks denoted by $M_1, M_2, \ldots, M_n$. Let cryptanalyst estimation of this sequence be $M_1^*, M_2^*, \ldots, M_n^*$. Average probability of correct decryptment for this sequence of n messages is defined as [67]

$$p_b = \frac{1}{n} \sum_{i=1}^{n} Pr[M_i^* = M_i]. \tag{1.19}$$

Similarly error probability of cryptanalysis will be $1 - p_b$. The designer attempts to make p_b and the key rate as small as possible. However, this secrecy criterion is relatively weak compared to other metrics like Eve's distinguishability or mutual information. First of all, since it quantifies error probability averaged over a large number of events, even for a very low success probability there could occur some rare events when Eve successfully decrypts the message and obtains the required information. Moreover, based on Fanos' lemma [18], a function of error probability provides an upper-bound for equivocation (Eve's uncertainty about the message given her knowledge) implying that high equivocation ensures a high error probability but not the other way around. Namely, mutual information and equivocation are much stronger secrecy metrics compared to average cryptanalysis error probability. Nevertheless, error probability gives a more intuitive understanding of performance of a cipher system. For instance, in Chapter 4 we adopt Eve's error probability of estimation of the plaintext as the secrecy criterion for the cipher that guarantees a primitive secrecy as the baseline to achieve a higher level of secrecy.

Chapter 2
Enhancement of Secrecy of Block Ciphered Systems by Deliberate Noise

In this chapter based on our work in [15], we consider the problem of end-to-end security enhancement by resorting to deliberate noise injected in ciphertexts. The main goal is to generate a degraded wiretap channel in application layer over which Wyner-type secrecy encoding is invoked to deliver additional secure information. More specifically, we study secrecy enhancement of DES block cipher working in cipher feedback model (CFB) when adjustable noise is introduced into encrypted data in application layer. A verification strategy in exhaustive search step of linear attack is designed to allow Eve to mount a successful attack in the noisy environment. Thus, a controllable wiretap channel is created over multiple frames by taking advantage of errors in Eve's cryptanalysis, whose secrecy capacity is found for the case of known channel states at receivers. As a result, additional secure information can be delivered by performing Wyner type secrecy encoding over super-frames ahead of encryption. These secrecy bits could be taken as symmetric keys for upcoming frames. Numerical results indicate that a sufficiently large secrecy rate can be achieved by selective noise addition.

2.1 Introduction

Traditionally, end-to-end secrecy delivery relies on symmetric or asymmetric encryption residing in the upper layer of a communication system, as well as sophisticated key management schemes [31, 69]. Without requiring a secure cipher, Wyner-type secrecy encoding provides a completely different solution to link-wise secret message delivery by random binning tailored to some presumed wiretap channel models in physical layer [1, 9]. In this chapter, we propose an encoding-encryption approach to end-to-end secrecy delivery by encoding over a degraded wiretap channel across super-frames transmitted in the application layer. The resulting wiretap channel is created by injecting controllable noise into ciphertext after encryption, and determined by both the adversary node's uncertainty about the key of cipher and its limited resources in launching cryptanalysis. Secrete information transmitted in such manner could be taken as keys for the subsequent super-frame.

In the proposed framework, we are essentially exploring the techniques developed for physical layer secrecy encoding and cryptanalysis against symmetric block ciphers to serve our purpose of realizing end-to-end secrecy enhancement without resorting to exogenous physical channel conditions. More specifically, Data Encryption Standard (DES) block cipher working in Cipher Feedback Mode (CFB) is taken to encrypt messages encoded using the Wyner type secrecy encoding scheme and then transmitted over multiple frames encrypted using different keys. Random binary noise is then deliberately added onto ciphertext, which are received by both legitimate user and an eavesdropper without any additional distortion. Such a hierarchical encoding-encryption framework allows us to transmit secrete messages over the resulting degraded wiretap channels in the application layer without making any assumption regarding end-to-end physical channel conditions.

In order to analyze secrecy enhancement achieved by utilizing our encoding-then-encryption approach, we need to study how Eve responds to the existing noise in her gathered data, and how that influences her cryptanalysis performance. In our case, Eve attempts to mount her linear attack with accumulated noisy ciphertexts, and thus applies a new verification strategy in the second phase of the linear attack while considering her possible resource constraints. Our statistical analysis shows that even when she uses a numerically optimized attacking strategy to obtain the key, it is likely for her to make mistakes in cryptanalysis. These possible failures of Eve over multiple frames make her channel degraded than the main channel, which can be further exploited by secrecy encoder to send additional secret bits over a super-frame. Therefore we could utilize generated secret bits over the last super-frame, whose secrecy is ensured by Wyner-type secrecy encoding scheme, to establish keys for next coming frames. The secrecy capacity of the system is computed assuming known channel states at Bob and Eve. Numerical results illustrate how deliberately added noise influences secrecy rate which can be further maximized at certain noise rate. It should be noted that the primary goal of our work is to demonstrate through such a case-study how secrecy encoding and symmetric encryption could be put together to enhance end-to-end security, and thus we only provide capacity computation of the resulting channel towards the end without dealing with the implementation of a particular secrecy encoder [28].

In literature, very few analytical approaches have focused on the impact of noisy ciphertexts on the attacking performance. In [70] different security schemes are analyzed from both reliability and secrecy perspectives in the presence of channel noise; nonetheless, they do not discuss what modified strategy Eve needs to take adaptively against degradation, and nor have they considered further leveraging adversary's failures in its cryptanalysis. In fact, our approach shares a common spirit with friendly jamming schemes proposed in physical layer secrecy encoding [71, 72] where deliberate noise is introduced in physical layer to interfere both legitimate link and eavesdropped link to improve the secrecy rate region. Unlike these works where link-wise physical channel features are explored to create a degraded wiretap channel, we essentially explore the adversary's disadvantages due to its uncertainty about the secrete key bits and resulting deteriorated success rate in cryptanalysis in the presence of deliberate noise.

In addition, deliberate additive noise in encryption process was used to improve security of ciphers in previous works [73–75]. The primary goals in these works were to enhance the secrecy of a cipher by random binning and additive noise, but we are interested in deploying encoding-then-encryption framework to enhance secrecy by further encoding over a resulting degraded wiretap channel. Random measurement noise has also been considered in side channel attacks (SCA) where information about cryptographic operation is leaked through some physical measurements conducted by an adversary [76]. In [77], authors proposed to use multi-linear approximation utilized in Differential Power Analysis (DPA)-like attacks, which is powerful due its robustness against noise.

The chapter is organized as follows. In section 2.2 the proposed security scheme is described, and in 2.3, we design an optimized verification strategy for Eve. In section 2.4 the main and wiretap channels are modeled, and then the resulting secrecy capacity is computed in section 2.5. The numerical results are presented in section 2.6, and finally we conclude this chapter in section 2.7.

2.2 The proposed scheme for security system

Fig. 2.1 illustrates the proposed scheme for secrecy improvement in which after encryption of the original message S, intentional noise is injected into it to generate a degraded wiretap channel. Since we consider end-to-end secrecy, physical channel is assumed to be error-free. Therefore, the ciphertexts that Bob obtains only include errors caused by intentional noise introduced into encrypted data in application layer with bit error rate of η. Moreover, because Alice and Bob agree on the key used for the current data frame, Bob can decrypt the obtained noisy ciphertexts and then apply the wiretap channel decoding algorithm that allows him to recover the original message $\hat{S}$ with arbitrarily small error probability. As indicated in Fig. 2.1, there exists an oracle, whereby Eve can query and obtain consecutive plaintext/ciphertext pairs. However, due to the deliberate noise, it provides Eve with noisy ciphertexts distorted with independent errors of rate η. As the main advantage over Eve, Alice and Bob share the same encryption and decryption key which is unknown to Eve. Therefore, she has to adopt an attack strategy that can exploit the gathered noisy data in order to guess the secret key.

We assume that legitimate users initialize with a shared set of keys in a highly secure manner at the beginning. As a result, Alice can divide the whole data into equal size data frames, each including M number of data blocks of size 64-bit which is the block size used in CFB mode. In this way, the same key will be used for M 64-bit blocks in each frame for encryption and decryption at the receiver end. In this chapter, we show that due to Eve's resource constraints, it is likely for her to make mistakes in assessing a frame key. As a result, Eve's channel is a degraded version of the main channel. We can leverage this advantage by applying Wyner secrecy encoding over

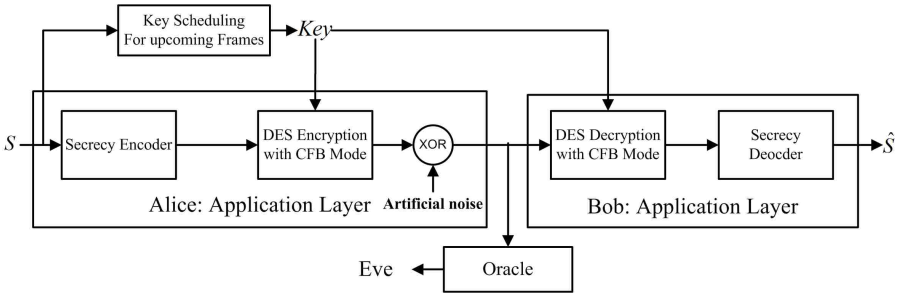


FIGURE 2.1. The proposed security scheme based on the intentional noise

super-frames to average over all possible failures by Eve. In Wyner-type encoder redundancy is added to correct errors that occur across the main channel, and randomness is added for keeping Eve ignorant across the wiretap channel [1], [28].

Another issue is key scheduling problem to provide highly confidential and distinctive keys for each frame while Bob is fully aware of them. The traditional methods of key management like master/session key scheduling approaches [31, 69] are sophisticated and costly. Here, we propose a simpler technique that derives the required secrecy for frame keys from secret bits delivered by Wyner secrecy encoder over the created wiretap channel. As a result, since encoder is performed over each super-frame, Alice can use input to the encoder to extract frame keys in next super-frame, for instance by applying a universal class of Hash functions over it [55]. Bob is able to decode encrypted data and obtain the encoded message, and thus he will be able to derive keys for next frames.

2.3 Eve's attack strategy and its analysis

This section studies the effect of the channel degradation on the performance of the linear cryptanalysis in terms of Eve's success rate. Since linear cryptanalysis is a known plaintext attack, Eve has to rely on the received plaintext/ciphertext pairs. Due to the existing errors in these ciphertexts, when Eve examines a key, she is unable to distinguish between errors caused by the received noisy ciphertext and the ones induced by using the wrong key. Thus, she needs to design a new verification approach that gives her the maximum possible success rate in finding the right key.

2.3.1 Designed Verification Strategy for Attack

Consider ciphertexts go through a binary symmetric channel whose cross-over probability is η. Let C_n be the ciphertext block at time n. After it passes through channel and Xors with channel noise, the received noisy 64-bit ciphertext $\hat{C}_n$ will have error with the probability of $1-(1-\eta)^{64}$. Therefore, Eve can not rely only on two successive ciphertexts to check the correctness of a key, because they might have errors that can lead her to make mistakes. Indeed, Eve has to try a number of successive pairs, using CFB mode in order to increase her success rate.

In Fig. 2.2, two consecutive stages of CFB that are used to check the key are shown, where P_i and $\hat{C}_i$ are respectively the plaintext and ciphertext for the i^{th} stage, S_i is the encrypted result of $\hat{C}_{i-1}$ that after Xor with P_i generates $\hat{C}_i^h$. Provided that the used key is correct, $\hat{C}_i^h$ must be the same as $\hat{C}_i$, but due to the possible errors in $\hat{C}_i$ or $\hat{C}_{i-1}$ there might be some differences between them. Hence, Hamming Weight (HW) of Xor of $\hat{C}_i^h$ and $\hat{C}_i$ denoted by E_i must be compared with a threshold denoted as τ. Then, a key trial for the i^{th} stage can be considered successful if this HW is less than τ.

Note that at stage i when there is an error either in the input to the cipher, i.e. $\hat{C}_{i-1}$ or in the key, there will be burst of errors in S_i, which makes $\hat{C}_i^h$ totally different and in special case of $\alpha = 0.5$ independent from $\hat{C}_i$. Therefore, by choosing a small value for threshold τ and comparing HW of $\hat{C}_i^h \oplus \hat{C}_i$, Eve can know that either input to the cipher or the key is noisy. In Table 2.1, the key verification strategy for Eve is given that she needs to follow in the brute-force attack phase of linear cryptanalysis to test the correctness of the examined key k_i. In this strategy, Eve examines each key candidate N_c times with N_c consecutive pairs where N_c is chosen such that with a high probability at least in one trial out of N_c tests, input to the cipher has no error. Then, Eve can recognize the correct key when at least one of trials is successful.

Since when the examined key is correct and $\hat{C}_{i-1}$ is error-free, all the discrepancies between $\hat{C}_i^h$ and $\hat{C}_i$ will be caused by the possible errors in $\hat{C}_i$, we can determine the minimum value for τ such that the probability that the number of bit errors in $\hat{C}_i$ exceeds τ denoted by P_{fault} becomes

TABLE 2.1. Verification strategy

1- Pick N_c number of consecutive pairs.
2- Try N_c chosen pairs over N_c chained CFB stages using the key k_i.
3- A trial is successful if $HW(E_i = \hat{C}_i \oplus \hat{C}_i^h) \leq \tau$.
4- If there exists at least one successful event out of N_c trials,
k_i is the correct key, otherwise it is wrong.

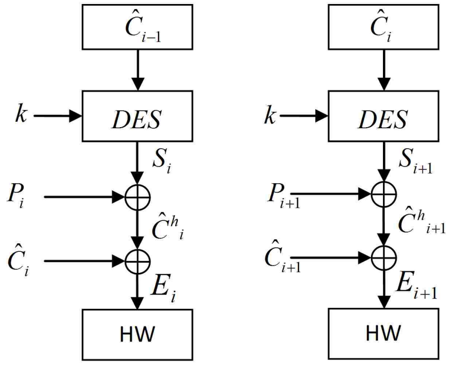


FIGURE 2.2. key verification process for Eve with two consecutive CFB stages

negligible.

$$P_{\mathit{fault}} = 1 - \sum_{i=0}^{\tau} \binom{64}{i} (\eta)^i (1-\eta)^{64-i}. \tag{2.1}$$

In the next step, we need to find the optimum value for N_c. Let K_0^h be the hypothesis when the examined key is wrong and K_1^h when it is right. Then, let A_i be a random variable where $A_i = 1$ defines successful trial at the i^{th} stage that happens when HW of E_i is less or equal to τ, and $A_i = 0$ otherwise. By proper selection of τ, we can make sure that whenever there is no error in the input to the cipher, Eve can recognize the right key. Thus, probability of success event is

$$P_1 = Pr[A_i = 1 | K_1^h] = P[\hat{C}_{i-1}\text{is error-free}] = (1-\eta)^{64}. \tag{2.2}$$

Eve misses the right key when all N_c trials fail that has the probability of

$$P_m = (1 - P_1)^{N_c}, \tag{2.3}$$

We call P_m key missing probability. Thus, we need to find minimum N_c such that keeps P_m below a threshold like T_m.

Now we need to compute the probability that Eve mistakenly admits a wrong key while examining a single candidate. When the used key is wrong due to the avalanche effect, $\hat{C}_i^h$ will have bit error rate of α, that after Xor with $\hat{C}_i$ with bit error probability of η, results in output bit error rate of γ as

$$\gamma = \alpha(1-\eta) + \eta(1-\alpha). \tag{2.4}$$

Since to admit a wrong key at the i^{th} stage as the right one, HW of E_i must be less than τ, the probability of a successful trial at this stage for a wrong key is

$$P_2 = Pr[A_i = 1 | K_0^h] = \sum_{i=0}^{\tau} \binom{64}{i} \gamma^i (1-\gamma)^{64-i}. \tag{2.5}$$

On the other hand, Eve accepts a wrong key when there happens at least one successful trial for it. Thus, the false key probability for a single candidate is

$$P_F = 1 - (1-P_2)^{N_c}, \tag{2.6}$$

Even though this probability seems negligible, it gets aggregated over a large number of examined wrong key candidates, and can result in a non-negligible false key probability, as will be seen in simulations.

2.3.2 Analysis of the Designed Attack Strategy for Eve

In [70] Yin et. al. showed that in noisy environment with bit error rate of η, for linear attack on DES cipher, the probability bias of the new linear equation denoted by $\hat{\varepsilon}$, as well as the success probability of attacker P_s can be computed based on the linear probability bias of the original linear equation ε and the number of obtained pairs by Eve N as

$$P_s = \Phi(2\sqrt{N}\hat{\varepsilon} - \Phi^{-1}(1-2^{-a-1})), \quad \text{where} \quad \hat{\varepsilon} = 2^{u+v}(1-\eta-0.5)^{u+v}\varepsilon. \tag{2.7}$$

If adversary uses the improved linear analysis technique, she needs to use Matsui's linear equation for DES that requires u bits of plaintext and v bits of corresponding ciphertext where $u + v = 26$

to guess $m = 26$ key bits [64]. As discussed in section 1.4, in linear attack with bit advantage of a, the total number of examined keys is 2^{56-a}. If the ciphertexts that Eve obtains are error-free, her success probability will be P_s which is the probability that the correct key is among top 2^{56-a} examined candidates. For the case that obtained ciphertexts are erroneous, the following theorem proven in Appendix 6.1 quantifies the probability that Eve obtains the correct key, fails to get any key and obligatorily drops the whole frame or gets a wrong key.

Theorem 2. *Consider a linear attack with bit advantage of a. Assume Eve's obtained ciphertexts contain bit errors with the rate of η, and that she uses the designed strategy in brute-force step of the linear attack. Then, Eve's total success probability can be computed by*

$$P_c = \frac{P_s(1-P_m)}{P_F 2^{56-a}}[1-(1-P_F)^{2^{56-a}}] \approx P_s(1-P_m), \tag{2.8}$$

where P_m, P_F and P_s are given in Eq.'s (2.3), (2.6) and (2.7). On the other hand, frame erasure probability will be

$$\begin{aligned} P_e &= (1-P_s)(1-P_F)^{2^{56-a}} + P_s P_m (1-P_F)^{(2^{56-a}-1)} \\ &\approx [1-2^{56-a}P_F][1-(1-P_m)P_s]. \end{aligned} \tag{2.9}$$

In addition, the probability that Eve accepts a wrong key denoted by P_w can be derived as $P_w = 1 - P_c - P_e$.

Conclusively, we showed that there is possibility that Eve is not able to obtain any key, or to falsely accepts a wrong key.

2.3.3 Restriction on Eve's Resources

In our secrecy analysis we consider Eve to be restricted in resources with a limited computational capability. In fact, the basis of analysis is that without any restriction, an unbounded adversary would be eventually able to obtain the correct key by examining all possible keys with as many as possible trials. However, when there is a limit on the number of evaluations that Eve can perform or on the number of pairs that she can obtain, she has to restrict the number of key evaluations that consequently impacts her success rate. We assume that adversary is bounded and can not perform more than θ DES evaluations that limits her computational capability.

We also put restriction on the total number of pairs that Eve can use in her cryptanalysis. Let us first consider the scenario where Eve has to gather all these N required number of pairs in her attack from the transmitted data frame throughout which the same key is used for encryption. Now, the question is that for a typical communication link what size this frame needs to have, and how long it takes for Eve in order to accumulate these many pairs? For special case study let us assume that Eve requires $N = 2^{46}$ number of pairs to mount her attack in noisy environment with acceptable success rate. We should note that since according to Eq. (2.7), bias probability for noisy case with $\eta > 0$ denoted by $\hat{\varepsilon}$ is less than the bias for error-free case indicated by ε, to have the same success probability in noisy environment as in the error-free case a higher number of pairs would be required.

First of all, we consider known plaintext attack. Since communication overhead is publicly known, Eve seeks to gather her required number of pairs from these transmitted overhead as plaintext blocks whose corresponding ciphertexts are received. For special case study we consider Internet Protocol version 4 (IPv4) which is a widely deployed protocol that routes most of the traffic in Internet [78]. Considering TCP data transmission, for every payload of 1500 bytes, there will be 20 bytes added IPv4 header and 20 bytes TCP header, i.e. header rate is approximately $h_r = 2.6\%$. In other words, in known plaintext attack 2.6$\%$ of the transmitted packets can be used by attacker to gather her required plaintext/ciphertext pairs. If we consider the plaintext and ciphertext size as the block size in CFB mode with DES cipher which is $l = 64$ bit, there have to be in total $L = Nl/h_r = 1.73 \times 10^{17}$ number of transmitted bits to allow Eve gather $N = 2^{46}$ pairs. For data speed of 100 Mbps, the total amount of time that is required to receive L number of bits by Eve is $T = 481153.8$ hours that amounts to 20048 days assuming that all transmitted data is encrypted using the same key. We can see that for known plaintext attack Eve requires a tremendous amount of time to gather her required pairs from a continuously transmitted frame.

For known ciphertext attack Eve only requires to gather all received ciphertexts that are encrypted blocks from the same data frame. Assuming that she requires $N = 2^{46}$ number of 64-bit ciphertext blocks, for data rate of 100 Mbps the total required time would be 12510 hours equivalent to 521 days. It implies that for both known plaintext and ciphertext attacks, for Eve to gather the required number of pairs in her attack from the transmitted data frame, a huge amount of time is required which is unrealistic. That is why we consider the worst case scenario and assume that there exists a virtual oracle providing Eve with the required number of pairs prior to her attack. In our analysis the maximum number of noisy pairs that Eve is allowed to accumulate is denoted by N_{max}.

2.3.4 Parameter Optimization of Adversary's Attack Strategy

Eave's objective is to mount a successful attack, and in order to achieve this goal, she maximizes the success probability of the utilized linear attack denoted by P_c, given in (2.8), knowing that her computational ability is restricted to θ DES encryptions, and there is a constraint on the number of plaintext/ciphertext pairs that she can accumulate. In linear cryptanalysis with modified exhaustive search phase for noisy environment, Eve runs at most $N_c 2^{56-a}$ DES encryptions, which due to her computational constraints, can not exceed θ. Moreover, we assume that prior to mounting attack on a frame of data, a virtual oracle provides Eve with as many number of pairs as her data storage capability allows denoted by N_{max}. As a result, she needs to design attack parameters including N_c, τ and a, to maximize the overall success probability subject to these constraints

$$\max_{N_c,\tau,a} P_c \qquad \text{subject to } \ \theta \geq N_c.2^{56-a}, \quad N \leq N_{\text{max}}. \tag{2.10}$$

From Eq. (2.8) we can see that to maximize P_c we need to minimize P_m and P_F. Since according to Eq. (2.3), P_m mainly depends on N_c, we can define threshold T_m and find the minimum N_c for which P_m remains below T_m. According to Eq.'s (2.5) and (2.6), to decrease P_F we need to reduce

TABLE 2.2. Parameter optimization algorithm for attack strategy:

```
1- Initialization: put $\tau = 1$, $N_c = 1$.
   Determine $T_m$ and $T_f$ as thresholds for $P_m$ and $P_{\textit{fault}}$
   also $N_{cmax}$ as the maximum value for $N_c$.
2- $\tau \leftarrow \tau + 1$ until $P_{\textit{fault}} > T_f$ and $\tau < 64$
   if $P_{\textit{fault}} \leq T_f$ or $\tau = 64$ go to the next step
3- $N_c \leftarrow N_c + 1$ until $P_m > T_m$ and $N_c < N_{cmax}$
   if $P_m \leq T_m$ or $N_c = N_{cmax}$ go to the next step
4- Compute $a_0 = \lceil 56 - \log_2(\frac{\theta}{N_c}) \rceil$
5- Compute $P_c$ for $a_0 \leq a \leq 56$
   choose $a$ for which $P_c$ has its largest value.
6- Output $\tau$, $N_c$ and $a$ as attack parameters.
```

τ as much as possible. If we define a threshold T_f for P_{fault}, we can find the smallest τ for which P_{fault} remains below T_f. Also, Eve has to choose an optimized value for a to have P_c maximized. The algorithm in Table 2.2, is designed to optimize the linear attack parameters to let Eve achieve the maximum success rate P_c, for a given η. In this algorithm, $P_{\textit{fault}}$ and P_m can be computed using Eq.'s (2.1), (2.3), respectively.

2.4 Main and Wire-tap Channel modeling

In this section, we model main and wiretap channels in block level (with 64-bit input and 64-bit output), using a stationary finite state Markov chain (MC). Since Eve might achieve the right frame key, get a wrong one or even get nothing and drop the whole frame, we also need to model her channel in frame level as a three state memoryless channel.

2.4.1 Main Channel Modeling Using MC

As it was described, the encrypted data goes through a BSC channel with cross over probability of η, created by intentionally introduced noise in application layer. We next model the CFB cipher, channel with deliberate noise and decipher altogether as a single channel, in order to analyze the effect of intentional noise at the output of decipher. Note that we assume there is no degradation in actual physical channel.

Fig. 2.3 illustrates the encryption and decryption structure of CFB mode with DES cipher in the presence of introduced noise to ciphertexts. As shown in this figure, $\{C_i\}$ and $\{\hat{C}_i\}$ are the

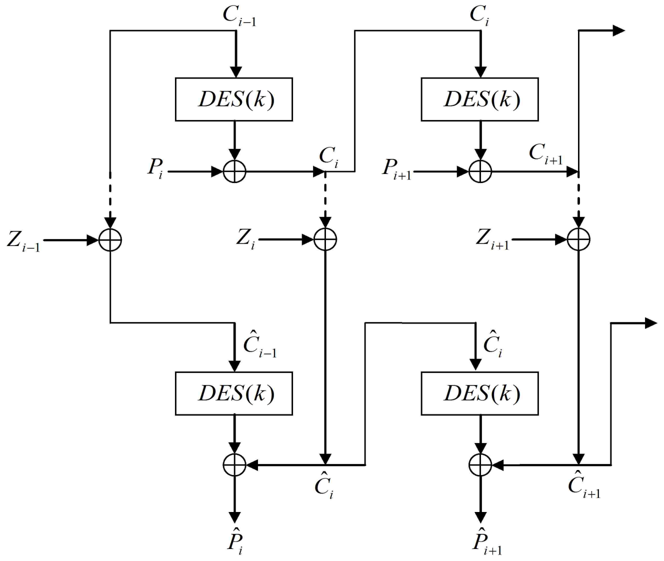


FIGURE 2.3. CFB enciphering and deciphering with channel error

sequences of transmitted 64-bit ciphertext and received noisy ciphertext blocks, respectively, and $\{\hat{P}_i\}$ is the sequence of decrypted blocks at time i for $i = 1, 2, \ldots$. In addition, $\{Z_i\}$ is the sequence of 64-bit blocks of intentional bit errors in channel Z_i^j that are independent and identically distributed with Bernoulli distribution as $Pr[Z_i^j = 1] = \eta$ for $j = 1, \ldots, 64$, such that $\hat{C}_i = C_i \oplus Z_i$. As Fig. 2.3 indicates when $\hat{C}_i$ is noisy, it introduces errors with the rate of η to the decryption output at time i, i.e. $\hat{P}_i$. Moreover, since $\hat{C}_{i-1}$ gets encrypted with DES at time i, due to the avalanche effect, it induces bit error rate of α in $\hat{P}_i$. As a result, to characterize the channel error state in decryption output at time i, it is required to consider errors in both currently received ciphertext $\hat{C}_i$ and the previous one $\hat{C}_{i-1}$. Hence, we need to define four states.

Note that in a particular case when we consider $\alpha = 0.5$, when $\hat{C}_{i-1}$ has error, due to the fact that half of the ciphertext will be in error, errors in $\hat{P}_i$ will be independent from $\hat{C}_i$ and consequently from the error state at time $i + 1$. However, when it has no error, errors in $\hat{C}_i$ will affect both decryption outputs at times i and $i + 1$, and therefore the current state will depend on the previous one. As a result, we have to take all four states into account, each with a different transition probability from the input plaintext block P_i denoted as 64-bit vector X to the output stored plaintext $\hat{P}_i$ denoted by 64-bit vector Y, and let $E = X \oplus Y$ denote the transition error vector.

The channel states are defined as: state S_0, in which there is no error from vector X to the vector Y and happens when there is no error in $\hat{C}_i$ and $\hat{C}_{i-1}$. State S_1, which happens when there is at least one bit error in $\hat{C}_i$, but no error in DES cipher input, $\hat{C}_{i-1}$. State S_2, which shows the situation in which there is at least one bit error in $\hat{C}_{i-1}$ without any error in $\hat{C}_i$. In this channel state, due to the avalanche effect, each bit at the output of DES cipher, flips independently with the probability of α causing bit error probability of α in Y. State S_3, in which both $\hat{C}_i$ and $\hat{C}_{i-1}$ have at least one bit error.

For state S_0 we have $Pr[e_j = 1|S_0] = 0$ and for S_2, $Pr[e_j = 1|S_2] = \alpha$, where e_j denotes the j^{th} bit of E for $j = 1, \ldots, 64$. On the other hand, we should note that in states S_1 and S_3, output bits can not be treated independently because S_1 and S_3 are based on a given condition on the whole 64-bit ciphertext $\hat{C}_i$. Let q denote the probability that there exists at least one bit error in Z_i as

$$q = 1 - (1 - \eta)^{64}. \tag{2.11}$$

The next lemma gives the input-output transition probability for states S_1 and S_3, which is proven in Appendix 6.2.

Lemma 3. *Let X be the input plaintext and Y be the stored plaintext in CFB mode. Assume that the generated ciphertexts go through a channel with error rate of η. We denote the HW of the resulted error vector E with $W(E)$. Then, for state S_1 the input-output vector transition probability will be*

$$Pr(Y|X, S_1) = \begin{cases} \frac{\eta^{W(E)}(1-\eta)^{64-W(E)}}{q} & W(E) \neq 0 \\ 0 & W(E) = 0 \end{cases} \tag{2.12}$$

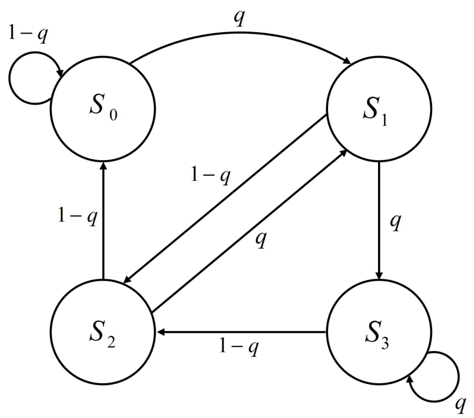


FIGURE 2.4. Alice-Bob channel model as a four state MC

where α is the avalanche bit error rate, and γ is given in Eq. (2.4). The transition probability in state S_3 for all $W(E)$ is

$$Pr(Y|X,S_3) = \frac{\gamma^{W(E)}(1-\gamma)^{64-W(E)} - \alpha^{W(E)}(1-\alpha)^{64-W(E)}(1-q)}{q}. \tag{2.13}$$

Next, we need to find state transition probabilities. For instance, when the state at time $i-1$ was S_2, apparently $\hat{C}_{i-1}$ has been error free, so the only condition required to have state S_0 happen at time i is to receive error free $\hat{C}_i$ which has the probability of $1-q$ that is the transition probability from state S_2 to S_0. Similarly, we can compute other state transition probabilities. Notably, since probability of occurrence of the current state only depends on the previous one, Bob's channel can be modeled as a four state MC that is depicted in Fig. 2.4 with the following state transition probability matrix:

$$T = \begin{bmatrix} 1-q & q & 0 & 0 \\ 0 & 0 & 1-q & q \\ 1-q & q & 0 & 0 \\ 0 & 0 & 1-q & q \end{bmatrix},$$

whose elements demonstrate the transition probabilities between different states. Note that in each state, input plaintexts undergo different channel conditions and error probabilities. In fact, the main channel can only be modeled as a BSC channel in states S_0 and S_2 with cross over probabilities of

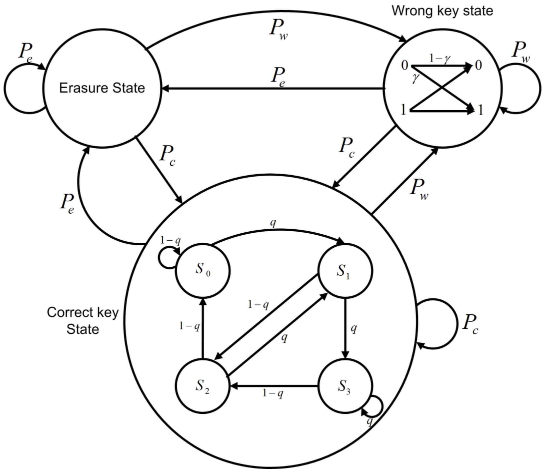


FIGURE 2.5. Eve's hierarchical channel model

0 and α respectively, whereas in other two states it can be modeled based on input-output transition probabilities in (2.12) and (2.13).

In particular, since in MC model for Alice-Bob channel, all four states can be reached from one another, it is an irreducible MC with positive recurrent states [79]. Then, with a supposedly large frame size, MC can reach its stable condition. Since all states are positive recurrent, the set of equations $\mathbf{P}^t\mathbf{T} = \mathbf{P}^t$, and $\mathbf{P}^t.\mathbf{1} = \mathbf{1}$ have a unique solution as $\mathbf{P}^t = [p_0, \ldots, p_3]$ where p_k denotes the steady state probability of state S_k for $k \in \{0, 1, 2, 3\}$ [79]. Where **1** is a 4×1 vector with all elements to be one, and **P** is steady state probability vector (SSPV). By solving this equation set, we get

$$\mathbf{P}^t = \begin{bmatrix} (1-q)^2 & q(1-q) & q(1-q) & q^2 \end{bmatrix}. \tag{2.14}$$

2.4.2 Wire-tap Channel Modeling

When Eve obtains the right key of a frame with the probability of P_c by using optimized verification strategy in linear attack, her decrypted data in that frame undergoes the same channel condition as Bob's. As shown in Fig. 2.5, we refer to this channel state for Eve as the correct key state in frame level which occurs with the probability of P_c and can be modeled as a MC with four channel states in block level. Nevertheless, with the probability of P_e, Eve will not be able to get any key for the attacked frame and has to drop the whole frame. We refer to this state as erasure state.

Moreover, Eve gets a wrong key with the probability of P_w, such that after using a wrong key due to the avalanche effect in DES cipher, each bit in DES output will be independently flipped with the probability of α. This induced error Xors with intentional i.i.d. channel noise that has bit error probability of η. Consequently, in wrong key state, Eve's channel can be modeled as a BSC with cross over probability of γ given in (2.4). Conclusively, wiretap channel is a degraded version of the main channel that only in the correct key state can it be as good as Bob's channel. In fact, Eve's channel behaves like a pseudo two-dimensional Markov Chain (P2DMC) [80] with three memoryless states in frame dimension, each acting like another MC in block dimension as shown in Fig. 2.5.

2.5 Secrecy capacity computation

The next step is to quantify the secrecy capacity of the analyzed security system. The capacity of finite state Markov chains was calculated in [81] and [82]. In [83] and [84] the capacity of the finite state Markov chains with binary symmetric channels associated in each state, was studied. In [85] secrecy capacity of a wiretap channel modeled as a finite state MC is computed. To compute capacities, we assume that the channel states are perfectly known to Bob and Eve in block level, so what we compute is mutual information between the input X and output Y given the current channel state, i.e. $I(X;Y|S_l)$. In frame level, it is assumed that Eve knows the correctness state of each used frame key towards the end of each frame. Specially, this can be considered as the best scenario for Eve, providing us a lower bound for secrecy rate. The main purpose of secrecy capacity computation is to design a secrecy encoder which is applied ahead of the encryption in application layer over multiple frames. Namely, when the message is transmitted at a rate below the secrecy rate to Bob using a Wyner-type encoding technique [25], [28], we can have an arbitrarily small error probability for Bob as well as the maximum entropy for Eve. In the asymptotic sense, by secrecy encoding, users utilize Eve's failures which cause her channel to be a degraded channel compared to Bob's.

2.5.1 Capacity of the Main Channel

When channel state information is available, the capacity is the average of capacities that each one of these MC states contribute to the overall channel capacity [81], [83]:

$$C = \sum_{k=0}^{K-1} p_k C(S_k), \tag{2.15}$$

where $C(S_k)$ is the channel capacity in state S_k in bit per channel use. It can be computed as the maximum information rate between input and output vectors, X and Y, respectively, assuming that the current state S_k is known to Bob:

$$C(S_k) = \max_{P_X} I(X; Y|S_k)/64. \tag{2.16}$$

Note that our modeled four state Markov channel is uniformly symmetric because in any state, channel is output symmetric [81]. For instance, in states S_0 and S_2, the channel behaves as a BSC channel. In states S_1 and S_3, if we define the transition probability matrix as $P_{ij} = Pr(Y = j|X = i, S_l)$ for $i \in \mathcal{Y}, i \in \mathcal{X}, l = 1, 3$, its rows and columns are permutations of each other because according to equations (2.12) and (2.13), its elements only depend on the HW difference of input-output vectors. As a result, also in states S_1 and S_3, the channel is output symmetric. In [81] it is shown that for uniformly symmetric channel in which noise is independent of inputs, like our modeled Markov channel, capacity can be achieved with distribution which is uniform and iid. Accordingly, in this finite state Markov channel by uniformly distributed inputs, the mutual information will be essentially maximized.

In state S_0, channel is error-free with capacity of 1, i.e. $C(S_1) = 1$, and in state S_2, it acts like a BSC with cross over probability of α and the capacity of $C(S_2) = 1 - h(\alpha)$, where h is binary entropy function. However, for S_1 and S_3 in which decryption bit errors are not independent, we need to compute the mutual information between input and output vectors, namely $I(X; Y|S_l)$ for $l = 1, 3$, that is

$$I(X; Y|S_l) = H(Y|S_l) - H(Y|X, S_l). \tag{2.17}$$

We assume that channel state is perfectly known to Bob. In the following theorem which is proven (in Appendix 6.3) using Lemma 3, we compute $H(Y|X, S_l)$ for $l = 1, 3$.

Lemma 4. *Consider our four state MC model for the main channel with input vector X and output vector Y. With equally likely input plaintexts, we can compute $H(Y|X, S_1)$ as*

$$H(Y|X, S_1) = \frac{-1}{q} \sum_{k=1}^{64} \binom{64}{k} \eta^k (1-\eta)^{64-k} . \log \left[\frac{\eta^k (1-\eta)^{64-k}}{q} \right] . \tag{2.18}$$

and $H(Y|X, S_3)$ will be

$$\begin{aligned} H(Y|X, S_3) = & \frac{-1}{q} \sum_{k=0}^{64} \binom{64}{k} . \left[\gamma^k (1-\gamma)^{64-k} - \alpha^k (1-\alpha)^{64-k} (1-q) \right] \\ & . \log \left[\frac{\gamma^k (1-\gamma)^{64-k} - \alpha^k (1-\alpha)^{64-k} (1-q)}{q} \right] . \end{aligned} \tag{2.19}$$

On the other hand, for both states S_1 and S_3, every output vector Y_j can be generated by introducing all possible error vectors over their corresponding input vectors. Hence, since all 64-bit input plaintexts are uniformly distributed, the output will also be equally likely, so for $l = 1, 3$ the output entropy is $H(Y|S_l) = 64$. Thus, by using Eq. (2.17) we can compute the mutual information for states S_1 and S_3 as

$$I(X; Y|S_l) = 64 - H(Y|X, S_l), \quad \text{for} \quad l = 1, 3, \tag{2.20}$$

where $H(Y|S_1, X)$ is given in Eq. (2.18), and $H(Y|S_3, X)$ in Eq. (2.19). According to Eq. (2.16) the channel capacity in states S_l for $l = 1, 3$ will be

$$C(S_l) = \frac{I(X; Y|S_l)}{64} \quad \text{(bits per channel use)}, \tag{2.21}$$

with $I(X; Y|S_1)$ and $I(X; Y|S_3)$ given in Eq. (2.20). We can analyze Alice-Bob channel as a finite state MC with steady state probabilities given in Eq. (2.14). Hence, according to Eq. (2.15) Bob's channel capacity C_B as the average of the state capacities can be computed as

$$C_B = (1-q)^2 + q(1-q)[C(S_1) + 1 - h(\alpha)] + q^2 C(S_3). \tag{2.22}$$

where α is the average bit error rate caused by the avalanche effect. In addition, $C(S_1)$ and $C(S_3)$ are given in Eq. (2.21), implying that these capacities mainly depend on q, γ, α and η . As a result, the main channel capacity depends on q and γ which according to Eq.'s (2.11) and (2.4) are themselves functions of η, for a fixed α. Therefore, Bob's channel capacity mainly depends on the original channel cross over probability η.

2.5.2 Secrecy Capacity of the Wire-tap Channel with Noise

When Eve with the probability of P_c obtains the right key, her channel capacity will be the same as Bob's, i.e. C_B, but when with the probability of P_w gets a wrong key, her channel will turn into a BSC with the cross over probability of γ, which has the capacity of $1 - h(\gamma)$. Note that, the erasure state does not contribute to the capacity. Hence, Eve's capacity will be

$$C_E = P_w(1 - h(\gamma)) + P_c C_B, \tag{2.23}$$

where C_B is given in Eq. (2.22). As discussed in Section 1.3, when the main channel is less noisy than the wiretap channel and the mutual information between Alice at Bob are individually maximized by the the same input distribution, the secrecy capacity can be computed as the difference of two capacities. In our channel model, the first condition holds and only uniformly distributed input maximizes both mutual informations, therefore the secrecy capacity will be $C_s = C_B - C_E$ as:

$$C_s = C_B(1 - P_c) - (1 - P_e - P_c)(1 - h(\gamma)). \tag{2.24}$$

This result implies that secrecy capacity mainly depends on P_c, P_e and C_B. Due to the fact that all P_c, P_e and C_B highly depend on the channel error rate η, the main parameter that impacts secrecy capacity of the system is intentional noise. Namely, if Alice can control the cross over probability of the channel, it is possible to adjust secrecy rate of the system. Note that Alice applies secrecy encoding over multiple frames in order to statistically average over Eve's possible failures in frame level, and also to enable Bob to do the error correction coding when burst of errors occurs. Basically, Alice and Bob has to use a well designed wiretap channel encoder, based on the computed secrecy rate in Eq. (2.24). Notably, the main issue in this scheme is delay that is imposed on the system by applying multiple frame encoding that makes this scheme applicable only for delay tolerant communication.

2.6 Numerical Results

The main objective of numerical analysis is to evaluate the effect of varying η on secrecy rate in order to see if there exists an optimum value for η for which secrecy capacity reaches its maximum. In simulations, we assume that Alice by controlling η is able to generate a degraded wiretap channel. In addition, we assume that the whole data is divided into equal size frames, each containing as many number of 64-bit data blocks as four-state MC reaches its steady state, such that for each frame, encryption and decryption key remains constant.

Let us assume that $\theta = 2^{48}$ is the maximum number of DES encryptions that Eve can perform to establish an attack on each frame. Because for instance, with a CPU having speed of 2.6 GHz, it takes for about 30 hours for her to accomplish these many encryptions. For attack optimization algorithm proposed in section 2.3.4, the initial values selected for n is $n_0 = 20$, maximum possible value for N_c is chosen $N_{cmax} = 100$, and the thresholds T_f and T_m are set to 10^{-5}. Furthermore, we chose α as avalanche effect bit error rate to be 0.5. To evaluate the effect of noise variation on the performance of the system, we changed η from 10^{-4} to 0.05 with 500 steps of size 10^{-4}. Moreover, suppose that Eve is able to detect these step size changes on η by probing the channel and each time is able to optimize all attack parameters using the parameter optimization algorithm. We assume that Eve is not allowed to use more than $N_{\text{max}} = 2^{46}$ number of pairs in her attack.

In Fig. 2.6, overall success probability, wrong key and frame erasure probabilities are depicted as functions of η for fixed number of pairs equal to 2^{46}. As this Figure displays with rising η, P_c is monotonically decreasing, reaching zero for $\eta > 0.017$, while wrong key probability P_w goes to 1 for $\eta = 0.05$ because of increase in P_F. As discussed in section 2.3.1, the obtained

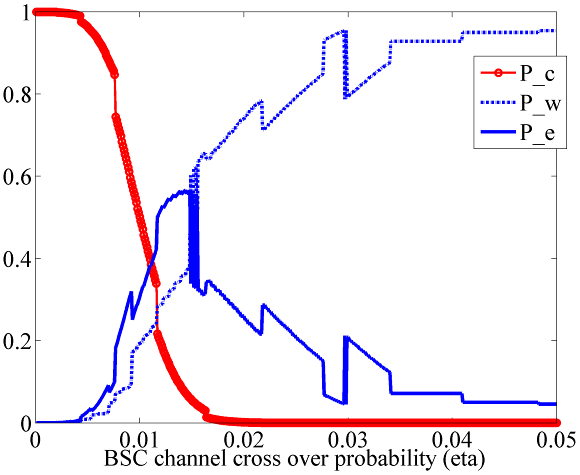


FIGURE 2.6. Overall success probability, frame erasure and wrong key probabilities versus channel cross over probability

results for P_w show that it becomes considerable for some channel conditions and can not be ignored. Moreover, the staircases in these curves occur in η's for which algorithm optimizes and changes attack parameters. In Fig. 2.7 curves of main and wiretap channel capacities as well as the secrecy capacity are drawn as functions of η. This Figure shows that Alice-Bob channel capacity is monotonically decreasing with increase in η while secrecy capacity C_s rises up to its maximum value 0.3442 for $\eta = 0.0125$ and then falls. Indeed, this cross over probability can be considered optimum value for which secrecy capacity achieves its maximum.

TABLE 2.3. Optimized attack parameters using proposed algorithm in subsection 2.3.4

η	0.001	0.005	0.01	0.0125
N_c	5	9	16	20
τ	3	5	6	7
a	23	24	24	27
P_c	0.9999	0.9636	0.5014	0.1618

In Table 2.3 optimized attack parameters using our proposed algorithm for four different η's, i.e. $0.001, 0.005, 0.01$ and 0.0125 are given. According to this table, with increase in η, the required number of trials N_c for each key increases from 5 to 20 in order to keep P_m below the threshold $T_m = 10^{-5}$ when it rises. The same holds for parameters a and τ which to achieve the determined

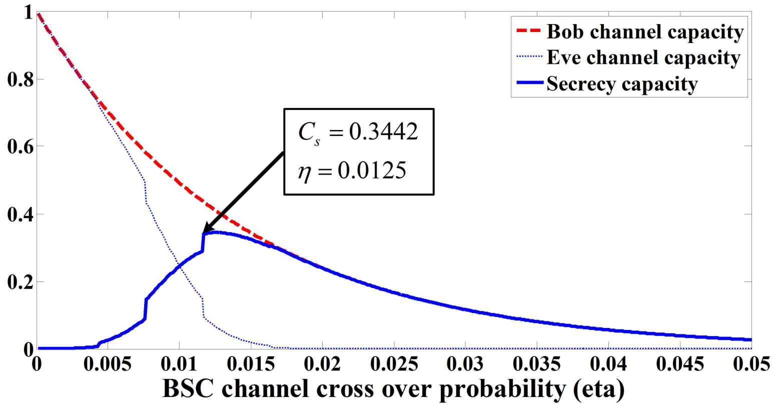


FIGURE 2.7. Main channel and Eve's channel capacities and secrecy capacity for varying channel cross over probability

thresholds, have to increase with rising channel noise to maximize the overall success probability. According to our numerical results, Alice can adjust channel conditions by introducing deliberate noise in application layer to have $\eta = 0.0125$, to achieve the desirable secrecy capacity.

2.7 Conclusion

In this chapter we showed that by introducing tunable noise in application layer upon the encrypted data, even though Eve utilizes an optimized attack strategy, the secrecy rate of the system can remarkably increase. In fact, Alice can achieve a sufficiently large secrecy capacity by adjusting the cross over probability of the channel using deliberate noise. This secrecy rate guarantees a highly secure and reliable communication using wiretap channel coding in application layer over multiple frames. For secrecy capacity computation we tailored the known channel states scenario. In our future work, we will focus on the unknown state case and also will consider a more generic cipher.

Chapter 3
ARQ Based Symmetric-Key Generation over Correlated Erasure Channels

In this chapter based on our work in [16], we focus on the problem of sharing secret keys using Automatic Repeat reQuest (ARQ) protocol. We consider cases where forward and feedback channels are erasure channels for a legitimate receiver (Bob) and an eavesdropper (Eve). In prior works, wiretap channel is modeled as statistically independent packet erasure channels for Bob and Eve. In this chapter, we go beyond the state-of-the-art by addressing correlated erasure events across the wiretap channel. The created randomness is shared between two legitimate parties through ARQ transmissions that will be mapped into a destination set using the first order digital filter with feedback. Then, we characterize Eve's information loss about this shared destination set, due to inevitable transmission errors. This set will be transformed into a highly secure key using privacy amplification in order to intensify and exploit Eve's lack of knowledge. We adopt two criteria for analysis and design of the system: outage probability as a measure of secrecy, and secret key rate as a metric for efficiency. The resulting secrecy improvement is presented as a function of the correlation coefficients and the erasure probabilities for both channels. It is shown that secrecy improvement is achievable even when Eve has a better channel than legitimate receivers, and her channel conditions are unknown to legitimate users.

3.1 Introduction

The broadcast nature of wireless transmissions makes it more vulnerable from security perspective. Traditionally, security can be provided using cryptographic approaches, mainly relying on generation, sharing and renewing of secret keys [33]. However, key management is deemed quite challenging in wireless networks. Maurer et.al. in [56] considered information theoretic key agreement in noisy communication channel based on common randomness and public discussion. They have defined secret key rate as the maximal achievable rate at which secret key can be generated by legitimate partners (Alice as transmitter and Bob as receiver) about which an eavesdropper (Eve) has virtually no knowledge.

Among physical layer based key management techniques, some have utilized the well known ARQ protocol to facilitate exchange of secret keys between Alice and Bob [53], [54]. In [12] authors have proposed using ARQ mechanism to generate secrets by taking advantage of Eve's inevitable information loss due to transmission errors. In this approach, dynamic secrets are extracted from created common randomness using universal class of hash functions [55]. However, in all of these works feedback channel is assumed to be error-free which is not satisfied in mobile radio environment. In this work, we consider a key management scheme similar to [12], and characterize a two-way communication channel model where feedback channel is assumed to be a Binary Erasure Channel (BEC). Previously, in ARQ communications, feedback transmission was also modeled as erasure channels [86],[87].

In all of these schemes, it is assumed that erasure events for Bob and Eve are statistically independent. However, in real radio communications, there could be correlation between channels from a transmitter to different receivers depending on the availability of line-of-sight, physical deployment of the receiver antennas and the presence or absence of scatterers [88]. In [89] information loss in terms of reduction in secrecy capacity due to the correlation in wiretap channel is quantified. In [90] the effects of correlation between packet erasures at Bob and Eve on the performance of LDPC based secrecy coding scheme was addressed.

Our work lies in a different category than the works in [1], [9], [89] that rely on secrecy capacity measure nor do we design specific codes for correlated wiretap channel as [90]. This work is based on Maurer's work [56] where key distilling problem from common randomness is studied. In cryptography community this problem is addressed based on extracting strong security form a weakly secure source that is common between two parties [5]. The main goal in this area is to increase generation rate of a sufficiently secure key.

In this work a key scheduling algorithm based on ARQ transmission mechanism used in [12] is revisited, analyzed more thoroughly, and further modified to address more challenging technical issues such as synchronization and correlation. The key contributions can be summarized below:

- One of the main issues in ARQ mechanism used to generate shared randomness is synchronization. We show that even with erasure feedback channel, synchronization between Alice and Bob in selection of a random body of transmitted data, called One-Time-Frame (OTF) set, can be guaranteed using the proposed reconciliation protocol.

- For performance analysis we design an optimized attack strategy based on binary hypothesis testing [91] allowing Eve to estimate this common randomness.

- We design a digital filter based mapping and apply it over OTF set to generate a destination set constituting shared random data between legitimate users. By using this mapping strategy Alice and Bob can take advantage of possible mistakes in Eve's decisions due to transmission errors in order to cause further information loss for her. This lack of knowledge, will next be manipulated by applying privacy amplification to establish secure keys.

- In our correlated wiretap channel model we consider correlation between erasures in main and eavesdropper's channel and then analytically and quantitatively study its negative influence on both secrecy and efficiency of the designed scheme. We study the trade-off between secrecy measured in terms of secrecy outage rate and efficiency in terms of secret key rate and design system parameters to achieve the required secrecy and efficiency.

In simulations, evaluation of the achieved secrecy shows that almost for all channel conditions the required security enhancement can be attained, even when erasures are correlated and Eve has a better channel than that between legitimate users. Simulations also demonstrate that even in unknown wiretap channel condition a good secrecy is achievable.

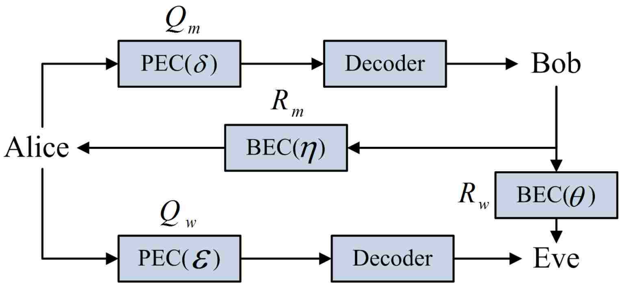


FIGURE 3.1. Erasure forward and backward channel model for wiretap channel

This chapter is organized as follows. Correlated wiretap channel model is illustrated in section 3.2, and reconciliation strategy is explained in section 3.3. The proposed attack strategy for Eve and its analysis is presented in section 3.4 followed by description of the mapping strategy in section 3.5. In Section 3.6 we analyze the performance of the designed system in terms of secrecy and efficiency. Numerical and simulation results are illustrated in section 3.7. We conclude this chapter in 3.8. Proofs are provided in Appendix.

3.2 Correlated Channel Model

We consider the wiretap channel with memoryless packet erasure channel (PEC) model, where erasures for Bob and Eve are correlated. In our model, ARQ is added for authenticated users as shown in Fig. 3.1. We use frame structure where M number of packets, encrypted using the same symmetric key and then encoded according to a specific encoding rule, will be encapsulated into a frame. Alice transmits these packets over the main channel Q_m to an intended recipient called Bob. Across Q_m packet erasures occur with probability δ. Bob is permitted to request retransmission of any missing packets up to K times using a feedback channel R_m. When he decodes a packet correctly sends back a bit 1 as an ACK, otherwise returns a bit 0 as a NACK. Alice receives these feedback bits through R_m modeled as a BEC with bit erasure probability η.

Eve as a passive eavesdropper observes transmitted or retransmitted packets through a wiretap channel Q_w modeled as a PEC with packet erasure probability ε. She is supposedly aware of the decoding rule and is also able to observe feedback messages through a backward wiretap channel R_w where bit erasures occur with probability θ. Since Q_m and Q_w are memoryless, erasures occur independently within each channel. However, packet erasures between two channels are correlated with correlation coefficient ρ. We define two Bernoulli random variables E_m and E_w with values in the set $\{0, 1\}$, where one indicates erasure and zero indicates correct reception of a packet at one-time transmission. Hence, $Pr(E_m = 1) = \delta$ and $Pr(E_w = 1) = \varepsilon$. Let $p_{ij} = Pr(E_m = i, E_w = j)$. Then, $\delta = p_{10} + p_{11} = \mathbb{E}[E_m] = \mathbb{E}[E_m^2]$ and $\varepsilon = p_{01} + p_{11} = \mathbb{E}[E_w] = \mathbb{E}[E_w^2]$. Pearson correlation coefficient between random variables E_m and E_w can be written as [92], [90]

$$\rho = \frac{Cov(E_m, E_w)}{\sqrt{Var(E_m)Var(E_w)}} = \frac{p_{11} - \delta\varepsilon}{\sqrt{\delta\varepsilon(1-\delta)(1-\varepsilon)}}. \tag{3.1}$$

We should note that given a value for δ and ε, ρ can not take every value in the interval $[0, 1]$ and will be bounded by the functions of erasure probabilities. By considering that $\delta = p_{11} + p_{10}$ and $\varepsilon = p_{11} + p_{01}$, and the fact that $\sum_{i,j=0}^{1} p_{ij} = 1$ where $p_{ij} > 0$, we can get the following bounds for ρ

$$\frac{\max(\delta + \varepsilon - 1, 0) - \delta\varepsilon}{\sqrt{\delta\varepsilon(1-\delta)(1-\varepsilon)}} \leq \rho \leq \frac{\min(\delta, \varepsilon) - \delta\varepsilon}{\sqrt{\delta\varepsilon(1-\delta)(1-\varepsilon)}}. \tag{3.2}$$

If we define Bernoulli random variables e_m and e_w for erasure events in feedback channels R_m and R_w, respectively, we will have $Pr(e_m = 1) = \eta$ and $Pr(e_w = 1) = \theta$. Let $q_{ij} = Pr(e_m = i, e_w = j)$. Then, across feedback channels these bit erasures are correlated with correlation coefficient of

$$\psi = \frac{Cov(e_m, e_w)}{\sqrt{Var(e_m)Var(e_w)}} = \frac{q_{11} - \eta\theta}{\sqrt{\eta\theta(1-\eta)(1-\theta)}}. \tag{3.3}$$

Similar to ρ, there also exist bounds for ψ. Finally, we have

$$p_{11} = \rho\sqrt{\delta\varepsilon(1-\delta)(1-\varepsilon)} + \varepsilon\delta, \qquad q_{11} = \psi\sqrt{\eta\theta(1-\eta)(1-\theta)} + \eta\theta. \tag{3.4}$$

3.3 Reconciliation Strategy

In this key management scheme only packets that are decoded correctly for the first transmission and their corresponding feedbacks are received error-free by Alice would be selected to be in OTF set. Once the number of packets in the collected OTF reaches the threshold n_{ts}, they will stop putting packets into it. The main purpose of reconciliation step is to make sure that legitimate users have no disagreement upon this randomly selected body of transmitted data. The next step is to apply a mapping strategy to generate a destination set that will be next used to extract secret keys by applying a mutually agreed universal hashing function over it. Each packet format contains three important fields: a retransmission flag that is set to 1 by Alice when a packet is retransmitted to let Bob know that it does not belong to OTF, a unique sequence number assigned to each packet, which is the sequence number of the previous packet in the frame incremented by one, and a dropping flag used for synchronization purposes.

In this scheme, we use Stop and Wait protocol (SW), that requires Alice to wait for the response from Bob, which is the feedback message represented by a bit belonging to the set $\{0, 1, e\}$. Whenever Alice receives ACK, represented by bit 1, she finds out that a new packet has to be transmitted, but once she receives a NACK feedback, represented by bit 0, she realizes that the packet has to be retransmitted, thereby suggesting that it is not in OTF. The erased bit e represents the case when Alice has not received the feedback message at the required time interval. In this protocol, if the current packet is received correctly at first transmission, and the next received packet is a new one with a different sequence number, the receiver can identify that the current packet belongs to OTF. Each packet can be retransmitted at most K times to make it more likely for Bob to correctly decode it. If no ACK is received within K retransmissions, Alice drops the packet.

One of the main problems in this algorithm is OTF synchronization issue because there is possibility of discrepancy between Alice and Bob. For instance, assume that Bob has received a packet correctly in the first transmission, yet ACK has not gone through the backward channel in any of its retransmissions. Since Alice has not received any ACK, she will decide to drop the packet and transmit a new one. Next, Bob receives a packet with a different sequence number, leading him to put the previous packet into OTF. We include a dropping flag in each packet to avoid such problems which is set to one for a packet when the number of consecutively dropped packets prior to it is odd, and zero otherwise.

TABLE 3.1. Denotations

i	The i^{th} correctly received packet by Bob
k_i	The Reception time for packet i
P_i	The assigned sequence number to packet i
F_i	The corresponding feedback of packet i
SR_i	The retransmission flag sign associated with packet i
SD_i	The dropping flag sign associated with packet i

Suppose that at the beginning of each frame, the timers in both sides launch and increments by one by each packet transmission. Consider the denotations in Tab. 3.1. Let the next correctly received packet $i+1$ arriving at time k_{i+1} have the sequence number of $P_{i+1} = P_i + j$ and the dropping flag sign of SD_{i+1}. Therefore, Bob realizes that there were $j-1$ dropped packets within the time interval $[k_i, k_{i+1}]$. Whenever $j-1$ is odd and $SD_{i+1} = 0$, or $j-1$ is even and $SD_{i+1} = 1$, he finds out that packet i is dropped and does not belong OTF. The pseudo-codes for Alice and Bob's OTF packets selection strategies are presented in Tab.'s 3.2, 3.3. Alice puts a packet i into OTF if at first transmission, the received feedback $F_i = 1$. On the other side, from SR_i, Bob can realize that it is not a retransmission, and also by observing P_i, P_{i+1} and SD_{i+1} she finds out it is not dropped and belongs to OTF.

TABLE 3.2. Alice's OTF strategy

```
If packet $i$ is transmitted more than once, set $SR_i = 1$ and $i \notin$OTF
Else set $SR_i = 0$ and wait for feedback $F_i$
    If $F_i = 1$ put packet $i$ into OTF, Else $i$ is not in OTF
```

TABLE 3.3. Bob's OTF strategy

```
If packet $i$ is received correctly, check $SR_i$
    If $SR_i = 1$, $i$ is not in OTF, Else check $P_{i+1}$
        If $P_{i+1} = P_i + j$ for $j \neq 0$, then
            If $j$ is odd (even) and $SD_{i+1} = 0(1)$, put $i$ into OTF
            Else, $i$ is not in OTF
```

When Alice and Bob make decisions based on these strategies, it can be guaranteed that their synchronization error on OTF set is zero, and both completely agree on n_{ts} OTF packets that later on will be used as a basis to establish secret keys. As a result, packet that are received correctly with probability $1-\delta$ and their feedbacks are received correctly with probability $1-\varepsilon$, will be in common OTF set with the probability of

$$P_c = (1-\delta)(1-\eta). \tag{3.5}$$

3.4 Eve's attack strategy and its performance

Even though Eve is able to eavesdrop retransmissions as well as feedback messages, unlike Alice and Bob, she is not certain of synchronization with users. In fact, that is because her transmission errors are partially independent from the errors in the main channel, and she is unable to directly communicate with the transmitter, or for instance ask for retransmission as Bob does. As a result, she has to determine a strategy to make decisions based on the eavesdropped data.

Let i_E indicate a packet that Eve has received correctly with sequence number P_{i_E}, associated feedback message F_{i_E} and retransmission flag SR_{i_E}. Let also i_E+1 denote Eve's next correctly received packet. Note that to decide which packets are in OTF, Eve has to make the best use of her obtained information about these packets. There are some cases that help Eve confidently know what exactly users did with the packet i_E. For instance, when SR_{i_E} is one, or F_{i_E} is zero, she can ascertain that packet i_E does not belong to OTF.

In other cases where $P_{i_E+1} \neq P_{i_E}$, Eve has to make a guess about packet i_E based on her main observation which is the feedback message, F_{i_E}. In this scheme Eve uses binary hypothesis testing based on Maximum A-Posteriori Probability (MAP) rule [91] as her strategy in distinguishing OTF packets. Let H_1 be the hypothesis that packet i is in Alice and Bob's OTF and H_0 otherwise. Assuming that packet i_E is the same packet i which is simultaneously received by Bob, according to the MAP decision rule, for the received feedback $F_{i_E} \neq 0$ by Eve, she decides that packet i_E belongs to OTF set if

$$Pr[H_1|F_{i_E}, E_w^i = 0] > Pr[H_0|F_{i_E}, E_w^i = 0], \tag{3.6}$$

E_w^i indicates the random variable E_w associated with one-time transmission of packet i_E, i.e. $E_w^i = 0$ means packet i_E is received correctly by Eve. The following theorem with the provided proof in Appendix 6.4 gives us a more explicit idea about Eve's decision rule.

Theorem 3. *Assume that Eve makes a decision based on the MAP rules in Eq. (3.6). Then, for a correctly received packet when she receives feedback* 1*, she makes a decision in favor of* H_1 *if* $\Gamma > 0$*, where* Γ *is defined as*

$$\Gamma \triangleq 1 - 2\eta - \theta + 2\psi\sqrt{\eta\theta(1-\eta)(1-\theta)} + 2\eta\theta. \tag{3.7}$$

On the other hand, when she receives an erased feedback, she makes decision in favor of H_1 *if* $\Lambda > 0$ *which is defined as*

$$\begin{aligned}\Lambda \triangleq 2\left[(1-\delta)(1-\varepsilon)\theta + \rho\theta\sqrt{\varepsilon\delta(1-\varepsilon)(1-\delta)}\right]. \\ \left[(1-\eta)(1-\varepsilon)\theta - \psi(1-\varepsilon)\sqrt{\eta\theta(1-\eta)(1-\theta)}\right] - (1-\varepsilon)\theta.\end{aligned} \tag{3.8}$$

Accordingly, the pseudo-code for Eve's attack strategy in distinguishing OTF packets is presented in Table 3.4.

TABLE 3.4. Eve's Attack strategy

```
If packet $i_E$ is received correctly, check $SR_{i_E}$
If $SR_{i_E} = 1$, then $i_E$ is not in OTF, Else, wait for feedback $F_{i_E}$
    If $F_{i_E} = 0$, then $i_E$ is not in OTF
    Else if $P_{i_E+1} \neq P_{i_E}$, then
            If $F_{i_E} = 1$ and $\Gamma > 0$, put $i_E$ into OTF
            Else if $F_{i_E} = e$ and $\Lambda > 0$, put $i_E$ into OTF
    Else $i_E$ is not in OTF
```

In order to analyze Eve's performance, we need to investigate how much discrepancy her OTF has with the actual one, namely with what probability, she misses an OTF packet, called OTF missing probability P_m, or chooses a non-OTF packet, called false OTF probability P_F. P_m is the probability that given hypothesis H_1 has occurred for packet i_E, Eve does not choose it as an OTF packet. P_F is the probability that given hypothesis H_0, Eve puts i_E into OTF. In Lemma 5, whose proof is given in Appendix 6.5, we compute these probabilities.

Lemma 5. *In our scheme, if Eve uses the proposed attack strategy, she misses one OTF packet with the probability of*

$$\begin{aligned} P_m =& \boldsymbol{1}_{(\Gamma<0)} \frac{(1-\eta-\theta+q_{11})(1-\delta-\varepsilon+p_{11})}{(1-\delta)(1-\eta)} \\ &+ \boldsymbol{1}_{(\Lambda<0)} \frac{(\theta-q_{11})(1-\delta-\varepsilon+p_{11})}{(1-\eta)(1-\delta)} + \frac{\varepsilon-p_{11}}{1-\delta}. \end{aligned} \tag{3.9}$$

Moreover, she puts a wrong packet into OTF with probability

$$P_F = \boldsymbol{1}_{(\Lambda>0)} \theta(\delta-p_{11}) + \left[\boldsymbol{1}_{(\Lambda>0)} q_{11} + \boldsymbol{1}_{(\Gamma>0)}(\eta-q_{11})\right](1-\delta-\varepsilon+p_{11}). \tag{3.10}$$

where $\boldsymbol{1}_{\mathcal{A}}$ is the indicator function, which is equal to 1 when $\mathcal{A}$ holds. p_{11} and q_{11} are provided in Eq. (3.4).

3.5 Eve's Misalignment and OTF Mapping strategy

Whenever Eve has a miss-detection, by missing a packet or putting a wrong packet into OTF, assuming that her next OTF packets are selected correctly, her gathered OTF set respectively moves one packet size backward or forward compared to the original set. Hereafter, she loses her OTF alignment with Alice and Bob, and in order to realign with the users, she has to have the same number of OTF missing events as the false OTF packets. However, If Alice and Bob take a strategy by mapping OTF into a destination set where once a misalignment occurs, the resulted error propagates to upcoming packets, any miss-detection for Eve would be equivalent to missing the rest of the transformed data.

A possible mapping strategy is a simple digital filter with a delayed feed back. Let X_i and W_i denote respectively the i^{th} packet in the original OTF and in the destination set, where $i = 1, \ldots, n_{ts}$. After applying this transformation, whose block diagram is depicted in Fig. 3.2, W_i will be the result of Xor of X_i and W_{i-1}. Note that only the random body of each OTF packet

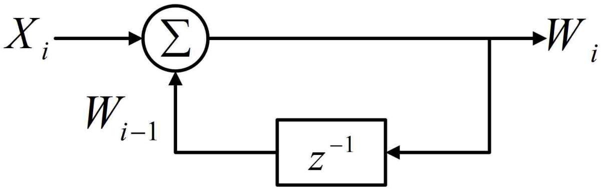


FIGURE 3.2. Block diagram of the simple digital filter used for mapping OTF set

will be used in this mapping. Let M_{max} be the maximum possible number of packets within the frame. If each packet has size n_b, by excluding the sequence number as well as two bit flags, only $n_r = n_b - \log_2 M_{max} - 2$ bits of each packet will be transformed, so X_i's have size n_r, and the generated destination set will be of size $n = n_{ts} n_r$.

TABLE 3.5. Alice-Bob and Eve's OTF and destination sets

OTF$_{AB}$	X_2	X_3	X_4
OTF$_E$	X_3	X_3'	X_4
DS$_{AB}$	$X_1 \oplus X_2$	$X_1 \oplus X_2 \oplus X_3$	$X_1 \oplus X_2 \oplus X_3 \oplus X_4$
DS$_E$	$X_1 \oplus X_3$	$X_1 \oplus X_3 \oplus X_3'$	$X_1 \oplus X_3 \oplus X_3' \oplus X_4$

Consider a simple case when the number of packets within OTF set is $n_{ts} = 4$. In Tab. 3.5 Alice and Bob's OTF as OTF$_{AB}$ and Eve's OTF as OTF$_E$ (starting from the second packet) are illustrated when Eve misses X_2 and has a false event by choosing X_3'. In this case even though OTF$_E$ has missed its alignment at the second packet, it realigns with OTF$_{AB}$ at X_4 resulting in only two packet discrepancies between them. The resulted destination set for legitimate users as DS_{AB} and for Eve as DS_E are given in Tab. 3.5. We assume that X_i's are generated uniform randomly, so for instance for the third and the fourth packets in DS_E, $X_2 \oplus X_3$ behaves like an additive noise with error rate of 0.5. That is why when a misalignment occurs, for the remaining packets in DS_E, missed or false OTF packets act like additive noise to further deceive Eve. In other words, every miss-detection causes an uncertainty for her that accumulates in upcoming packets, resulting in a larger uncertainty for Eve in her destination set. In general, when there is a miss-detection at j^{th} packet, by utilizing the suggested mapping strategy, any realignment for Eve becomes highly unlikely, and it can be guaranteed that there will be errors in the rest of $n_{ts} - j$ packets of Eve's destination set.

3.6 Secrecy Scheme Design and Analysis

Throughout transmission of each frame by using the ARQ protocol and mapping strategy, Alice and Bob will generate a destination set upon which they both completely agree. When a function is chosen uniform-randomly from a universal class of hash functions, regardless of what distribution the actual input has, for sufficiently short output, the expected hash output will have a distribution close to uniform with maximum entropy. By the last packet of the frame, Alice will transmit this chosen function to Bob that will be applied over the produced destination set to extract secret keys, later on being used as a symmetric key for encryption of the next frame. As a result, for a short hash output they can make sure that Eve, given her knowledge, gets arbitrarily negligible information about it.

In order to analyze the designed secrecy scheme, we define appropriate metrics whereby the required secrecy and efficiency for the system can be regulated. We define outage probability as the probability that the aimed information theoretic secrecy is not achieved, based on which system parameters will be designed. Furthermore, we use secret key rate to measure secrecy throughput and efficiency of the scheme.

3.6.1 Outage Probability based on a New Oracle Model

In [55], the additional information that a virtual oracle freely gives Eve is considered as an auxiliary random variable that simplifies secrecy analysis for privacy amplification. Assume that a virtual oracle freely informs Eve that in which packet she first missed her alignment with Alice-Bob OTF. Let this packet be the $N_c + 1^{\text{st}}$ OTF packet, so that Eve knows with a high probability she has observed N_c packets, with length $t = N_c n_r$ denoted by V, correctly from the actual destination set W. Nonetheless, she will have error propagation in the remaining packets because of using the proposed mapping strategy. Eve can not correct her mistake by using this additional information because she has no idea what kind of miss-detection has occurred or what happened after this misalignment. Literally, the secrecy that system obtains in the presence of this oracle provides a lower-bound of the actual secrecy that scheme could have gained without giving such a privilege to Eve.

Let $V = e(W)$ and function $e : \{0,1\}^n \longrightarrow \{0,1\}^t$ be an arbitrary eavesdropping function, with $t < n$, where $n = n_{ts}n_r$ is the length of the input string W. Alice and Bob arbitrarily choose a function G from a universal class of hash functions, mapping $\{0,1\}^n$ into $\{0,1\}^r$, and then apply it over W to get a secret key Q of size r, where $r = n - t - s$. According to Lemma 1 (corollary 4 in [55]), Eve's expected information about the secret key, given G and V, satisfies $I(Q; G, V) \leq 2^{-s}/\ln 2$. As information theoretic secrecy goal, if we require the upper-bound of $I(Q; GV)$ to be I_{sup}, the necessary s is

$$s = -\log[\ln(2) I_{sup}], \tag{3.11}$$

for logarithm of base 2. But $n-t = r+s$ is the length of the input string after misalignment. Hence, for the required s and given r, the minimum required number of packet discrepancies between two sets denoted by l has to be

$$l = \left\lceil \frac{n-t}{n_r} \right\rceil = \left\lceil \frac{r+s}{n_r} \right\rceil. \tag{3.12}$$

Consequently, if we design the system in a way that with a high probability misalignment in OTF set happens at one of the first $n_{ts} - l + 1$ OTF packets, we can make sure that after mapping, it is very likely to have the number of different packets between V and W, denoted by N_e, be more than l. We define outage probability as the probability that $N_e < l$, which actually is the probability that determined secrecy goal as $I \leq I_{sup}$ is not satisfied. The following Theorem, proven in Appendix 6.6, provides an upper-bound for outage probability.

Theorem 4. *Let secrecy outage P_{out} be the probability that there exists less than l packet discrepancies between Eve's destination set and the actual set. For the proposed secrecy scheme, P_{out} is upper-bounded as*

$$P_{out} \leq \left[\frac{(1-P_m)P_c}{1-(1-P_c)(1-P_F)} \right]^{n_{ts}-l+1}, \tag{3.13}$$

where n_{ts} is the number of packets in OTF. P_m, P_F and P_c can be computed using Eq.'s (3.9), (3.10) and (3.5).

Note that in our analysis we will consider the worst case scenario where equality in Eq. (3.13) holds. Now we can determine the minimum average uncertainty that Eve has about the generated secret key. Let W be a random n-bit string with uniform distribution over $\{0,1\}^n$, and V be the random variable indicating what Eve observes correctly form W with the help of the oracle. Let us define P_{out} as the probability that the length of V is larger than t bits for some $t < n$, and let $s < n-t$ be a positive safety parameter, such that $r = n-t-s$. With the probability $1-P_{out}$, V will take on values of v that belong to the set $\mathcal{A}_v$ constituted of subsets of W with less than or equal to t bits. In this case, as the most optimistic scenario for Eve, she will know t bits correctly out of W. If Alice and Bob choose G as their universal hashing function from $\{0,1\}^n$ to $\{0,1\}^r$, according to corollary 5 in [55] her information about the secret key $Q = G(W)$ with length r will be upper-bounded as $I(Q; G, V = v) < 2^{-s}/\ln 2$ or in other words $H(Q|G, V = v) \leq r - 2^{-s}/\ln 2$. Since this holds for every $v \in \mathcal{A}_v$, by statistical averaging over $\mathcal{A}_v$, Eve's average entropy about Q given G and V will be lower-bounded as

$$\begin{aligned} H(Q|GV) \geq \sum_{v \in \mathcal{A}_v} P_v(V = v) H(Q|G, V = v) = Pr(\mathcal{A}_v)[r - 2^{-s}/\ln 2] \\ = (1 - P_{out})[r - 2^{-s}/\ln 2] \geq (1 - P_{out})r - 2^{-s}/\ln 2 \qquad (3.14) \end{aligned}$$

For W with the length of $n_{ts}n_r$, and $t = (n_{ts} - l)n_r$ bits, we can replace P_{out} with its upper-bound in Eq. (3.13) to consider the most pessimistic scenario.

3.6.2 Secret Key Rate

The next step is to quantify and analyze efficiency of the designed secrecy system in terms of secret key rate. First of all, we need to design system parameters including the size of OTF set and data frame, to guarantee that the system is sufficiently secure. As will be described later, these are two parameters that mainly affect efficiency of the system. In order to maintain a large uncertainty for

Eve, according to Eq. (3.14), we need to have s large enough and P_{out} as small as possible. If s is chosen based on the determined I_{sup} in Eq. (3.11), with outage probability sufficiently close to 0, we can have a highly likely secure system, with Eve's average entropy close to maximum. The number of packets in OTF, n_{ts}, can be lower-bounded accordingly to have outage probability stay below a threshold T_{out} chosen to be sufficiently small, i.e. $P_{out} < T_{out}$

$$n_{ts} = \left\lceil (l-1) + \frac{-\log T_{out}}{\log P_d - \log[(1-P_m)P_c]} \right\rceil, \text{ where } P_d = 1-(1-P_c)(1-P_F). \qquad (3.15)$$

n_{ts} only takes integer values, and l is obtained by Eq. (3.12). Note that $-\log(T_{out})$ is positive.

We also need to have enough number of packets within each frame to make sure that the number of OTF packets reaches to the threshold n_{ts}. The probability that a packet is in OTF is P_c. The total number of packets being in OTF out of M packets has binomial distribution with parameter P_c. We call the probability of having at least n_{ts} OTF packets within M packets, success probability and denote it by P_s. In order to have enough number of packets within OTF set with a high probability, we can choose a threshold T_s sufficiently close to 1 and determine the smallest M for which

$$P_s = \sum_{k=n_{ts}}^{M} \binom{M}{k} P_c^k (1-P_c)^{M-k} \geq T_s. \qquad (3.16)$$

Clearly, with increase in n_{ts} the required number of packets in a frame, i.e. M, goes up.

There is an outage probability $1-P_s$ that the number of OTF packets does not reach to the required threshold n_{ts}. When such an outage occurs, Alice and Bob can use the existing OTF packets to complete OTF set. Suppose that Alice has already finished transmission of the whole frame but the created OTF set still lacks h number of packets. In this case, since they both agree on the $n_{ts}-h$ accumulated OTF packets, one possible alternative would be OTF refilling protocol which divides the existing OTF into h partitions with equal size of $\lfloor \frac{n_{ts}-h}{h} \rfloor$ packets and then selects one packet out of each subset in order to refill the remaining h vacant positions. Note that rarely does this outage event occur for a well designed system, and hence its overall effect on Eve's knowledge will be negligible.

Secret key rate is the maximal rate $R > 0$ such that for every $\alpha > 0$, there exists a public communication over an insecure but authenticated channel, over which Alice and Bob who agree upon a random data can generate keys Q and Q' respectively, where $Q = Q'$ with probability at least $1 - \alpha$. Also, $I(Q; V) \leq \alpha$, and $H(Q)/N \geq R - \alpha$, where V is data observed by Eve, and N is the number of channel uses [56]. In our secrecy scheme, Alice and Bob both agree on a random data called destination set by using reconciliation protocol and mapping strategy, then they transform it into the secret key Q of length r which is the same for both of them. Moreover, according to Eq. (3.14) since $H(Q) = r$, we can compute Eve's information about the key Q given her knowledge G, V as $I(Q; G, V) = H(Q) - H(Q|G, V) \leq 2^{-s}/\ln 2 + P_{out}r$. Namely, design of a system with a very low outage probability and sufficiently large s results in a negligible key information for Eve. As a result, we achieved the required public transmission and can compute secret key rate as the length of the generated hash value over the total transmission cost which is the number of channel uses including retransmissions.

Assume that for the designed key generating ARQ protocol, due to throughput requirements the maximum number of allowed retransmissions per packet is set to be K. In our scheme, given that a packet is received correctly, the probability that it is transmitted for R times with $1 \leq R \leq K+1$ is $P_c(1 - P_c)^{R-1}$. On the other hand, not being received correctly by Bob implies that the packet was transmitted for $K + 1$ times. It is straightforward to show that the average number of trials per packet denoted by μ_r is

$$\mu_r = \frac{1 - (1 - P_c)^{K+1}}{P_c}. \tag{3.17}$$

When M is fixed and also sufficiently large, by the Strong Law of Large numbers (SLL), the total number of transmissions denoted by R for M packets in the frame will be $M\mu_r$. For n_b as the number of bits per packet, the number of channel uses is Rn_b bits. Since secret key rate is the ratio of the generated key entropy over all channel uses, it can be obtained as

$$R_s = \frac{H(Q)}{N} = \frac{r}{Rn_b} = \frac{rP_c}{[1-(1-P_c)^{K+1}]Mn_b}. \tag{3.18}$$

It should be noted that when to meet the secrecy requirements, M is chosen to be the minimum possible value for which Eq. (3.16) is satisfied, R_s gives us the maximum achievable key rate.

To study the trade-off between secrecy and efficiency of the system, we evaluate system performance in various settings of design parameters. If it is required to have a higher information theoretic secrecy meaning that a lower upper-bound for Eve's information about the key, i.e. I_{sup}, is mandated, Eq.'s (3.11) and (3.12) show that higher s and l are needed. However, a system that is designed to guarantee a higher discrepancy between Bob and Eve turns out to have a lower secret key rate and a larger secrecy outage rate. That is because with decrease in the exponent of Eq. (3.13) due to the increase in l since its base is less than 1, P_{out} ascends, whereas according to Eq. (3.15) with increase in l, n_{ts} and consecutively M go up that brings about a lower R_s based on Eq. (3.18). Accordingly, the threshold I_{sup} should be precisely determined, otherwise unnecessarily low I_{sup} can negatively affect both secrecy and efficiency.

If for a fixed channel condition, and specified I_{sup} and r resulting in a fixed l, the system designer tailors to a higher secrecy or a lower secrecy outage rate by regulating a lower outage threshold T_{out}, according to Eq. (3.15), it elevates n_{ts} that causes M to rise and R_s to descend. Conversely, raising R_s by reducing M according to Eq. (3.16) lowers n_{ts} and causes P_{out} to ascend, as Eq. (3.13) indicates. Namely, P_{out} increases with rising R_s, or having a higher efficiency requires a lower secrecy and vice versa. This trade-off between secrecy and efficiency should be taken into account in system architecture.

3.6.3 The Effect of Correlation on System Performance

To study the effect of correlation on the system secrecy, we need to investigate how it affects two defined secrecy metrics. Suppose that with some fixed forward and backward erasure rates, for a predetermined secrecy requirement, system parameters including n_{ts}, M and l are designed. We want to analyze how increase in correlation between erasures in main and eavesdropper channels influences outage probability. We only consider the case $\Gamma > 0$ which is more conforming to the real world conditions in which transmission error rates are much smaller than 0.5. For $\Lambda > 0$, based on Eq.'s (3.9) and (3.10) we can obtain missing and false OTF probabilities as

$$\begin{aligned} P_m &= \frac{\varepsilon - p_{11}}{1-\delta} = \varepsilon - \frac{\rho\sqrt{\delta\varepsilon(1-\delta)(1-\varepsilon)}}{1-\delta}, \qquad (3.19) \\ P_F &= \theta\delta + \eta(1-\delta-\varepsilon) + p_{11}(\eta - \theta). \end{aligned}$$

According to Eq. (3.4) with increase in ρ, p_{11} increases. Assuming that feedback erasure rates η and θ are close to each other, the effect of p_{11} and consequently ρ on P_F will be insignificant. However, Eq. (3.19) shows that with rising ρ and therefore p_{11}, P_m falls that accordingly increases P_{out} based on Eq. (3.13). Thus, for an already designed system, increase in correlation leads to a larger outage rate. On the other hand, if we design new system parameters, with increase in ρ, as a result of reduction in P_m, according to Eq. (3.15), system will require a larger n_{ts} as well as a larger M to produce a lower secrecy key rate R_s. It is also intuitively correct that the more correlated Eve's forward channel erasures are with Bob's, the more conforming her decisions about the received packets to Bob's, reducing her uncertainty, so that more data will be transmitted to carry the same amount of uncertainty for her, thereby reducing secret key rate. In this case ψ does not have any effect on R_s because for $\Lambda, \Gamma > 0$, according to Tab. 3.4, Eve's decision does not depend on whether the received feedback bit is erased, making her performance independent of the correlation across backward channels. It could also be inferred from independence of P_m and P_F from ψ in Eq. (3.19).

For $\Lambda < 0$ by Eq.'s (3.9), (3.10), missing and false OTF probabilities can be rewritten as

$$\begin{aligned} P_m &= \frac{(\theta - q_{11})(1-\delta) + \varepsilon(1-\eta)}{(1-\eta)(1-\delta)} - p_{11}\frac{1 + q_{11} - \theta - \eta}{(1-\delta)(1-\eta)}, \\ P_F &= (\eta - q_{11})(1-\delta-\varepsilon) + p_{11}(\eta - q_{11}). \qquad (3.20) \end{aligned}$$

In this scenario for already designed system, with increased ρ and then p_{11}, P_m decreases whereas P_F increases. However, from Eq.'s (3.20), when η and θ are much smaller than 1, the effect of p_{11} on increasing P_F can be assumed to be negligible. This prevailing effect on reducing P_m causes P_{out} to go up, by Eq. (3.13), and for a new design, according to Eq. (3.15), requires system to have a larger n_{ts} and M reducing secret key rate. Unlike $\Lambda > 0$, here increase in ψ impacts system performance as for an erased feedback, Eve decides not to put packet in OTF. For an already designed system parameters, by Eq (3.4) once q_{11} rises with increase in ψ, according to Eq. (3.20), both P_m and P_F decrease causing P_{out} to increase. On the other hand, for a new design it reduces R_s by requiring a larger n_{ts}. Overall, correlation in both forward and backward channels influences secrecy and efficiency of the system in a negative way by decreasing R_s and increasing P_{out}.

3.7 Simulation Results

Our objective in simulations is to evaluate secrecy and efficiency of the designed scheme in various channel conditions. We assume that there exists no discrepancy between Alice and Bob using reconciliation strategy, and that the number of packets in OTF always reaches to n_{ts} by OTF refilling protocol. In these simulations, we require $s = 20$ implying that the upper-bound on Eve's information about secret key does not exceed $I_{sup} = 2^{-20}/\ln 2$ which is sufficiently negligible. For the maximum number of packets within each frame chosen to be $M_{max} = 4096$ with each packet of length $n_b = 78$-bits, we exclude $\log_2 4096 = 12$ number of bits dedicated for sequence number as well as two flag bits from the packet to get $n_r = 64$-bit random part used for key establishment. For the generated key length of $r = 640$-bit, according to Eq. (3.12), the minimum required number of packet discrepancies for Eve will be $l = 11$. We set the thresholds $T_s = 0.99$, $T_{out} = 0.01$ and choose $K = 0$, so packets can only be transmitted once.

3.7.1 Numerical Analysis Based on Secret Key Rate

In numerical analysis we experiment how secret key rate changes with varying correlation. It is assumed that wiretap channel quality is better than the main channel as $\delta = \eta = 0.2$ but $\varepsilon =$

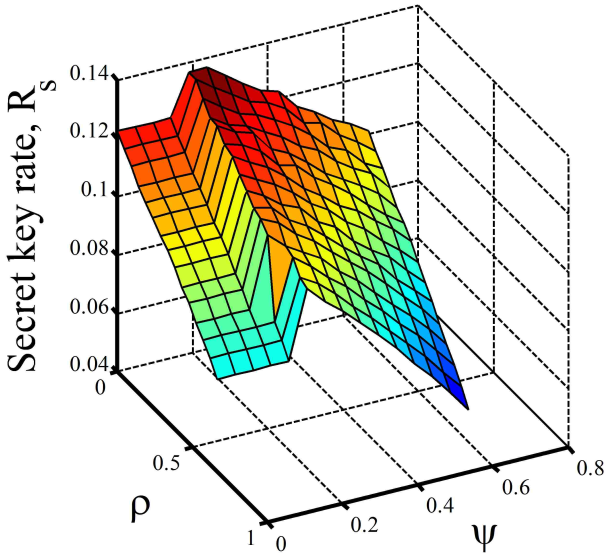


FIGURE 3.3. Obtained secret key rate in terms of forward and backward correlation coefficients with $\delta = \eta = 0.2$ and $\varepsilon = \theta = 0.1$.

$\theta = 0.1$. Then, for different forward and backward correlation coefficients, based on the secrecy requirement $P_{out} < T_{out}$, n_{ts} and M are computed using Eq.'s (3.15), (3.16). Namely, for an upper-bounded P_{out}, each ρ and ψ result in a different secret key rate R_s based on Eq. (3.18). For $\psi < 0.2$, since $\Lambda > 0$, increase in ρ from 0 to 0.8 reduces R_s from 0.135 to 0.075 as illustrated in Fig. 3.3 which conforms with our analysis. As was expected, in this case ψ does not have any effect on R_s. However, for $\psi > 0.2$, we get $\Lambda < 0$, and therefore with increase in ψ, secret key rate goes down to about 0.04 for large ψ and ρ, as shown in Fig. 3.3. Note that correlation coefficients are upper-bounded based on Eq. (3.2). These results show that even when Eve has a better channel than legitimate users, our scheme can provide secrecy for the established key except for highly correlated channel errors.

3.7.2 System Robustness Against Various Channel Conditions

In our simulation we study whether for all channel conditions, the designed system maintains its robustness for required secrecy criterion, i.e. $P_{out} < T_{out}$. To study how forward channel erasure rates influence system performance, throughout this simulation a consistent condition for feedback channel as $\eta = \theta = 0.2$, as well as fixed correlation coefficients $\rho = \psi = 0.2$ are considered. For

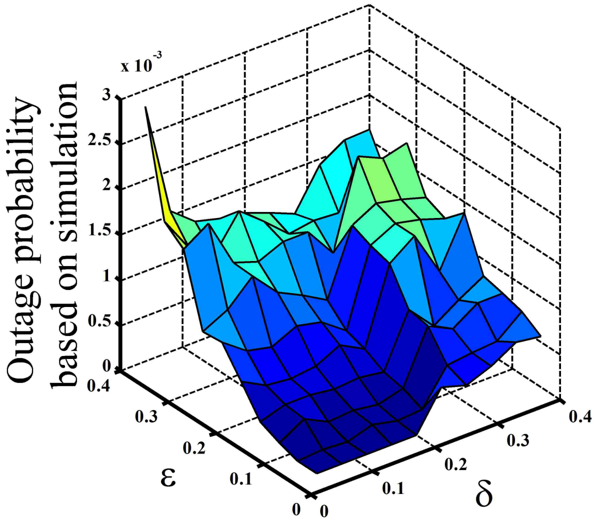


FIGURE 3.4. Simulated outage rate for different forward packet erasure rates in main and wiretap channels, with $\rho = \psi = 0.2$ and $\eta = \theta = 0.2$.

the predetermined I_{sup}, we get $l = 11$, meaning that outage occurs when the number of mismatches between Eve's destination set and the actual set is less than 11. Suppose that Alice is aware of the main and wiretap channel conditions such that for each different δ and ε, determines n_{ts} and M. Then, for the designed system, with 50000 frames, we apply the OTF packet selection within each frame based on Alice and Bob's strategy in Tab.'s 3.2, 3.3 by simulating the erasure rates on their packet and feedback receptions. Similarly, based on Eve's strategy in Tab. 3.4, we find Eve's chosen OTF packets. For each frame, due to mapping strategy, the number of correct packets in Eve's destination sets is the number of packets in her OTF before the first mismatch which is known to Eve by a virtual oracle. Then, by counting the number of frames with outage event we get the average outage rate or experimental P_{out} for each channel condition. In Fig. 3.4, the simulated outage rate is depicted for varying forward channel conditions. It illustrates that even when $\varepsilon < \delta$, namely when wiretap channel has advantage over the main channel, the experimental outage rate is below 0.003 which is much lower than the required threshold $T_{out} = 0.01$, indicating that system is sufficiently secure and robust.

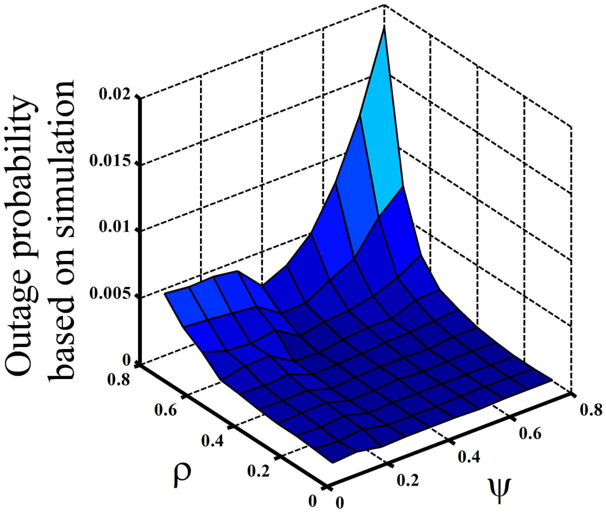


FIGURE 3.5. Simulated outage rate in terms of correlation coefficients across forward and backward channels, with $\delta = \varepsilon = 0.2$ and $\eta = \theta = 0.2$.

3.7.3 System Robustness Against Unknown Wiretap Channel

To study the situation in which Alice is unaware of wiretap channel condition, we conducted another simulation with the same secrecy parameters assuming that Alice designs the system and determines n_{ts}, M based on a presumed correlation coefficients $\rho = \psi = 0.2$, such that this design remains consistent throughout the simulation. All channel erasure rates are supposed to be fixed and equal to 0.2. Then, for different ρ, ψ's simulation is run with 50000 frames to obtain the average outage rate. In Fig. 3.5 the experimental secrecy outage rate is drawn in terms of various forward and backward channel correlations. As it shows, for the most of the region, outage probability is very low, and the system is stable, but when ρ and ψ go above 0.4, outage rate rises very sharply, with P_{out} remaining below $T_{out} = 0.01$ except for $\rho, \psi > 0.7$. As a result, even with the lack of knowledge about wiretap channel correlations, the designed system remains sufficiently secure except for very highly correlated case.

We repeat this simulation but this time with presumed wiretap channel erasure rates $\varepsilon = \theta = 0.2$, and correlation coefficients that are fixed and equal to 0.2. Then, we draw experimentally obtained P_{out} in terms of the varying ε and θ in Fig. 3.6. It illustrates that backward erasure rate θ has little effect on average secrecy outage rate except for very low ε's. However, as forward erasure rate

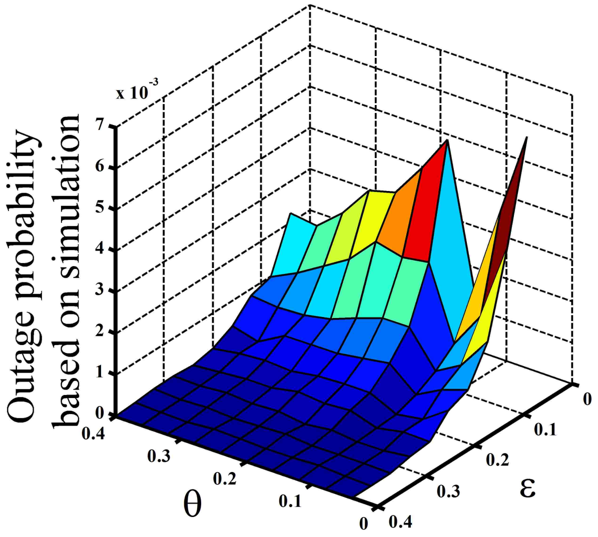


FIGURE 3.6. Simulated outage rate in terms of wiretap channel forward and backward erasure rates, with $\rho = \psi = 0.2$ and $\delta = \eta = 0.2$.

exceeds the presumed $\varepsilon = 0.2$, secrecy outage goes up steeply till it reaches to 0.006 for $\varepsilon = 0$ due to the reduction in P_m, never exceeding the threshold 0.01. These two simulations show that without prior knowledge about Eve's channel conditions, system preserves its robustness from secrecy point of view. Note that simulated outage probability shows much better results than the numerically computed outage rate in Eq. (3.13) because system is designed based on the upper-bound for the actual outage probability (as explained in Appendix 6.6). It provides a pessimistic design of the protocol giving a safety margin when presumptions about channel conditions no longer hold.

3.8 Conclusion

In this chapter, a key scheduling scheme based on ARQ mechanism and privacy amplification is studied. We considered a correlated main and wiretap channel model with noisy feedback channels. The system is designed and its secrecy is analyzed based on outage probability and secret key rate. With numerical and theoretical analysis we showed that correlation between Eve's and legitimate users transmission errors has negative effect on system secrecy. The conducted simulations proved that this scheme delivers its security and maintains its stability even when wiretapper has advantage over legitimate users in channel quality or when wiretap channel conditions are unknown to legitimate users.

Chapter 4
Two-Layer Secrecy System with Exponential Security Against Unbounded Adversary

This chapter is based on our work in [17]. In this work tailoring to any presumed condition on communication channel or any restriction on adversary's resources we design a secrecy scheme with information leakage that decays at an exponential rate. The only requirement for such an exponentially secure system is existence of a common key source between legitimate users that is partially known by Eve. A key extractor based on a sampler and a Rényi extractor derives secret keys with the required entropy from this source. A general cipher uses this key to ensure the required equivocation for the plaintext and to establish the first layer of secrecy. Using privacy amplification on top of this cipher based on inverse universal$_2$ hashing constitutes the second layer of secrecy for highly confidential message transmission with information leakage that is exponentially decreasing. We provide secrecy exponent analysis and optimization to minimize information leakage in terms of two metrics: mutual information and Eve's distinguishability based on L_1 norm distance from uniformity. The required key rate is characterized for different source entropies in order to guarantee the secrecy that is demanded in terms of secrecy exponent for the second secrecy layer and Eve's error rate for the first layer.

4.1 Introduction

The basic secrecy system includes a sender Alice who attempts to transmit as many messages as possible to Bob, which are secured against an eavesdropper who attempts to attain the source information from Alice based on her prior knowledge and observation. In order to design and optimize a secrecy system, we need to evaluate its secrecy by quantifying the amount of information leaked to Eve.

In our secrecy model we consider an unbounded passive adversary and first measure information leakage in terms of mutual information between Alice and Eve's variables. Some works in information theory community adopted information leakage based on mutual information as secrecy criterion [1, 9, 47]. These works only consider a security metric based on mutual information which is required to be negligible for uniformly distributed random message sources. However, in reality, we cannot expect any finitely long messages to be uniformly random since no universally source independent compression exists for such finite sources [62]. Rather, we use a stronger notion of security and require that mutual information to be negligible for any given message distribution.

In cryptography community, security of ciphers has been mainly evaluated on the basis of computationally based metrics against resource constrained attackers. However, recently some researchers have used statistical measures, like variational distance, as secrecy criterion against adversary with unbounded computational power [4–7]. Variational distance is closely related to practical notions of secrecy like Eve's distinguishability and can be used to provide a universally composable notion of secrecy that allows to express secrecy requirement for any protocol environment. As in [7] Eve's distinguishability is defined as half of the L_1 norm distance that is closely related to universal composable security. We adopt Eve's distinguishability based on L_1 distance as another metric to evaluate information leakage from cryptographic point of view.

As studied by [55, 56, 93] in privacy amplification when equivocation of the original information source is larger than the random number generation rate, it is possible to generate a random variable about which Eve's information converges to zero asymptotically. In realistic setting the speed of convergence is of paramount importance because we can only manipulate finite length of random variables. In information theory community the rate of exponential decrease, i.e. error exponent, has been widely discussed [9, 58, 94].

Some works have utilized privacy amplification in the context of physical layer secrecy [58, 63, 95]. Hayashi in [58] showed that when input has equivocation in terms of Rényi entropy of order $1 + \alpha$, after application of universal$_2$ hash function, Eve's information about the generated random variable decreases at an exponential rate that can be lower-bounded. This bound is more generalized and in some cases even tighter than the bound obtained by Bennett in [55]. In [63] Hayashi also provided a lower-bound of the L_1 distance between the output of universal hashing and the uniform random number. However, these works similar to the most of work done in physical layer secrecy (as described in [96]) make some presumptions and require some knowledge about physical channel conditions. Conversely, we do not rely on any physical channel, instead we consider end-to-end secrecy which can be multihop or through internet.

As the only advantage over adversary, we assume that legitimate users have an initial source of randomness in common that is not uniformly distributed for which Eve has some partial knowledge. Such assumption was also made in previous works [6, 10]. This random data shared between Alice and Bob does not need to be absolutely secret, and obtaining such a randomness through public discussion or reconciliation is much easier than providing completely secret bits. In particular, extracted keys in many applications can be repeatedly derived from this source by each time independently sampling of it. This initial key source can be generated through outputs of an imperfect random number generator, a statistical sampler [10] from unpredictable events or a key exchange protocol such as the technique, we used in Chapter 3. Although we consider error free physical channel, physical layer and wireless channel characteristics in wiretap channel model can be used as another means to create such correlated randomness whereby the required keys can be extracted [51].

Utilizing a general cipher like Shannon-type cipher can guarantee the required secrecy for encrypted message in terms of equivocation given Eve's knowledge. The question we address is that how on top of this cipher we can leverage the existing uncertainty about this weak source of randomness to ensure that decreasing exponent of information leakage against an unbounded adversary is sufficiently large making its secrecy asymptotically close to perfect secrecy. For this purpose, we adopt privacy amplification using an invertible universal$_2$ hash function [4] based on $\odot$ multiplication in GF(q^n) that can be implemented with less amount of calculation than most physical layer secrecy approaches whose construction resort to some sophisticated error correction codings [28]. Our contribution can be itemized as:

1. We design a two-layer secrecy scheme in which the privileged advantage for Alice and Bob, that is quantified by Eve's prior uncertainty about the key source, is amplified by an extractor, an inverse universal hash, and a general cipher. In the first layer, enciphering with a secure key assures the required equivocation for regular message transmission that are encapsulated into plaintext blocks. The second layer secrecy that provides exponential secrecy consists of an inverse universal hash operation that transforms the input source message into multiple plaintext blocks being encrypted using the extracted key. At the receiver end after deciphering and recovering these blocks, and then applying universal hash over them, information that leaks to an unbounded adversary and is measured in terms of mutual information and L_1 distance approaches zero at an exponential rate.

2. In this two-layer secrecy scheme with a given random number generation rate and source distribution we provide exponent analysis for both mutual information and L_1 distance as metrics of security. Our secrecy analysis demonstrates quantitatively how the obtained exponent relies on the entropy of the message source as well as Eve's prior information about the key source.

3. We adopt a key extractor that samples a data frame from this partially secure key source and then utilizes an extractor to obtain the cipher key with the required Rényi entropy. All existing extractors measure the extracted randomness in terms of statistical distance from uniformity [97]. In our scheme privacy amplification is based on an uncertainty measures using Rényi entropy, and hence we develop a new notion of extractor that extracts the required randomness on the basis of Rényi entropy. What is notable is that Rényi entropy is a stronger secrecy measure compared to statistical distance.

4. For a particular case where source messages consist of independently and identically distributed (i.i.d) symbols, we optimize the Rényi entropy order to maximize the lower-bound for secrecy exponent. We characterize the required key generation rate that guarantees achiev-

able secrecy for both secrecy layers. For the second layer that is used for transmission of highly confidential part of the message, secrecy is determined in terms of decreasing exponent of information leakage measured by variational distance and mutual information. For the first layer used for regular message transmission, secrecy is measured in terms of adversary's error probability in cryptanalysis also called decryption error probability.

In [4] a similar analysis is used where encryption in utilized to provide underlying secrecy measured in terms of correct decryption probability over which by applying privacy amplification a higher level of secrecy is built up measured in terms of distinguishing security. What mainly distinguishes our work from Bellare et.al. work in [4] is that here we also reach to the goal of exponential secrecy where Eve's advantage vanishes at exponential rate, and moreover unlike their work we do not rely on any physical channel error.

In Section 4.2 the whole scheme of design as well as denotations are illustrated. Section 4.3 discusses a construction of universal$_2$ hashing that is utilized in this work. Key extractor and cipher are described and analyzed in Sections 4.4 and 4.5, respectively. Privacy amplification and exponent analysis based on mutual information and variational distance metrics are detailed in Section 4.6. Numerical analysis and optimization based on dual mode transmission are presented in Section 4.7, and then we conclude in Section 4.8. Proofs for this Chapter are also given in Appendix.

4.2 Proposed Secrecy Scheme model

The transmitter and the receiver side of our proposed secrecy scheme are shown in Fig. 4.1 and 4.2 respectively. We assume that there exists a source of information denoted by V about which Eve has a lower-bounded uncertainty measured in terms of Rényi entropy. Key extractor module that is shown in Fig. 4.3 is used to derive nearly uniform secret key from this weakly random source of data. By independently sampling a segment of this source at time i we obtain a data frame Λ^i that will have the required randomness given Eve's knowledge in terms of Rényi entropy. We show that a key Q^i can be extracted out of Λ^i, by using extractor based on universal hashing, with Rényi

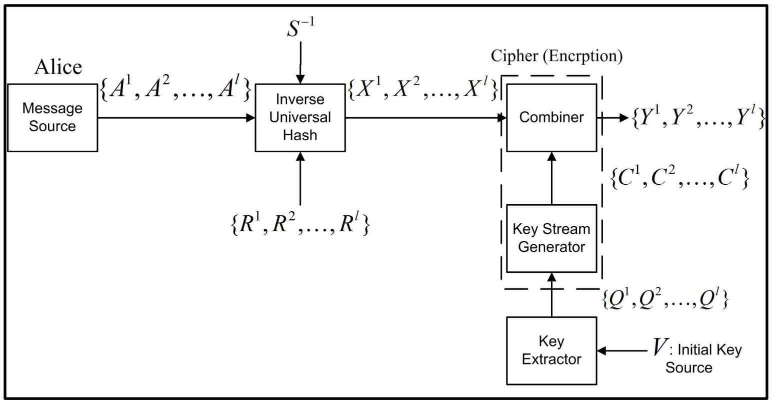

S^{-1}
Alice
Message Source
$\{A^1, A^2, \ldots, A^l\}$
Inverse Universal Hash
$\{X^1, X^2, \ldots, X^l\}$
Cipher (Encrption)
Combiner
$\{Y^1, Y^2, \ldots, Y^l\}$
$\{C^1, C^2, \ldots, C^l\}$
Key Stream Generator
$\{R^1, R^2, \ldots, R^l\}$
$\{Q^1, Q^2, \ldots, Q^l\}$
Key Extractor
V: Initial Key Source

FIGURE 4.1. Transmitter side in the proposed secrecy scheme

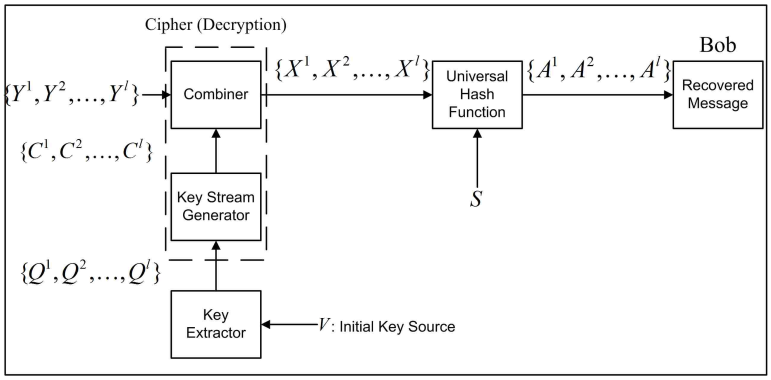

Cipher (Decryption)
Bob
$\{Y^1, Y^2, \ldots, Y^l\}$
Combiner
$\{X^1, X^2, \ldots, X^l\}$
Universal Hash Function
$\{A^1, A^2, \ldots, A^l\}$
Recovered Message
$\{C^1, C^2, \ldots, C^l\}$
Key Stream Generator
S
$\{Q^1, Q^2, \ldots, Q^l\}$
Key Extractor
V: Initial Key Source

FIGURE 4.2. Receiver side in the proposed secrecy scheme

entropy that is asymptotically close to the maximum value. This generated key can be used as a symmetric key for encryption in a general cipher.

Consider a uniformly distributed and randomly chosen function from a universal class of hash functions that is applied upon a source of data with a sufficient equivocation (conditional Rényi entropy given Eve's knowledge). It is proven in [58, 63] that the generated output hash value will have exponentially decreasing information leakage measured in terms of mutual information or L_1 norm distance from uniform distribution.

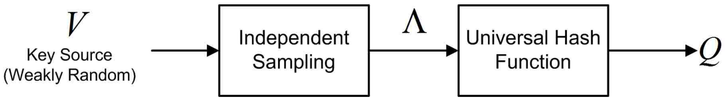


FIGURE 4.3. Key extractor from a weakly random source

As shown in Fig. 4.1, the secure transmission mechanism is applied over a sequence of l blocks with the size of b-symbols. As convention a message block at time i is shown as A^i, with symbols of $\{A_1^i, A_2^i, \dots, A_b^i\}$. A sequence of l concatenated blocks is denoted as $A^{(l)} = \{A^1, A^2, \dots, A^l\}$. Inverse universal hash maps this sequence into a sequence of plaintext blocks $\{X^1, X^2, \dots, X^l\}$ using the same random seed S that is publicly known and a sequence of random vectors $\{R^1, R^2, \dots, R^l\}$ that are uniformly generated. In our scheme we consider an invertible universal hash function based on modulo n multiplication in $GF(q^n)$. Inverse universal hash maps its input into its pre-image that increases its length by adding some randomness through binning. This mapping has a similar functionality as the homophonic encoder in approach proposed in [73] or the random binning based encoding proposed by Wyner and Ciszar in [1, 9]. However, our adopted inverse universal hashing can be considered as a particular encoding approach tailored to Eve's uncertainty over the key source.

Each of the generated n-symbol plaintext blocks at the output of inverse universal hash function will be encrypted independently using a general cipher. The cipher is comprised of a key stream generator to derive key stream C^i from this key Q^i as well as a combiner that combines this key stream with the plaintext block X^i. For $i = 1, \dots, l$, this encryption results in a sequence of ciphertexts $Y^{(l)} = \{Y^1, Y^2, \dots, Y^l\}$, that will be transmitted to Bob and eavesdropped by Eve.

Key extractor and the cipher constitute the first layer of secrecy that ensures sufficient equivocation of plaintext blocks provided that the extracted key has the required Rényi entropy. Upon receiving these ciphertexts, Bob has the same initial key source V and uses inverse mappings to recover the plaintext sequence $X^{(l)} = \{X^1, X^2, \dots, X^l\}$ with a sufficient equivocation. As will be stated in Theorems 7 and 8, after applying the universal hash over this sequence, Eve's information about the resorted message sequence $A^{(l)}$ approaches zero exponentially fast, whose exponent can be bounded properly.

4.3 Universal$_2$ hashing

We adopt universal hashing for privacy amplification and key extraction. An ensemble of the functions h_s that maps set Ω to $\{1, \ldots, M\}$, where S determines statistical behavior of the function h, is called universal$_2$ when it satisfies the following conditions [93]:

Condition 1: $\forall x^1 \neq x^2 \in \Omega$, the probability that $h_s(x^1) = h_s(x^2)$ is at most $\frac{1}{M}$.

Condition 2: For any S, the cardinality of $h_s^{-1}\{i\}$ is independent of the input i.

To make this concrete we give an example of a universal$_2$ hash function with an efficiently computable inverter that can be used for key derivation and privacy amplification in our scheme. The construction was used in [4] as randomness extractor. Here, we use a more general symbol-wise format of this construction. If we interpret n-symbol strings as elements of the finite field GF(q^n), we shall define a multiplication operator $\odot$ on them. Let set Ω be $\{0, \ldots, q-1\}$ and consider seed S that is drawn uniformly from the set $\mathcal{SD} = \Omega^n \backslash 0^n$. We define the universal hash function $h : \mathcal{SD} \times \Omega^n \rightarrow \Omega^b$ that operates on inputs $X \in \Omega^n$ and $S \in \mathcal{SD}$ to output the first truncated b-symbols of $X \odot S$ as $A = h(X, S) = \text{trunc}_b(X \odot S)$.

Let S^{-1} be the inverse of S with respect to multiplication in GF(q^n). Then, we can efficiently invert this universal hashing by the function $h^{-1} : \mathcal{SD} \times \Omega^{n-b} \times \Omega^b \rightarrow \Omega^n$ defined as $h^{-1}(S, R, A) = (A||R) \odot S^{-1}$, for R uniform over Ω^{n-b}. In Appendix 6.7 we show that both conditions 1 and 2 hold for this function, meaning that in addition to uniformity of the the output hash value, every point in the range has the same number of preimages.

4.4 Key Extractor

With the existence of an initial key source that contains some good amount of randomness but is non-uniformly distributed or partially known by Eve, we need to design a key extracting function based on essential cryptographic components that derives required keys from this imperfect source with a randomness close to uniform. The assumption on existence of such a source was also used in some key extracting techniques like [6, 10]. This random data can be produced through different means such as hardware devices based on thermal noise, statistical sampling of user's keyboard strokes or timing data obtained from the hard disk or packet transmission in a network [6].

Consider random variable V common between Alice and Bob, consisting of ν random variables as $V = (V_1, V_2, \ldots, V_\nu)$ that is used as initial keying source. This keying source V gathered by users has to contain enough uncertainty at Eve's side in terms of Rényi entropy of order 2 denoted by $H_2(V)$. As shown in Fig. 4.3 the first step in key extractor is a sampling module that each time independently samples a λ-tuple from this source such that each symbol can only be sampled once. For any λ-tuple $\underline{i} = (i_1, i_2, \ldots, i_\lambda)$ with $1 \leq i_1 < i_2 < \ldots < i_\lambda \leq \nu$ let $V_{\underline{i}}$ be the sampled string $(V_{i_1}, V_{i_2}, \ldots, V_{i_\lambda})$. Then, it is shown in [98] that

$$H_2(V_{\underline{i}}) \geq H_2(V) - (\nu - \lambda). \tag{4.1}$$

For $H_2(V) - (\nu - \lambda) = \delta$ if we denote the randomly sampled λ-tuple string at time i as Λ^i, it will have collision entropy of at least δ. Rényi entropy is a decreasing function with respect to its order [20], so $H_{1+\alpha}(\Lambda^i) \geq H_2(\Lambda^i) \geq \delta$ for $0 < \alpha \leq 1$, and Eve's uncertainty about the sampled output in terms of Rényi entropy of order $1 + \alpha$ will be at least δ.

Randomness extractors are well suited to address the need for key derivation functionality which maps input distributions with sufficient entropy into outputs with distributions statistically close to uniform [97]. To the best of authors' knowledge, so far all definitions of extractors measure randomness of the extracted output on the basis of statistical distance from uniformity. However, since in our scheme privacy amplification is characterized based on Rényi entropy, we need to develop a new notion of extractor that extracts randomness in terms of Rényi entropy. Therefore, we resort to the use of cryptographic hash functions as the basis for such extractor. We prove the following Theorem in Appendix 6.8.

Theorem 5. *Consider a universal class of hash functions $h_s : \Omega \rightarrow \mathcal{K}$; where S is uniform over $\mathcal{SD}$. If we apply h_s over input $X \in \Omega$ with Rényi entropy of $H_{1+\alpha}(X) \geq \delta$, for $0 < \alpha \leq 1$, the generated hash value $Q = h_s(X)$ such that $Q \in \mathcal{K}$ attains Rényi entropy with the following* *lower-bound*

$$H_{1+\alpha}(Q) \geq \log|\mathcal{K}| - \frac{1}{\alpha} e^{-\alpha[\delta - \log|\mathcal{K}|]}. \tag{4.2}$$

Now, we can define the new notion of Rényi extractor:

Definition 2. *RenExt: $\Omega \times \mathcal{SD} \to \mathcal{K}$ is a (δ, ε) Rény extractor if for every seed S uniformly chosen on $\mathcal{SD}$ and every source $X \in \Omega$ of Rényi entropy $H_2(X) \geq \delta$, it holds that RenExt(X, S) has Rényi entropy of at least $\log|\mathcal{K}| - \varepsilon$.*

Rényi extractor based on universal hashing results in entropy loss ε which is exponentially decreasing. By applying this Rényi extractor over sampled data frame Λ^i which has Rényi entropy of at least δ, provided that $\delta > \log|\mathcal{K}|$, we obtain a key $Q^i \in \mathcal{K}$ with Rény entrpy

$$H_{1+\alpha}(Q^i) \geq \log|\mathcal{K}| - \frac{1}{\alpha} e^{-\alpha[\delta - \log|\mathcal{K}|]} = \log|\mathcal{K}| - \varepsilon. \tag{4.3}$$

If we adopt universal hashing technique based on $\odot$ multiplication in GF(q^n) for key extraction, we need to use independent seeds for each key derivation. It ensures that then due to independent sampling and mapping used in key extractor the generated keys will be independent of each other.

4.5 Cipher

Consider a deterministic cipher that consists of a key stream generator and a combiner. Let the plaintext block at time i be $X^i = (X_1^i, X_2^i, \ldots, X_n^i)$ where $X_j^i \in \Omega$. The key extractor output is Q^i that takes values in the set $\mathcal{K}$ with total of e^{nR_s} elements. At time i key stream generator maps the input key Q^i to the key stream of length n, $C^i = (C_1^i, C_2^i, \ldots, C_n^i)$ with components from the set Ω using the mapping $\Phi : \mathcal{K} \to \Omega^n$. We define the cipher as the set $C^* = \{C^1, C^2, \ldots, C^{e^{nR_s}}\}$ of key streams with length n and key rate of $R_s = \frac{\log|\mathcal{K}|}{n}$.

For $i = 1, \ldots, l$ the cipher produces the ciphertext block Y^i from the i^{th} plaintext block X^i and the i^{th} key stream C^i using the combiner $f(.,.)$ that maps $\Omega^n \times \Omega^n \to \Omega^n$.

$$Y^i = f(X^i, C^i) \quad i = 1, 2, \cdots. \tag{4.4}$$

Bob is aware of the key Q^i used at time i and can generate the same key stream C^i using key stream generator. He applies inverse mapping $g(.,.)$ to the received ciphertext block C^i in order to recover the plaintext block X^i, where $g : \Omega^n \times \Omega^n \to \Omega^n$ and $X^i = g(Y^i, C^i)$. Depending on the mapping f, the cipher could be block, stream cipher, or additive-like cipher.

Now the question is how much Eve knows about the plaintext. She has knowledge about the system and all the mappings and can receive ciphertexts. However, she lacks a complete knowledge, thereby resulting in uncertainty about the cipher key. We use the following Lemma, proven in Appendix 6.9, in Theorem 6 to quantify equivocation of the plaintext in terms of Rényi entropy.

Lemma 6. *For conditional Rényi entropy of order $1 + \alpha$ with $\alpha > 0$ we have*

$$H_{1+\alpha}(X|Y) \geq H_{1+\alpha}(X, Y) - H(Y). \tag{4.5}$$

Theorem 6. *Let $X = (X_1, X_2, \ldots, X_n)$, $Y = (Y_1, Y_2, \ldots, Y_n)$ and $C = (C_1, C_2, \ldots, C_n)$ be random vectors representing plaintext block, ciphertext block and the key stream, respectively, where $X_i, Y_i, C_i \in \Omega$. Let Q denote the random vector representing the key. For $\alpha > 0$, equivocation of the plaintext satisfies*

$$H_{1+\alpha}(X|Y) \geq H_{1+\alpha}(X) + H_{1+\alpha}(Q) - n \log |\Omega|. \tag{4.6}$$

Proof. Let $H_{1+\alpha}(X, Y)$ be the joint Rényi entropy of the plaintext and ciphertext. Since $f(.,.)$ is a one-to-one mapping, it is easy to see that $H_{1+\alpha}(X, Y) = H_{1+\alpha}(X, C)$. But we know that the key stream C is independent of the input plaintext X implying that $H_{1+\alpha}(X, C) = H_{1+\alpha}(X) + H_{1+\alpha}(C)$. Moreover, key stream generator that uses mapping Φ does not increase entropy against adversary and therefore Eve's lack of knowledge about the key Q will be transformed to her uncertainty about the key stream C, i.e. $H_{1+\alpha}(C) = H_{1+\alpha}(Q)$. For $Y \in \Omega^n$ we have $H(Y) \leq n \log |\Omega|$. Then, using by Eq. (4.5) gives us the equivocation of order $1 + \alpha$. □

In our secrecy analysis we will only consider $0 < \alpha \leq 1$. To measure redundancy of the input plaintext X we use Rényi divergence of order $1+\alpha$ as $D_{1+\alpha}(P_X||\tilde{P}_X)$ where $\tilde{P}_X$ is uniform distribution over Ω^n. If we define normalized redundancy of X as $d_\alpha^X \triangleq D_{1+\alpha}(P_X||\tilde{P}_X)/n$, we will have

$$H_{1+\alpha}(X) = n \log |\Omega| - nd_\alpha^X. \tag{4.7}$$

As a result, we can rewrite Eq. (4.6) as

$$H_{1+\alpha}(X|Y) \geq H_{1+\alpha}(Q) - nd_\alpha^X. \tag{4.8}$$

The key generation rate R_s has to be specified in order to guarantee the required secrecy for the first layer of the scheme including the cipher and key extractor. It aims at a minimum required equivocation for Eve about the message that can be characterized in terms of the average error probability in Eve's estimation of the plaintext block. Let X^* be the estimate of adversary from the plaintext block X based on the maximum aposteriori probability (MAP) given the received ciphertext Y as $X^* = \max_{X \in \Omega^n} Pr[X|Y]$. MAP decision rule minimizes the average probability of error per plaintext block defined as

$$P_e = 1 - E_Y[\max_{X \in \Omega^n} Pr(X|Y)]. \tag{4.9}$$

As it is proven in [99], Rényi entropy can be used to bound error probability of MAP decision rule based on an analogue of Fano's lemma that is stated below:

Theorem 6 in [99]: Let P be the set of aposteriori probabilities as $P = \{P_1, P_2, \ldots, P_m\}$, and $H_\beta(P)$ be the conditional Rényi entropy that is defined based on P. Let estimation error probability be $P_e = 1 - \max P_i$. Then, for $\beta \neq 1$ the maximum upper-bound for $H_\beta(P)$ is attained as

$$\bar{h}_\beta(P_e) = \frac{1}{1-\beta} \log \left[(1-P_e)^\beta + (\frac{1}{m-1})^{\beta-1} P_e^\beta \right]. \tag{4.10}$$

In plaintext estimation by adversary, with aposteriori probability of $Pr(X|Y)$ and estimation error probability given in Eq. (4.9), noting that X belongs to the alphabet of size $|\Omega|^n$, we can obtain the upper-bound for conditional Rényi entropy of order $1+\alpha$ with $0<\alpha\leq 1$ as

$$H_{1+\alpha}(X|Y) \leq \bar{h}_{1+\alpha}(P_e) = -\frac{1}{\alpha}\log\left[(1-P_e)^{1+\alpha} + (\frac{1}{|\Omega|^n-1})^{\alpha} P_e^{1+\alpha}\right], \alpha > 0. \tag{4.11}$$

We adopt error probability of attacker in estimation of the plaintext block using MAP as the secrecy metric for the first layer of the scheme. Such metric was also used in previous works [66, 68] as secrecy criterion. Let us determine a threshold as P_e^{th} and design the system with a key stream generation rate assuring that Eve's block error probability exceeds this threshold. It is easy to see that for $0<\alpha\leq 1$, $\bar{h}_{1+\alpha}(P_e)$ is a monotonic increasing function of P_e. In other words, if we ensure that $\bar{h}_{1+\alpha}(P_e^{th}) \leq H_{1+\alpha}(X|Y)$ due to inequality (4.11) and monotonic behavior of $\bar{h}_{1+\alpha}(P_e)$ we will have $\bar{h}_{1+\alpha}(P_e) \geq \bar{h}_{1+\alpha}(P_e^{th})$ that infers $P_e \geq P_e^{th}$. Let $\tau_\alpha \triangleq \bar{h}_{1+\alpha}(P_e^{th})$, so we need to make sure that equivocation in Eq. (4.8) never drops below τ_α, which requires that

$$H_{1+\alpha}(Q) - nd_\alpha^X \geq \tau_\alpha. \tag{4.12}$$

If we use our proposed key extracting technique, $H_{1+\alpha}(Q)$ can be lower-bounded according to Eq. (4.2). Then, for the above inequality to hold we need to have

$$\log|\mathcal{K}| \geq nd_\alpha^X + \tau_\alpha + \frac{1}{\alpha}e^{-\alpha(\delta-\log|\mathcal{K}|)}, \tag{4.13}$$

Hence, we can infer that key stream generation rate has to be

$$R_s \geq d_\alpha^X + \frac{\tau_\alpha + \frac{1}{\alpha}e^{-\alpha(\delta-\log|\mathcal{K}|)}}{n}. \tag{4.14}$$

It characterizes the minimum required key rate for the first layer of secrecy. Note that we require $\delta > \log|\mathcal{K}|$.

4.6 Privacy Amplification

4.6.1 Secrecy Exponent Analysis Based on Mutual Information

The main objective of using universal hashing in our scheme is bringing secrecy up to the second layer by privacy amplification. Let h_s be an ensemble of universal hash functions that maps set

Ω^n to $\{1, \ldots, M\}$ and satisfies both conditions 1 and 2. As proven in [58], information leakage in terms of mutual information, averaged over possible seeds S, satisfies

$$E_s\left[I(h_s(X);Y)\right] \leq \min_{0<\alpha\leq 1} \frac{e^{-\alpha(H_{1+\alpha}(X|Y)-\log M)}}{\alpha}. \tag{4.15}$$

So we can find a function h_s from Ω^n to $\{1, \ldots, M\}$ that

$$I(h_s(X);Y) \leq \min_{0<\alpha\leq 1} \frac{e^{-\alpha(H_{1+\alpha}(X|Y)-\log M)}}{\alpha}, \tag{4.16}$$

where Y denotes the obtained knowledge by Eve. Eq. (4.16) implies that when legitimate users have a common randomness denoted by X with equivocation of at least $H_{1+\alpha}(X|Y)$, we can make sure that the upper-bound for Eve's knowledge about the output of $h_s(X)$ decreases with the exponent of $\alpha(H_{1+\alpha}(X|Y) - \log M)$. The larger this exponent is, the closer system secrecy will be to the perfect secrecy with zero information leakage, i.e. $I(h_s(X);Y) = 0$.

In our scheme Alice and Bob exchange random vector X through encryption that enables them to have a shared body of random data with equivocation of $H_{1+\alpha}(X|Y)$. As the next step if we apply the universal hash function based on $\odot$ multiplication in GF(q^n), that maps Ω^n to Ω^b, it can be assured that output $A = h_s(X)$ will have information leakage with the decreasing exponent of at least $\alpha(H_{1+\alpha}(X|Y) - \log M)$, for $M = |\Omega^b|$. Now if we reverse this process and obtain X from input A using inverse universal hash that maps Ω^b to Ω^n, we will get the same results since condition 2 guarantees that the cardinality $|h_s^{-1}(A)|$ does not depend on A. Let Alice generate uniformly random string R over Ω^{n-b} and apply inverse function h_s^{-1} over the input A, R and S^{-1} to obtain plaintext $X = (A||R) \odot S^{-1}$. Then, decreasing exponent of information leakage about A will be at least $\alpha(H_{1+\alpha}(X|Y) - \log M)$.

Consider l-fold scenario of the abovementioned mechanism where input message is framed into a sequence of l blocks denoted by $A^{(l)} = \{A^1, A^2, \ldots, A^l\}$ for $A^i \in \Omega^b$. Alice generates the sequence of l uniformly random $(n-b)$-symbol strings $R^{(l)} = \{R^1, R^2, \ldots, R^l\}$, and then by using inverse universal hash, outputs $X^i = (A^i||R^i) \odot S^{-1}$, to map $A^{(l)}$ to the sequence of plaintexts $X^{(l)} = \{X^1, X^2, \dots, X^l\}$. Through encryption using the sequence of l key streams $C^{(l)} = \{C^1, C^2, \dots, C^l\}$ that are generated using the cipher keys $\{Q^1, Q^2, \dots, Q^l\}$, $X^{(l)}$ will be mapped to the sequence of ciphertexts $Y^{(l)} = \{Y^1, Y^2, \dots, Y^l\}$ as $Y^i = f(X^i, C^i)$. Let $A^{(l)} \in \mathcal{I}$. We can obtain the random number generation rate in this l-fold transmission mechanism as

$$\rho \triangleq \lim_{l\to\infty} \frac{\log|\mathcal{I}|}{l} = \lim_{l\to\infty} \frac{\log|\Omega^b|^l}{l} = \log|\Omega^b| = \log M. \tag{4.17}$$

At the receiver end after deciphering and recovering of $X^{(l)}$ that contains l plaintext blocks, universal hashing will be applied over them to restore message blocks as $A^i = \text{trunc}_b(X^i \odot S)$. Note that the same seed S, uniformly chosen over Ω^n, is used for hashing of all l blocks that has to be publicly known before their transmission. We define mutual information based secrecy exponent for l-fold scenario as

$$e_I^{(l)} \triangleq \lim_{l\to\infty} \frac{-\log I(h^{(l)}(X^{(l)}); Y^{(l)})}{l}, \tag{4.18}$$

whose lower-bound is given in the following Theorem:

Theorem 7. *Let random variable A represent the message block of size b with components in set Ω where $M = |\Omega^b|$ and Q represent the cipher key. Then, for the described l-fold transmission mechanism in two layer secrecy scheme, mutual information based secrecy exponent satisfies:*

$$e_I^{(l)} \geq \max_{0<\alpha\leq 1} \alpha(H_{1+\alpha}(Q) + H_{1+\alpha}(A) - 2\log M). \tag{4.19}$$

Proof. Since R^i and A^i are independent of R^j and A^j for $i \neq j$, for a given S, X^i and X^j will be independent of each other. Namely, revealing any information about any of the plaintexts does not assist Eve to reduce her uncertainty about other ones. Consequently, we shall write

$$H_{1+\alpha}(X^{(l)}|Y^{(l)}) = lH_{1+\alpha}(X^i|Y^i) = lH_{1+\alpha}(X|Y). \tag{4.20}$$

We use Cartesian product construction of universal class of hash functions in order to enlarge the domain of hash family. In this construction hashed outputs, that are generated using hash function with the same seed, are concatenated, where $h^{(l)}(X^{(l)})$ is defined as $h_s(X^1)||h_s(X^2)||\dots||h_s(X^l)$. Stinson showed in [100] that Cartesian product based universal hashing denoted by $h^{(l)}$ results in the same collision probability as h_s. Namely, using only one seed for l transformations does not compromise security.

At receiver l-fold universal hash function $h^{(l)}$ maps $X^{(l)}$ to $A^{(l)}$. Joint distribution of $X^{(l)}$ and $Y^{(l)}$ denoted by $P_{X^{(l)},Y^{(l)}}$ can be obtained by l-fold identical and independent distribution of $P_{X,Y}$ as $(P_{X,Y})^l$, so we can infer from Eq. (4.20) that

$$\begin{aligned} I(h^{(l)}(X^{(l)});Y^{(l)}) &\leq \min_{0<\alpha\leq 1} \frac{e^{-\alpha(H_{1+\alpha}(X^{(l)}|Y^{(l)})-\log|\mathcal{A}^b|^l)}}{\alpha} \\ &= \min_{0<\alpha\leq 1} \frac{e^{-\alpha l(H_{1+\alpha}(X|Y)-\log|\mathcal{A}^b|)}}{\alpha}. \end{aligned} \tag{4.21}$$

According to Eq. (4.21), $e_I^{(l)}$ can be obtained by

$$e_I^{(l)} \geq \max_{0<\alpha\leq 1} \alpha(H_{1+\alpha}(X|Y)-\log M). \tag{4.22}$$

Based on the secrecy analysis for the first layer of the scheme which is constituted of the cipher and the key extractor, we obtained equivocation of each plaintext block in Eq. (4.8). On the other hand, redundancy of the plaintext can be quantified in terms of entropy of the message block from which it is derived using inverted universal hashing. Random vector R is of size $n-b$ with components in the set Ω meaning that R is uniformly generated over Ω^{n-b}. For a given seed S, distribution of random vector X that is obtained as $X=(A||R)\odot S^{-1}$ is determined based on the distribution of R and input A that are independent of each other. As a result,

$$P_X(X|S) = P_A(A).P_R(R) = \frac{P_R(A)}{|\Omega|^{n-b}}. \tag{4.23}$$

Thus, we can compute Rényi entropy of the plaintext X

$$H_{1+\alpha}(X) = (n-b)\log|\Omega| + H_{1+\alpha}(A). \tag{4.24}$$

Replacing Eq. (4.6) in this Eq. shows that equivocation satisfies

$$H_{1+\alpha}(X|Y) \geq H_{1+\alpha}(Q) + H_{1+\alpha}(A) - \log M. \tag{4.25}$$

Then, substituting it in Eq. (4.22) gives us the desired lower bound of the secrecy exponent in Eq. (4.19). □

Eq. (4.19) indicates that the decreasing exponent of information leakage depends only on the entropy of the generated key, uncertainty about the source message as well as the random number generation rate. Note that initial source V will be used for encryption of l blocks such that l keys has to be derived out of l times λ-tuple sampling of it, therefore its length ν should satisfy $\nu \geq l\lambda$. To have positive secrecy exponent for a source with the entropy of $H_{1+\alpha}(A)$, extracted key entropy for single block encryption has to be at least

$$H_{1+\alpha}(Q) \geq 2\log M - H_{1+\alpha}(A). \tag{4.26}$$

By replacing the entropy for extracted key given in Eq. (4.2), and defining $\gamma \triangleq \delta - \log|\mathcal{K}|$, we get the following requirement for key size

$$\log|\mathcal{K}| \geq 2\log M + \frac{1}{\alpha}e^{-\gamma} - H_{1+\alpha}(A) \text{ where } \gamma > 0. \tag{4.27}$$

This condition guarantees exponential security for highly confidential message transmission.

4.6.2 Secrecy Exponent Analysis Based on L_1 distance

We use Eve's distinguishability as the second metric to characterize leaked information. Hayashi in [63] adopted it as secrecy criterion that is close to the universal composable security used in [7]. Consider an ensemble of functions h_s that maps the random number $X \in \Omega^n$ to $\{1, 2, \ldots, M\}$ satisfying both universality conditions. Alice and Bob apply the same function to the common random variable X to obtain $h_s(X)$. Let $Y \in \Omega$ be the random variable representing Eve's knowledge where $P_{h_s(X),Y}$ denotes the joint distribution of $h_s(X)$ and Y. Let $\tilde{P}_{h_s(X)}$ be the uniform distribution on $\{1, 2, \ldots, M\}$. Eve's distinguishability is defined [63] as

$$d_1(P_{h_s(X),Y}|Y) = d_1(P_{h_s(X),Y}, \tilde{P}_{h_s(X)} \times P_Y). \tag{4.28}$$

According to Eq. (1.2) it can be rewritten as

$$d_1(P_{h_s(X),Y}|Y) = \sum_{y} P_Y(y) d_1(P_{h_s(X)|Y=y}, \tilde{P}_{h_s(X)}). \tag{4.29}$$

It measures randomness of the output value of a particular hash function h_s from Eve's perspective in terms of L_1 distance from the uniform distribution, averaged over Eve's possible knowledge Y. If we average this distance over all possible seeds $S \in \mathcal{SD}$, when the resulted value is sufficiently small, we can be certain that $h_s(X)$ is independent of random variables S and Y. Thus, the generated random variable will be suitable even when we randomly choose the hash function.

In [63] it is shown that for $0 < \alpha \leq 1$

$$E_s\left[d_1(P_{h_s(X),Y}|Y)\right] \leq \min_{0<\alpha\leq 1} 3M^{\frac{\alpha}{\alpha+1}} e^{-\frac{\alpha}{\alpha+1}H_{1+\alpha}(X|Y)},$$

where E_s denotes expectation in terms of the random variable S. As a result, there exists a function h_s such that

$$d_1(P_{h_s(X),Y}|Y) \leq \min_{0<\alpha\leq 1} 3e^{-\frac{\alpha}{\alpha+1}(H_{1+\alpha}(X|Y)-\log M)}. \tag{4.30}$$

This equation implies that when equivocation of X is larger than the random number generation rate, $\log M$, distribution of the generated random variable $h_s(X)$ asymptotically approaches to uniformity. Consider our two layer secrecy scheme in which inverse of l-fold universal hash function $h^{(l)}$, using the same publicly known seed, maps a sequence of message blocks $A^{(l)}$ to a sequence of plaintext blocks $X^{(l)}$. We define decreasing exponent of L_1 distance of generated secret messages from uniform random numbers as

$$e_1^{(l)} \triangleq \lim_{l\to\infty} \frac{-\log d_1(P_{h^{(l)}(X^{(l)}),Y^{(l)}}|Y^{(l)})}{l}, \tag{4.31}$$

whose lower-bound can be characterized using Theorem 8:

Theorem 8. *Let random variable A represent the message block of size b with components in the set Ω where $M = |\Omega^b|$, and Q represent the cipher key. Then, for the proposed secrecy scheme with*

l-fold transmission mechanism, the secrecy exponent based on L_1 distance satisfies:

$$e_1^{(l)} \geq \max_{0<\alpha\leq 1} \frac{\alpha(H_{1+\alpha}(Q) + H_{1+\alpha}(A) - 2\log M)}{1+\alpha}. \tag{4.32}$$

Proof. At receiver l-fold universal hash function $h^{(l)}$ generates the message sequence $A^{(l)}$ that belongs to the set $\mathcal{I}$ where $|\mathcal{I}| = |\Omega^b|^l$. By using Eq.'s (4.20) and (4.30) we write

$$\begin{aligned} d_1(P_{h^{(l)}(X^{(l)}),Y^{(l)}}|Y^{(l)}) &\leq \min_{0<\alpha\leq 1} 3e^{-\frac{\alpha}{\alpha+1}(H_{1+\alpha}(X^{(l)}|Y^{(l)}) - \log|\Omega^b|^l)} \\ &= \min_{0<\alpha\leq 1} 3e^{-\frac{\alpha l}{\alpha+1}(H_{1+\alpha}(X|Y) - \log|\Omega^b|)}. \end{aligned}$$

For the random number generation rate of $\rho = \log M$ where $M = |\Omega^b|$, we can obtain decreasing exponent of L_1 norm based on its definition in Eq. (4.31) as

$$e_1^{(l)} \geq \max_{0<\alpha\leq 1} \frac{\alpha(H_{1+\alpha}(X|Y) - \log M)}{1+\alpha}. \tag{4.33}$$

Random variable X represents the plaintext blocks that are generated by the inverse universal hash operation and then encrypted using the cipher, giving them equivocation obtained in Eq. (4.25). Substituting it in Eq. (4.33) completes the proof. □

As a result, if we use the proposed key extractor to derive cipher keys, the same condition in Eq. (4.27) needs to hold to have information leakage in terms of variational distance or Eve's distinguishability decay exponentially to zero.

4.6.3 Comparison Between Bounds and Metrics

First of all, we compare two bounds presented for the exponent of information leakage, one based on mutual information in Eq. (4.22) and the other one based on L_1 distance in Eq. (4.33). Mutual information between two random variables X and Y can be written in terms of KL divergence

$$I(X;Y) = D(P_{X,Y}||\tilde{P}_X \times P_Y). \tag{4.34}$$

Hence, according to the definition of secrecy exponent $e_I^{(l)}$ in Eq. (4.18) we shall rewrite Eq. (4.22) as

$$\lim_{l\to\infty} -\frac{1}{l} \log D(P_{h^{(l)}(X^{(l)}),Y^{(l)}}||\tilde{P}_{h^{(l)}(X^{(l)})} \times P_{Y^{(l)}}) \geq \max_{0<\alpha\leq 1} \alpha(H_{1+\alpha}(X|Y) - \log M), \tag{4.35}$$

so by using the property in Eq. (1.7), the exponent for L_1 distance can be lower-bounded according to

$$\lim_{l\to\infty} -\frac{1}{l}\log d_1(P_{h^{(l)}(X^{(l)}),Y^{(l)}}|Y^{(l)}) \geq \max_{0<\alpha\leq 1} \frac{\alpha(H_{1+\alpha}(X|Y)-\log M)}{2}. \tag{4.36}$$

However, this bound is smaller than the lower-bound we used to characterize exponent of information leakage in terms of variational distance, implying that the bound in Eq. (4.33) is tighter than the one for mutual information in Eq. (4.22). Moreover, these two bounds become equivalent when α that maximizes the lower-bound in Eq. (4.33) is equal to 1.

On the other hand, If we compare these two metrics, based on the equivalence of mutual information and KL-distance and inequality (1.7), we infer that

$$\frac{1}{2}\log I(h^{(l)}(X^{(l)});Y^{(l)}) \geq \log d_1(P_{h^{(l)}(X^{(l)}),Y^{(l)}}|Y^{(l)}). \tag{4.37}$$

This inequality indicates that whenever system is secure from mutual information point of view, and the left hand side is smaller than a sufficiently small number, the right hand side will also be upper-bounded making information leakage in terms of variational distance negligible. Not to mention that mutual information is a stronger metric compared to L_1 distance. Nevertheless, the main reason that makes variational distance a more suitable secrecy metric from cryptographic perspective is that it simplifies formulation for practical analysis of any protocol environment and can be augmented with practical notions of secrecy like Eve's distinuishability.

4.7 Optimization and Analysis of dual mode transmission mechanism

As discussed in previous sections the proposed secrecy scheme provides exponential secrecy if privacy amplification is applied on top of the cipher with the stipulation on the key length that is formulated in Eq. (4.27). It implies that there exists a trade-off such that exponential secrecy that guarantees a higher level of secrecy requires a relatively high key rate. On the other hand, only highly confidential part of the message requires exponential secrecy and a higher key rate,

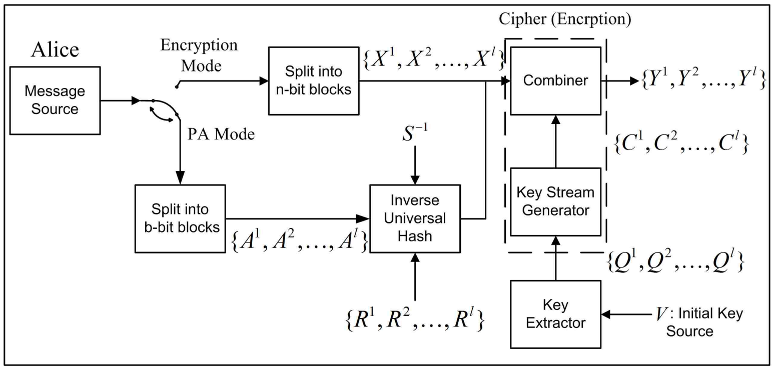


FIGURE 4.4. Dual mode transmission with two layers of secrecy

whereas normally it is not demanded in regular transmission. As a result, we design a dual mode transmission mechanism depicted in Fig. 4.4 that, depending on the demanded level of secrecy for transmission, switches to either encryption or privacy amplification (PA) mode.

For analysis, we consider special case with binary i.i.d. message source such that $\Omega = \{0, 1\}$. For regular and efficient transmission that does not require additional security, transmission mechanism switches to encryption mode where input message is framed into n-bit blocks and then encrypted by a cipher. For this layer of secrecy, we adopt Eve's probability of failure in estimating the correct plaintext block as the secrecy criterion. Such error probability was also considered as a metric to measure security in [66, 68]. As discussed in Section 4.5 average block error probability of Eve denoted by P_e exceeds the required threshold P_e^{th} if the required condition for the length of the extracted key in Eq. (4.13) is satisfied.

In encryption mode each plaintext block is equivalent to n-bit message block. Let us represent a plaintext block with random vector $X \in \{0, 1\}^n$ whose components X_i, for $i = 1, \ldots, n$, are independently and identically generated Bernoulli random variables with $Pr(X_i = 1) = p$. It is easy to see that X has Rényi entropy of

$$H_{1+\alpha}(X) = -\frac{n}{\alpha} \log[p^{1+\alpha} + (1-p)^{1+\alpha}]. \tag{4.38}$$

As a result, normalized redundancy of d_α^X can be written according to Eq. (4.8) as

$$d_\alpha^X = \log|\Omega| + \frac{1}{\alpha} \log[p^{1+\alpha} + (1-p)^{1+\alpha}]. \tag{4.39}$$

Therefore, the minimum required key length for encryption mode given in Eq. (4.13) will be a function of p, n, P_e^{th}, α and γ that we denote as $\Gamma_e(p, n, P_e^{th}, \alpha, \gamma)$.

As shown in Fig. 4.4 when a part of message requires a higher level of secrecy, transmission mechanism switches to PA mode where the message source is encapsulated into b-bit blocks, and then a sequence of l concatenated message blocks will be mapped to a sequence of plaintext blocks using inverse universal hashing. In Section 4.6 we obtained the lower-bound for decreasing exponent of information leakage in Eq. (4.19) that has to be maximized in terms of the order of Rényi entropy to have the highest possible decreasing rate for information leakage. Considering i.i.d. message source whose components are Bernoulli random variables with probability p, and extracted cipher key whose entropy is given in Eq. (4.2), we can obtain the lower-bound for the decreasing exponent as

$$G_I(\alpha, \gamma, p) = \alpha(\log|\mathcal{K}| - 2\log M) - e^{-\alpha\gamma} - b\log[p^{1+\alpha} + (1-p)^{1+\alpha}], \tag{4.40}$$

where $\gamma = \delta - \log|\mathcal{K}|$. We need to maximize $G_I(\alpha, \gamma, p)$ with respect to α where $0 < \alpha \leq 1$. δ is the minimum entropy of the sampled data frame from which key Q was extracted. To extract key we need to have $\gamma > 0$, i.e. for a larger k, larger δ would be needed. Therefore, in our numerical analysis we make this assumption that γ takes a constant positive value. As a result, the optimization problem can be formulated as

$$\max_{0<\alpha\leq 1} G_I(\alpha, \gamma, p), \quad \text{where } \gamma > 0 \text{ and } G_I(\alpha, \gamma, p) > 0.$$

We use numerical optimization through an exhaustive search over $0 < \alpha^* \leq 1$ with the step size of 10^{-4}, and denote the optimized order as α^* and the maximized lower-bound as G_I^{max}. As the secrecy requirement for PA mode, we determine a threshold for secrecy exponent as G_I^{th} and find the minimum required cipher key length for which $G_I^{\text{max}} \geq G_I^{\text{th}}$ as

$$\log|\mathcal{K}| \geq \frac{1}{\alpha}\left(G_I^{\text{th}} + e^{-\alpha^*\gamma} + b\log[p^{1+\alpha^*} + (1-p)^{1+\alpha^*}]\right) + 2\log M. \tag{4.41}$$

We denote this required lower-bound for key length in PA mode as $\Gamma_{pa}(p, b, G_I^{\text{th}}, \alpha^*, \gamma)$.

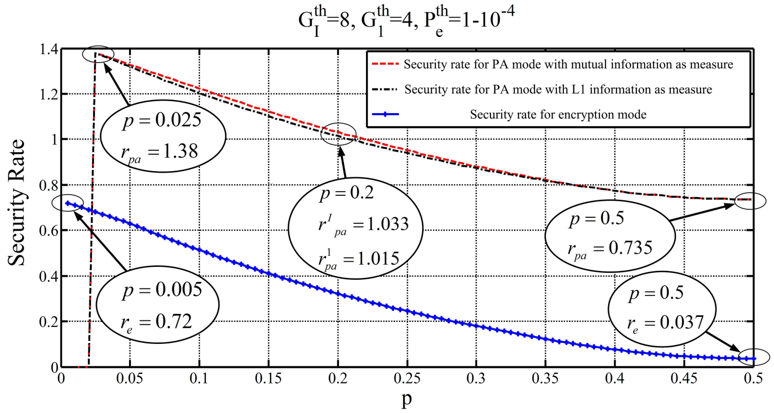


FIGURE 4.5. Security rate in different modes with different metrics

Considering L_1 norm distance as secrecy metric, according to Eq. (4.32), function G_1 representing the lower bound for decreasing exponent of information leakage turns out to be

$$G_1(\alpha,\gamma,p) = \frac{1}{1+\alpha}\Big(\alpha(\log|\mathcal{K}| - 2\log M) - e^{-\alpha\gamma} - b\log[p^{1+\alpha} + (1-p)^{1+\alpha}]\Big). \tag{4.42}$$

Similarly G_1 can be maximized for $0 < \alpha \leq 1$ and $G_1 > 0$. Then, for optimized α^*, if we determine the required threshold for G_1^{max} as G_1^{th}, we can obtain the requirement for the key length to have $G_1^{\text{max}} \geq G_1^{\text{th}}$ as:

$$\log|\mathcal{K}| \geq \frac{1}{\alpha}\left((1+\alpha)G_1^{\text{th}} + e^{-\alpha^*\gamma} + b\log[p^{1+\alpha^*} + (1-p)^{1+\alpha^*}]\right) + 2\log M. \tag{4.43}$$

Let security rate be the minimum required key length for transmission of each message symbol or bit with the required security. In encryption mode cipher keys are generated per message block of length n-bit, hence security rate that meets the secrecy requirement is $\Gamma_e(p,n,P_e^{th},\alpha,\gamma)/n$. In PA mode each generated key is applied per message block of length b-bit, therefore security rate will be $\Gamma_{pa}(b,\alpha^*,p,\gamma,G_I^{\text{th}})/b$. Fig. 4.5 depicts security rate in terms of varying Bernoulli parameter $0 < p < 0.5$ for i.i.d. input binary distribution. It shows security rates that in PA mode satisfy the required lower-bound for secrecy exponent based on mutual information (denoted by r_{pa}^I) or variational distance (denoted by r_{pa}^1). It also illustrates security rate in encryption mode that is denoted by r_e that meets the demanded block error rate for cryptanalyst. G_I^{th} is chosen to be 8 and G_1^{th} is set to be half of it due to the inequality (4.37), meanwhile we select P_e^{th} to be $1 - 10^{-4}$. The required security rate for PA mode is much higher than encryption mode. As can be seen, in PA mode for $p = 0.05$ the required security rate is relatively high about 1.35 while with increase in p, it goes down to 0.735. Note that the results obtained for both metrics in PA mode are almost the same with slight differences for $0.05 < p < 0.3$. That is because mutual information is a stronger metric compared to L_1 distance, and has a looser lower bound that will require a slightly higher security rate to meet the secrecy requirement.

When both modes of operations are utilized in dual transmission mechanism, it is necessary to simultaneously satisfy their demanded secrecy. That allows us to use the same key stream generation rate R_s defined as $\frac{\log|\mathcal{K}|}{n}$ in both operational modes that needs to satisfy

$$R_s \geq \max\left[\Gamma_{pa}(p, b, G_I^{\text{th}}, \alpha^*, \gamma), \Gamma_e(p, n, P_e^{th}, \alpha, \gamma)\right]/n.$$

Fig. 4.6 shows how required key stream rate for dual mode transmission varies with respect to the parameter p of input binary distribution and conversion rate of inverse universal hashing (b/n). Note that low conversion rate indicates that more redundancy is added through inverse hashing. Exponential secrecy in PA mode is much stronger than the error probability metric in encryption mode, so in most areas key length is determined by the secrecy criterion in PA mode. However, in circled area the necessary key stream rate is determined based on the secrecy requirement of encryption mode. That is because for low conversion rate (b is much smaller than n) and low source entropy (low p) the required key length for strong secrecy of b-bit message in PA mode might not be sufficient even for weak secrecy of n-bit message in encryption mode.

4.8 Conclusion

In this chapter we designed a secrecy system, based on a general cipher as the first layer and privacy amplification as the second layer, that is exponentially secure, namely, its information leakage decays at exponential rate. Without resorting to any physical channel condition or restriction on Eve's resources, the only advantage that we considered for legitimate users was a key source that is partially known by Eve. A key extracting module derives keys from this source about which

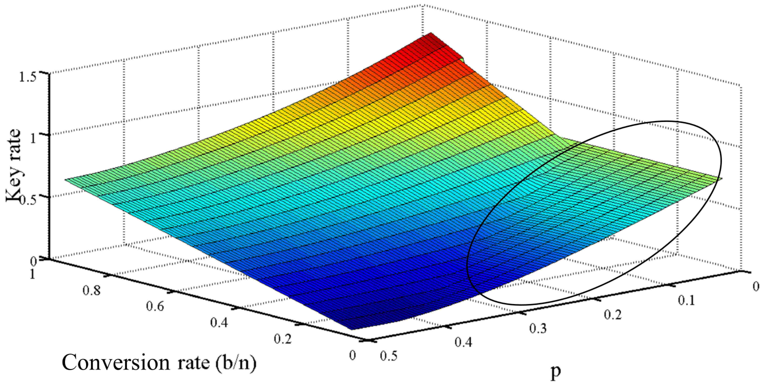


FIGURE 4.6. Required key stream generation rate in dual mode transmission

Eve has the required uncertainty. We characterized and optimized the lower bound for decreasing exponent of information leakage called secrecy exponent in terms of Eve's information and distinguishability. Then, it is adopted as the criterion to determine the minimum required key length for encryption. In numerical analysis we considered a dual mode transmission mechanism with two levels of secrecy based on which the required key stream rate for different source entropies is evaluated.

Chapter 5
Future Work

5.1 Multiuser and Asynchronous Key Scheduling

In Chapter 3 we considered a protocol in which the legitimate users are 100% confident about the shared randomness by utilizing the proposed synchronization scheme. However, in many applications there can be some synchronization errors between two parties, and since they both need to apply a privacy amplification technique over the generated common randomness that can be erroneous, the derived secret keys will be different that causes decryption error when these keys are used for symmetric encryption. Moreover, approaches that require complete synchronization between two parties will result in a higher communication overhead and lower secrecy throughput. As a result, if we relax the requirement of absolute agreement between targeted recipient and the transmitter by allowing bounded error pattern for shared data while still generating the same key, we can achieve a secure key which consequently produces a higher secrecy throughput with a less expenditure and overhead. Hence, we need to find a random extraction method, that with a similar input but with a margin of difference from the original one, generates the same key. Moreover, the effect of relaxed requirements in synchronization needs to be illustrated in improvement of system efficiency.

In Chapter 3 to simplify the secrecy analysis problem, we assumed there exists a virtual oracle giving Eve information regarding where she has lost her alignment with users. Nevertheless, for rigorous analysis we should note that in reality there does not necessarily exist such an oracle, so we will need to either adopt analysis based on a new metric or completely analyze possibility of realignment. In addition, the mapping algorithm, that we proposed in this work to cause error propagation for Eve, results in a constant distance between the original set and her destination set after resynchronizations of her OTF with users' set. If we assume that Eve is able to guess where she has got back her OTF synchronization, she will be able to find this constant distance for the rest of her destination set, remove it from her set and consequently obtain a set with a much lower Hamming distance from the actual set and with a less entropy about the generated key. Therefore, a new way of analysis and a new metric or some modification in generating the key and randomness is required to address these issues with a more concrete analysis.

With increasing application of point to multipoint communication over broadcast links such as file distribution, teleconferencing and video text systems, this trend will continue in future communications. Due to poor throughput efficiency of pure stop and wait ARQ protocols in systems where channel round trip delay is large, and where there are a large number of receivers, they can not be used in systems like satellite broadcast channels. In other words, if we try to extend our proposed secrecy approach in Chapter 3 to broadcast communications, we will need to change it in a way that can fit in the new channel conditions. Overall, a new or modified approach for generating secret keys that tolerates a margin of errors between legitimate users and meanwhile generates a highly secure key with a higher throughput is needed. This system should also be designed for sharing secret keys between a base station and multiple users with a high secrecy and data efficiency.

5.1.1 Problem with the Proposed Mapping Strategy

In Chapter 3 we discussed about a mapping strategy based on the first order IIR filter that can be used to map the generated OTF set into another set called destination set about which Eve will have a higher uncertainty. Then, we can make sure that every missing or wrong OTF packet behaves like an additive noise that propagates for the rest of packets in Eve's destination set after misdetection. The problem is that if we use first order mapping, every noisy packet will contribute with a constant weight for the difference between Eve's destination set and the original set. Accordingly, Eve needs to simply estimate this constant difference and error between these two sets in order to recover the original set. To prevent Eve from successfully mounting such kind of attack in guessing possible packets that act like additive noise in her destination set, we may need to design a more sophisticated mapping strategy that minimizes the possibility of error estimation by Eve. In this new mapping strategy we should take into account that misdetected packets should contribute with varying weight in error propagation of Eve's set to further intensify her confusion.

5.2 Proposed Approach

5.2.1 Asynchronous Key Agreement Based on Fuzzy Extractors and Edit Distance

In [59] a key generation technique is proposed based on fuzzy extractors that extract a uniformly random string R from its input w. This extraction is error-tolerant such that R will be the same even when the input changes, as long as it remains within a certain distance from the original. To assist in recovering R from w', fuzzy extractor produces a public string P. However, still given P, R remains uniformly random. As a result, if we design our scheme in a way that with a fewer transmissions or overheads, the distance between gathered random data for both partners is upper-bounded, we can use fuzzy extractor to derive the same keys for them. Since fuzzy extractor tolerates errors within this upper-bound of distance between two inputs, with the help of public string equivalent keys with a high min-entropy can be generated, to make sure that with a higher secret key rate the required level of secrecy can be achieved.

If we consider possibility of errors in the proposed scheme in Chapter 3, every mistake in putting a wrong packet in OTF or missing an OTF packet can cause a misalignment between two parties and make the rest of their sets different. Thus, in this case it is not appropriate to use Hamming distance as a metric to measure distance between two strings. We therefore need to tailor to a metric that measures distance based on insertion and deletion of packets. In [59] Hamming distance, edit distance as well as set difference are used to measure distance from the original input data. Edit distance between w and w' is defined to be one half of the smallest number of character insertions and deletions needed to transform w into w'. If we use edit distance instead of Hamming distance to measure the difference between two gathered random strings by Alice and Bob, we do not need to worry about asynchronization between them since what needs to be measured is the number of packets inserted or deleted to transform one set to another. In [59] a fuzzy extractor based on edit distance is constructed that allows creation of the same keys when edit distance of two sets is less than a threshold.

In this new context of using fuzzy extractors, we need to make sure that with a high probability Eve's gathered string will have a higher edit distance than the determined threshold. Due to using a new difference metric, no longer can misalignment be a problem for Eve, yet what really matters here is the number of mistakes that she probably makes by OTF packet missing or false OTF packet error. Thus, the system has to be designed in a way that with a high probability the number of Eve's miss-detections exceeds the required upper-bound for edit distance with the original set, to make sure that even with the use of public information she can not generate the same key. This can be ensured by taking advantage of statistical independence between legitimate users and adversary's channels as well as an authenticated but insecure feedback channel between Bob and Alice modeled as a binary erasure channel. Note that in this scenario there will not be any need for a virtual oracle as it was assumed in Chapter 3 to analyze secrecy of the system.

5.2.2 More Efficient ARQ Protocols and Broadcasting Scenario

Stop and wait ARQ protocol that we utilized in our key management algorithm is not efficient in real-life implementation and using it brings about a low throughput for communications. There are some other more efficient ARQ protocols like Go-back-N and selective repeat ARQ. In Go-back-N protocol, the transmitter continues sending a number of frames specified by a window size even without receiving an ACK from receiver which generates a higher throughput compared to stop and wait protocol. Selective repeat ARQ results in even a higher efficiency because unlike Go-back-N, in this protocol after a lost frame, the sender continues to send a number of frames specified by its window size, and the receiver accepts and acknowledges packets after a transmission error. If we adopt either Go-back-N or selective repeat ARQ to achieve a higher transmission throughput in our key scheduling scheme, OTF gathering strategies for Alice and Bob have to change, and a new secrecy analysis will be required. In this analysis, we need to take into account communication efficiency and throughput, quality of secrecy (secrecy outage probability) and efficiency of secrecy establishment (secret key rate). Furthermore, the trade-off between these metrics is needed to be studied.

Another issue is how to extend ARQ based key scheduling to broadcasting scenario for the purpose of sharing keys between base station and users. In order to prevent disadvantages of using simple ARQ in broadcast channels including throughput deficiency, we can utilize Hybrid ARQ (HARQ) mechanism that takes advantage of both error detection and error correction to deliver a higher throughput. In [101] different HARQ schemes are proposed to be used in broadcast channel that can provide acceptable throughput for the system. In [102] and [103] two different secure HARQ schemes are presented that make use of the exiting potential in HARQ mechanism to improve security of the system by combining encoding and symmetric key encryption in one step called secrecy encoder. As a result, instead of pure stop and wait protocol we used in Chapter 3, we can design a HARQ mechanism that along with a high secrecy efficiency, provides the required throughput over multiuser channel. Note that a new secrecy design and analysis for key scheduling step is required since a new HARQ protocol is exploited by considering both error correction ability of HARQ and multiuser scenario. Then, the obtained secret keys can be utilized in a well-designed secrecy encoding that is a secret-key-based randomized encoder to ensure reliability as well as highly confidential message transmission. In this scenario each user generates its keys based on the statistical data obtained from HARQ mechanism when it is guaranteed that their gathered data maintains its required edit distance from the original set through feedback messages that is independently transmitted by each user.

5.2.3 New Non-Invertible Mapping Strategy

Let us consider a higher order IIR filter instead of a first order filter used for mapping OTF set into a destination set. Let $X(n)$ and $W(n)$ be respectively the input OTF packet and output destination packet at time n. If for instance we consider a second order IIR filter with the following relationship between input and output packets as

$$W(n) = b_0X(n) + b_1X(n-1) + b_2X(n-2) + a_1W(n-1) + a_2W(n-2), \tag{5.1}$$

the transfer function for this second order IIR filter can be written as

$$H(Z) = \frac{B(Z)}{A(Z)} = \frac{b_0 + b_1Z^{-1} + b_2Z^{-2}}{1 - a_1Z^{-1} - a_2Z^{-2}}. \tag{5.2}$$

In this case the first output packet will be $W(1) = b_0X(1)$ and the second one $W(2) = b_0X(2) + (b_1 + a_1b_0)X(1)$. As can be seen, $X(1)$ has weight b_0 for $W(1)$ and weight $b_1 + a_1b_0$ for $W(2)$. Namely, by using a higher order IIR filter we can infer that each input packet contributes with a varying weight to each output packet. Basically, by using a higher order IIR filter we cause more confusion for Eve such that she has to take a more sophisticated attack strategy rather than just simply estimating an error that appears as a constant difference between her set and the original set after misdetection. We should note that in case of using higher order IIR filter, we should also take into account the issue of stability and make sure that coefficients of this transfer function are designed in a way that stability is guaranteed.

If a higher order IIR filter is used as mapping strategy, Eve needs to first deal with the ambiguity problem to know where errors have occurred and then estimate error sequence in her set. If we consider that each misdetected packet acts like an additive noise in Eve's destination set, in case of using a higher order IIR filter, it is like this additive noise goes through a channel with transfer function of the utilized digital filter. As a result, a possible attack strategy would be applying inverse mapping in order to estimate this noise and to recover the original OTF packets. In this strategy Eve models these misdetected packets in her OTF set as additive noise that goes through this filter. Therefore, she can use some estimation techniques like Kalman filter or MMSE (Minimum Mean Square Error) to estimate the existing error in her destination set and then to apply inverse mapping in order to find the original packet errors in her guessed OTF set.

As a possible approach, if we design $A(Z)$ and $B(Z)$ in denominator and nominator of the filter transfer function as feedback and feedforward filters to control their coefficients, we can have a mapping strategy that is invertible impossible. By adopting this non-invertible mapping, we ensure that Eve will not be able to apply inverse operation to estimate the original packet errors. Namely, singularity in mapping obstructs Eve from estimating original errors in her OTF set. Moreover, since Eve is not able to directly communicate with Alice, it is not possible for her to resolve this singularity problem. On the other hand, Bob can communicate with Alice and use reconciliation strategy or, in case of using fuzzy extractor, utilize received helper string to correct possible errors in her gathered OTF set. This two-way communication guarantees that singularity in mapping will not cause any problem for legitimate users such that they can be confident of generating the same OTF sets that later on will be mapped to the same destination sets resulting in exactly the same secret keys after universal hashing.

5.2.4 Novelty

The most important innovation in this work is using fuzzy extractors in order to allow some margin of difference between generated random sets in the receiver and transmitter measured with edit distance. This allows us to design a system with less required overhead and thus increased secrecy throughput. Meanwhile, it will make secrecy analysis of the system more straightforward by removing the necessity to have an oracle and focusing on only the number of miss-detections by Eve. On the other hand, using a higher order IIR filtering that is singular prohibits Eve from applying inverse mapping in order to estimate her original packet errors unlike the previously proposed mapping that allows her to just estimate original errors that appear as a constant difference between her set and the original set.

Another novelty of this proposed scheme is extending the key scheduling algorithm proposed in Chapter 3 to broadcast channels by utilizing efficiency of the application of HARQ transmission mechanism. We can also utilize potentiality of HARQ to design a secure HARQ by converting its pure encoder into a secrecy encoder providing both error correction and security based on the generated key stream in previous transmissions. The new analysis for both secrecy outage rate and secrecy throughput will be required for the newly designed scheme in the multiuser framework.

Chapter 6
Appendix

6.1 Proof of Theorem 2

Proof. Suppose that all possible 2^{56-a} key candidates are arranged as $k_1, k_2, \ldots, k_{2^{56-a}}$ from the lowest rank to the highest. Let H_i be the hypothesis that k_i is the original key and $g_i = 1$ be the event that Eve decides that k_i is correct. We define a Bernoulli random variable B which is equal to 1 when the right key is among top 2^{56-a} candidates, and 0, otherwise. Thus, $Pr[B = 0] = 1 - P_s$ and $Pr[B = 1] = P_s$. Let P_c be the total success probability for Eve. Note that when $B = 0$, the right key will not be tested and consequently can not be found. Therefore, we have

$$P_c = \sum_{i=1}^{2^{56-a}} Pr[g_i = 1, H_i|B = 1].Pr[B = 1]. \tag{6.1}$$

The probability that Eve can realize the right key k_i is

$$Pr[g_i = 1, H_i|B = 1] = Pr[g_i = 1|H_i, B = 1].Pr[H_i|B = 1].$$

For Eve to be able to find the correct key at rank i, since she starts the test from upper ranks to the lower ones, there should not be any false key acceptance for ranks higher than i, as well as a key missing event for rank i. Hence,

$$Pr[g_i = 1|H_i, B = 1] = (1 - P_F)^{2^{56-a}-i}(1 - P_m). \tag{6.2}$$

Moreover, Decisions about all 2^{56-a} keys are independent, and all of the tested keys are equally probable to be the right one, i.e. $Pr[H_i|B = 1] = \frac{1}{2^{56-a}}$. Therefore, by using Eq.'s (6.1) and (6.2), we obtain Eq. (2.8) for total success probability.

The next step is to compute the frame erasure probability. Assume that the right key is k_i and is located among top 2^{56-a} candidates. In order to obtain no key, Eve should not have any false key admission for k_j, $j \neq i$ for $i, j = 2^{56} - 2^{56-a} + 1, \ldots, 2^{56}$, i.e. top 2^{56-a} candidates except the right key itself, and in addition to that she has to miss the right key k_i. When k_i is not among top candidates, since it will not be examined, Eve gets nothing provided that there has been no wrong key acceptance event for top 2^{56-a} tested candidates. As a result, frame erasure probability can be computed according to Eq. (2.9). By a similar technique, we can prove that the wrong key probability is $P_w = 1 - P_e - P_c$. □

6.2 Proof of Lemma 3

Proof. We need to compute vector transition probabilities between all possible input and output vectors X and Y for states S_1 and S_3. Hence, for $k = 1, 3$

$$Pr[Y|X, S_k] = Pr[X \oplus Y|X, S_k] = Pr[E|S_k], \tag{6.3}$$

where E is the decryption error vector which is bit-wise Xor of input and output vectors. The last equality is because E depends on channel errors in previous and current ciphertexts, so given the state, it is independent from input vector X. To analyze states S_0 and S_1, we define two events, A and B as

$$\begin{aligned} A: &\quad \text{There exists at least one bit error in} \quad \hat{C}_i \\ B: &\quad \text{There exists at least one bit error in} \quad \hat{C}_{i-1}. \end{aligned}$$

As a result, $S_1 = A \cap \bar{B}$, and we can write

$$Pr(E|S_1) = Pr(E|A, \bar{B}) = \frac{Pr(E|\bar{B})Pr(A|E, \bar{B})}{Pr(A|\bar{B})}. \tag{6.4}$$

The fact that events A and B are caused by two independent channel error vectors Z_{i-1} and Z_i implies that A is independent of B and its complementary, i.e. $Pr(A|\bar{B}) = Pr(A) = q$. When event B has not occurred, since only $\hat{C}_i$ can induce bit errors with rate of η into the stored plaintext, the probability that a particular decryption error vector E with Hamming weight of $W(E)$ takes place will be

$$Pr(E|\bar{B}) = \eta^{W(E)}(1-\eta)^{64-W(E)}. \tag{6.5}$$

In state S_1, HW of error vector E can not be zero because we know that the only source that can induce error at stage i is Z_i that surely has a non-zero bit. In this case, given an error vector E with $W(E) \neq 0$ and knowing that event B did not occur, we can infer that this error is induced by error in $\hat{C}_i$, hence event A has certainly occurred, i.e. $Pr(A|E, \bar{B}) = 1$. Thus, using equations (6.3), (6.4) and (6.5), we can obtain the input-output transition probability in S_1 as in Eq. (2.12). Similarly, for state S_3, $Pr(Y|X, S_3)$ can be computed with the detailed proof provided in our technical report [104]. □

6.3 Proof of Lemma 4

Proof. If we assume that all 2^{64} possible input plaintexts are equally likely, for $l = 1, 3$ we can write

$$\begin{aligned} H(Y|S_l, X) &= -\sum_i \sum_j \frac{1}{2^{64}} Pr(Y = Y_j|X = X_i, S_l). \log Pr(Y = Y_j|X = X_i, S_l) \\ &= -\sum_{j=1}^{2^{64}} Pr(E = E_j|S_l) \log Pr(E = E_j|S_l). \end{aligned} \tag{6.6}$$

The second equality is resulted from Eq. (6.3) for $E_{i,j} = X_i \oplus Y_j$ as the decryption error vector. Furthermore, for state S_1 as discussed in subsection 2.4.1, HW of the error vector E can not be zero. Thus, we can take E_1 as a 64-bit zero vector and exclude it from this summation. Then, using Eq. (2.12) brings about the following result

$$H(Y|S_1, X) = \frac{-1}{q} \sum_{j=2}^{2^{64}} \eta^{W(E_j)} (1-\eta)^{64-W(E_j)}. \log \left[\frac{\eta^{W(E_j)} (1-\eta)^{64-W(E_j)}}{q} \right]. \tag{6.7}$$

We know that out of all 2^{64} error vectors, the number of possible vectors with HW of W is the number of possibilities of choosing W bits out of 64 bits which is equal to W-combinations from 64 elements. Finally, Eq. (6.7) can be rewritten as Eq. (2.18) by excluding $k = 0$. For state S_3, we compute $H(Y|S_3, X)$ using Eq. (6.6) for $l = 3$. In this case, $j = 1$ is not excluded because unlike state S_1 in state S_3, it is possible to have decryption error vector E_1 with zero weight. Thus, by using Eq. (2.13), we obtain $H(Y|S_3, X)$ in Eq. (2.19). □

6.4 Proof of Theorem 3

Proof. Let E_m^i and e_m^i be the random variables E_m and e_m, respectively, associated with one-time transmission and feedback reception of packet i by Bob. Also, let E_w^i and e_w^i be the random variables E_w and e_w, associated with one-time transmission and feedback reception of packet i_E by Eve. We assume that transmission of each packet and its associated feedback is independent for different packets while their corresponding events of correct receptions or failures for different packets are equally likely. Thus, at final steps of the following proofs we can replace E_m^i, e_m^i, E_w^i and e_w^i with E_m, e_m, E_w and e_w, respectively. Eq. (3.6) can be rewritten as

$$Pr[H_1|F_{i_E}, E_w^i = 0] > \frac{1}{2}. \tag{6.8}$$

Since hypothesis H_1 occurs when $F_i = 1$ and packet i is received correctly by Bob, using Bayesian rule we get

$$Pr[H_1|F_{i_E}, E_w^i = 0] = Pr[F_i = 1|E_m^i = 0, E_w^i = 0, F_{i_E}]Pr[E_m^i = 0, F_{i_E}|E_w^i = 0]. \tag{6.9}$$

In this Eq. since the erasure in the received feedback by Bob, i.e. F_i, is independent from the erasure in Eve's received packet, the first term can be written

$$Pr[F_i = 1|E_m^i = 0, F_{i_E}] = \frac{Pr[F_i = 1, F_{i_E}|E_m^i = 0]}{Pr[F_{i_E}|E_m^i = 0]}. \tag{6.10}$$

First of all, we consider the case where Eve has received feedback $F_{i_E} = 1$. Then, the second term in Eq. (6.9) will be one because receiving feedback $F_{i_E} = 1$ by Eve implies that it was initially received error-free by Bob. By using Eq. (6.10), we can rewrite the first term in Eq. (6.9) as

$$Pr[F_i = 1|E_m^i = 0, F_{i_E} = 1] = \frac{Pr[e_m^i = 0, e_w^i = 0|E_m^i = 0]}{Pr[e_w^i = 0|E_m^i = 0]} = \frac{q_{00}}{1-\theta} = \frac{1-\eta-\theta+q_{11}}{1-\theta}. \tag{6.11}$$

The second equality is resulted from the definition of joint backward erasure probabilities. The third equality comes from relationships $q_{00} + q_{01} = Pr[e_m^i = 0] = 1 - \eta$ and $q_{01} + q_{11} = Pr[e_w^i =$

$1] = \theta$, with q_{11} given in Eq. (3.4). Thus, by Eq's (6.9) and (6.11), we can write the decision rule (6.8) as

$$\left[1-\eta+\frac{\psi\sqrt{\eta\theta(1-\eta)(1-\theta)}}{1-\theta}\right]-\frac{1}{2}>0. \tag{6.12}$$

This is equivalent to $\Gamma > 0$ as it was defined in (3.7).

Next, suppose that Eve has received an erased feedback $F_{i_E} = e$. Due to independence of the packet reception by Bob and feedback reception by Eve, we can show that the second term of Eq. (6.9) will be

$$Pr[E_m^i = 0|E_w^i = 0] = \frac{Pr[E_m^i = 0, E_w^i = 0]}{Pr[E_w^i = 0]} = \frac{p_{00}}{1-\varepsilon} = \frac{1-\delta-\varepsilon+p_{11}}{1-\varepsilon}. \tag{6.13}$$

Similarly, by Eq. (6.10), the first term in Eq. (6.9) will be

$$Pr[F_i = 1|E_m^i = 0, F_{i_E} = e] = \frac{\theta - q_{11}}{\theta}. \tag{6.14}$$

Now by replacing Eq.'s (6.13) and (6.14) into Eq. (6.9), we can get the decision rule in Eq. (6.8) as

$$\left[1-\delta+\frac{\rho\sqrt{\varepsilon\delta(1-\delta)(1-\varepsilon)}}{1-\varepsilon}\right]\left[1-\eta-\frac{\psi\sqrt{\eta\theta(1-\eta)(1-\theta)}}{\theta}\right]-\frac{1}{2}>0. \tag{6.15}$$

According to the definition of Λ in (3.8), it is equivalent to the decision rule $\Lambda > 0$ for $F_{i_E} = e$. □

6.5 Proof of Lemma 5

Proof. According to the definition of OTF packet missing probability, by using Bayesian rule we have

$$\begin{aligned} P_m =& Pr[i_E \notin \text{OTF}_E|H_1] \qquad (6.16)\\ =& Pr[i_E \notin \text{OTF}_E|H_1, F_{i_E} = 1, E_w^i = 0]Pr[F_{i_E} = 1, E_w^i = 0|H_1]\\ +& Pr[i_E \notin \text{OTF}_E|H_1, F_{i_E} = e, E_w^i = 0]Pr[F_{i_E} = e, E_w^i = 0|H_1]\\ +& Pr[i_E \notin \text{OTF}_E|H_1, E_w^i = 1]Pr[E_w^i = 1|H_1], \end{aligned}$$

where OTF$_E$ denotes Eve's chosen OTF set, and '$E^i_w = 1$' means Eve did not receive i_E correctly. By definition, H_1 is equivalent to the event $[e^i_m = 0, E^i_m = 0]$, so we have

$$\begin{aligned} Pr[F_{i_E} = 1, E^i_w = 0|H_1] &= Pr[e^i_w = 0|e^i_m = 0]Pr[E^i_w = 0|E^i_m = 0] \\ &= \frac{q_{00}}{1-\eta}.\frac{p_{00}}{1-\delta} = \frac{1-\eta-\theta+q_{11}}{1-\eta}.\frac{1-\delta-\varepsilon+p_{11}}{1-\delta}, \end{aligned} \tag{6.17}$$

where p_{11} and q_{11} are given in Eq. (3.4). Similarly, we can show

$$Pr[F_{i_E} = e, E^i_w = 0|H_1] = \frac{(\theta - q_{11})[1-\delta-\varepsilon+p_{11}]}{(1-\eta)(1-\delta)}. \tag{6.18}$$

We can compute the last term in Eq. (6.16) as

$$Pr[E^i_w = 1|E^i_m = 0, F_i = 1] = \frac{\varepsilon - p_{11}}{1-\delta}. \tag{6.19}$$

For a correctly received packet by Eve, with the received feedback as $F_{i_E} = 1$, she will not put packet i_E into OTF$_E$ if $\Gamma < 0$. If $F_{i_E} = e$, i_E will not belong to Eve's OTF if $\Lambda < 0$. Apparently, when '$E^i_w = 1$', regardless of what the received feedback would be, she has no way to put i_E in OTF$_E$. Hence,

$$\begin{aligned} &Pr[i_E \notin OTF_E|H_1, F_{i_E} = 1, E^i_w = 0] = \mathbf{1}_{(\Gamma<0)} \\ &Pr[i_E \notin OTF_E|H_1, F_{i_E} = e, E^i_w = 0] = \mathbf{1}_{(\Lambda<0)} \\ &Pr[i_E \notin OTF_E|H_1, E^i_w = 1] = 1. \end{aligned} \tag{6.20}$$

By replacing Eq.'s (6.17)-(6.20) into Eq. (6.16), we can get the formula for P_m in Eq. (3.9).

To compute the false OTF probability which is $P_F = Pr[i_E \in \text{OTF}|H_0]$, we split hypothesis H_0 into two events: H_{01} when packet i is received incorrectly by Bob, and H_{02} when i is received without error, but $F_i = e$. It should be noted that according to Eve's strategy, false detection event only occurs when $SR_{i_E} = 0$ and $P_{i_E+1} \neq P_{i_E}$ because she only cares about fresh packets. We define P_{F_1} as the false OTF probability when H_{01} takes place, which is

$$\begin{aligned} P_{F_1} =& Pr[i_E \in \text{OTF}|H_{01}] \qquad (6.21) \\ =& Pr[i_E \in \text{OTF}|H_{01}, F_{i_E} = e, E^i_w = 0]Pr[F_{i_E} = e, E^i_w = 0|H_{01}]. \end{aligned}$$

That is because when H_{01} occurs, since Bob has not decoded i correctly, he will send back a Nack which can be received either erased bit or zero that in the latter case Eve will certainly not put it into OTF. According to Eve's strategy in Tab. 3.4, for a correctly received packet i_E once receiving $F_{i_E} = e$, Eve puts i_E into OTF if $\Lambda > 0$, so

$$Pr[i_E \in \text{OTF}|H_{01}, F_{i_E} = e, E_w^i = 0] = \mathbf{1}_{(\Lambda>0)}. \tag{6.22}$$

Moreover, the erasure event in the received feedback by Eve is independent of the reception of packet i_E and i. As a result, the second term in Eq. (6.21) will be

$$Pr[F_{i_E} = e, E_w^i = 0|E_m^i = 1] = Pr[E_w^i = 0|E_m^i = 1]Pr[e_w^i = 1] = \frac{p_{10}}{\delta}\theta = \frac{(\delta - p_{11})\theta}{\delta}. \tag{6.23}$$

Therefore, we have

$$P_{F_1} = \mathbf{1}_{(\Lambda>0)}\frac{(\delta - p_{11})\theta}{\delta}. \tag{6.24}$$

We also define P_{F_2} as the false OTF probability when H_{02} occurs. We can similarly show that

$$P_{F_2} = \frac{[\mathbf{1}_{(\Lambda>0)}q_{11} + \mathbf{1}_{(\Gamma>0)}(\eta - q_{11})](1 - \delta - \eta + p_{11})}{\eta(1-\delta)}. \tag{6.25}$$

Now, we can obtain the total false OTF probability as

$$P_F = P_{F_1}Pr[H_{01}] + P_{F_2}Pr[H_{02}] = P_{F_1}\delta + P_{F_2}(1-\delta)\eta,$$

replacing P_{F_1}, P_{F_2} from Eq.'s (6.24), (6.25) completes the proof. □

6.6 Proof of Theorem 4

Proof. Let X_i denote the packets in OTF for $i = 1, \ldots, n_{ts}$, and Y_i indicate the packets in Eve's OTF. We denote the number of Alice and Bob's Bernoulli trials between $i-1^{st}$ and i^{th} successes in putting in OTF as T_i. T_i's are i.i.d. random variables with geometric distribution. Let N_e denote the number of mismatches between two destination sets. Outage probability is defined as the probability that there exists less than l packet discrepancies between two destination sets that occurs when there is at least $n_{ts} - l + 1$ packets to be the same for V and W. It means that misalignment would happen after $n_{ts} - l + 1^{\text{th}}$ packet in OTF$_E$. Hence, we have,

$$\begin{aligned}
P_{out} =& Pr(N_e < l) \overset{(1)}{=} Pr[X_i = Y_i, i \in \{1, \ldots, n_{ts} - l + 1\}] \\
&\overset{(2)}{=} \prod_{i=1}^{n_{ts}-l+1} Pr[X_i = Y_i] \\
&\overset{(3)}{=} \prod_{i=1}^{n_{ts}-l+1} \sum_{t=1}^{M-\sum_{j=1}^{i-1} T_j} Pr[X_i = Y_i | T_i = t] Pr[T_i = t] \\
&\overset{(4)}{\leq} \prod_{i=1}^{n_{ts}-l+1} \sum_{t=1}^{M} Pr[X_i = Y_i | T_i = t] Pr[T_i = t] \\
&\overset{(5)}{=} \prod_{i=1}^{n_{ts}-l+1} \sum_{t=1}^{M} (1 - P_m)(1 - P_F)^{t-1} P_c (1 - P_c)^{t-1} \\
&\overset{(6)}{\leq} \left[\frac{(1 - P_m) P_c}{1 - (1 - P_c)(1 - P_F)} \right]^{n_{ts}-l+1}.
\end{aligned} \tag{6.26}$$

Equality (2) is because the decision that receiver makes about each packet is independent of other packets. Equality (3) is based on Bayesian rule by summing over all possible number of trials for each Bernoulli success. For the first success it can reach to the total number of packets within the frame, i.e. M, but for the next ones, we should subtract the number of all previous trials. Equality (5) holds since to have $X_i = Y_i$, neither should there be missing OTF event for Eve for packet X_i at the i^{th} Bernoulli success nor any false detection event for the rest of unsuccessful OTF Bernoulli events that are totally $T_i - 1$ trials. Conclusively, inequalities (4) and (7) show that Eq. (3.13) provides an upper-bound for P_{out}. □

6.7 Universality of $\odot$ multiplication in GF(q^n)

Proof. We first prove Condition 1 of universality for the function h_s defined as $h(S, X) = \text{trunc}_b[S \odot X]$. For the collision probability of this function we can write

$$\begin{aligned}
Pr[S \in \mathcal{SD} : h(S, X) = h(S, X')] &= Pr\Big[S \in \mathcal{SD}, \exists R \in \Omega^{n-b} \backslash 0^{n-b} : S \odot (X \oplus X') = (0^b, R)\Big] \\
&\leq (q^{n-b} - 1) \frac{1}{q^n - 1} \leq \frac{1}{q^b}.
\end{aligned}$$

Since $X \oplus X' \neq 0$, we can find at most one $S \in \Omega^n \backslash 0^n$ for which $S \odot (X \oplus X') = (0^b, R)$ with $R \neq 0$. The last inequality holds since for $a \leq b$ we have $\frac{a-1}{b-1} \leq \frac{a}{b}$.

For the second condition consider a randomly chosen S. We can see that $h^{-1}(R, S, A) = S^{-1} \odot (A||R)$ is uniformly distributed over the preimage set $\mathcal{X}_A = \{X \in \Omega^n, h(X, S) = A\}$ which has cardinality of q^{n-b}, implying that $|h_s^{-1}(A)| = q^{n-b}$. That is because a uniformly chosen R over the set Ω^{n-b} determines which $X \in \mathcal{X}_A$ generates $(A||R)$ after multiplication by S. There exists q^b such preimage sets that are disjoint and have cardinality that does not depend on A. Contrarily, if there exists an element in both $\mathcal{X}_A$ and $\mathcal{X}_{A'}$ with $A \neq A'$, it means that $S^{-1} \odot (A||R) = S^{-1} \odot (A'||R')$. For $R \neq R'$ it is impossible to hold, but for $R = R'$ it requires that $A = A'$ which is contradictory. Therefore, $|h_s^{-1}(A)| = |\mathcal{X}_A| = q^{n-b}$ which does not depend on A. □

6.8 Proof of Theorem 5

Proof. It is sufficient to prove that the statement holds for $E_s H_{1+\alpha}(Q)$, where E_s denotes the expectation over S. Then, we can find a function h_s for which the inequality (4.2) holds. Due to convexity of the function $\frac{-1}{\alpha}\log(.)$ for $\alpha > 0$, we can write

$$\begin{aligned} E_s H_{1+\alpha}(h_s(X)) &= E_s \frac{-1}{\alpha} \log \sum_{h_s(x)} Pr[h_s(x)]^{1+\alpha} \\ &\geq \frac{-1}{\alpha} \log E_s \sum_{h_s(x)} Pr[h_s(x)]^{1+\alpha}. \end{aligned} \tag{6.27}$$

Where x is a realization of random variable X. We can rewrite the right hand side as

$$\sum_{h_s(x)} Pr[h_s(x)]^{\alpha+1} = \sum_{\zeta} Pr[h_s(x) = \zeta] Pr[h_s(x') = h_s(x) = \zeta]^{\alpha}$$

where ζ is a realization of random variable Q. If condition 2 of universality holds for the ensemble of functions h_s, it implies that preimages of different ζ's (i.e. $h_s^{-1}(\zeta)$) are distinct. Therefore, taking expectation over random variable Q is equivalent to averaging over X. For the second term in this equation the probability of occurrence of a particular ζ is equivalent to finding probability of its preimage, so we have

$$\sum_{h_s(x)} Pr[h_s(x)]^{\alpha+1} = \sum_{x} Pr(X=x)Pr[h_s^{-1}(\zeta)]^{\alpha}$$
$$= \sum_{x} P_X(x)\left[\sum_{x':h_s(x')=h_s(x)} P_X(x')\right]^{\alpha}.$$

According to this equation we can state that

$$E_s \sum_{h_s(x)} Pr[h_s(x)]^{\alpha+1} = \sum_{x} P_X(x) E_s \left[\sum_{x':h_s(x')=h_s(x)} P_X(x')\right]^{\alpha}$$
$$\leq \sum_{x} P_X(x)\left[E_s \sum_{x':h_s(x')=h_s(x)} P_X(x')\right]^{\alpha}. \qquad (6.28)$$

The last inequality is due to concavity of $f(x)=x^{\alpha}$ for $0<\alpha\leq 1$.

$$E_s \sum_{x':h_s(x')=h_s(x)} P_X(x') \overset{1}{=} \sum_{S} Pr(S=S')\left[Pr(x'=x)+Pr(h_s(x')=h_s(x)|x'\neq x)\right]$$
$$\overset{2}{\geq} \sum_{S} P_s(S)[P_X(x)+\frac{1}{|\mathcal{K}|}] \overset{3}{=} P_X(x)+\frac{1}{|\mathcal{K}|}. \qquad (6.29)$$

Inequality 2 is resulted from the universality property of the family of hash functions h_s that maps Ω to the set of size $|\mathcal{K}|$. The equality 3 holds since the random variable S is uniform randomly distributed. We know that for $0<\alpha\leq 1$ we have $(x+y)^{\alpha}\leq x^{\alpha}+y^{\alpha}$. Therefore

$$\left[P_X(x)+\frac{1}{\mathcal{K}}\right]^{\alpha} \leq P_X(x)^{\alpha}+\frac{1}{\mathcal{K}^{\alpha}}. \qquad (6.30)$$

substituting Eq.'s (6.28)-(6.30) into Eq. (6.27) results in

$$E_sH_{1+\alpha}(h_s(X)) \geq \frac{-1}{\alpha}\log\sum_{x} P_X(x)\left[P_X(x)^{\alpha}+\frac{1}{|\mathcal{K}|^{\alpha}}\right]$$
$$= \frac{-1}{\alpha}\log\left[\sum_{x} P_X(x)^{1+\alpha}+\frac{1}{|\mathcal{K}|^{\alpha}}\right]$$
$$= \frac{-1}{\alpha}\log\left[e^{-\alpha H_{1+\alpha}(X)}+\frac{1}{|\mathcal{K}|^{\alpha}}\right]$$
$$= \frac{1}{\alpha}\log|\mathcal{K}|^{\alpha}-\frac{1}{\alpha}\log\left[1+e^{\alpha(\log|\mathcal{K}|-H_{1+\alpha}(X))}\right].$$

Finally, using the inequality $\log(1+x) \leq x$ and the facts that $h_s(X) = Q$ and $H_{1+\alpha}(X) \geq \delta$ proves the statement for $E_s H_{1+\alpha}(Q)$ and hence for Eq. (4.2). □

6.9 Proof of Lemma 6

Proof. Based on the definition of the Shannon and Rényi entropy we shall write

$$\begin{aligned} H_{1+\alpha}(X|Y) + H(Y) &= \frac{-1}{\alpha} \sum_y P_Y(y) \log \sum_x P_{X|Y}(x|y)^{1+\alpha} - \sum_y P_Y(y) \log P_Y(y) \\ &= \frac{-1}{\alpha} \sum_y P_Y(y) \left[\log \sum_x P_{X|Y}(x|y)^{1+\alpha} + \alpha \log P_Y(y) \right] \\ &= \sum_y P_Y(y) \frac{-1}{\alpha} \log \left[P_Y(y)^{\alpha} \sum_x P_{X|Y}(x|y)^{1+\alpha} \right]. \end{aligned}$$

For $\alpha > 0$, $\frac{-1}{\alpha}\log(.)$ is a convex function, so by using Jensen's inequality it can be concluded that

$$\begin{aligned} H_{1+\alpha}(X|Y) + H(Y) &\geq \frac{-1}{\alpha} \log \left(\sum_y P_Y(y)^{\alpha+1} \sum_x P_{X|Y}^{\alpha+1}(x|y) \right) \\ &= \frac{-1}{\alpha} \log \sum_{x,y} P_{X,Y}(x,y)^{\alpha+1} = H_{1+\alpha}(X,Y). \end{aligned}$$

□

Bibliography

[1] A. Wyner, “The wire-tap channel,” *Bell Syst. Tech. J.*, vol. 54 (8), pp. 1355–1387, Oct. 1975.

[2] U. M. Maurer, “Secret key agreement by public discussion from common information,” *IEEE Transactions on Information Theory*, vol. 39, no. 3, pp. 733 –742, may 1993.

[3] S. T. M. Bellare and A. Vardy, “A cryptographic treatment of the wiretap channel,” April 2010, preprint.

[4] M. Bellare, S. Tessaro, and A. Vardy, “Semantic security for the wiretap channel,” in *CRYPTO*, 2012, pp. 294–311.

[5] R. Renner and S. Wolf, “Simple and tight bounds for information reconciliation and privacy amplification,” in *Advances in Cryptology - ASIACRYPT 2005, 11th International Conference on the Theory and Application of Cryptology and Information Security*, Chennai, India, 2005, pp. 199–216.

[6] B. Barak, R. Shaltiel, and E. Tromer, “True random number generators secure in a changing environment,” *CHES*, pp. 166–180, 2003.

[7] R. Canetti, “Universally composable security: a new paradigm for cryptographic protocols,” in *42nd IEEE Symposium on Foundations of Computer Science, 2001,*, oct. 2001, pp. 136 – 145.

[8] A. Russell and H. Wang, “How to fool an unbounded adversary with a short key,” *IEEE Transactions on Information Theory*, vol. 52, no. 3, pp. 1130–1140, 2006.

[9] I. Csiszár and J. Körner, “Broadcast channels with confidential messages,” *IEEE Trans. Inform. Theory*, vol. 24(3), pp. 339–348, May 1978.

[10] H. Krawczyk, “Cryptographic extraction and key derivation: The hkdf scheme,” *CRYPTO*, pp. 631–648, 2010.

[11] I. Csiszár and P. Narayan, “Common randomness and secret key generation with a helper,” *IEEE Transactions on Information Theory*, vol. 46, no. 2, pp. 344–366, Mar.

[12] S. Xiao, W. Gong, and D. Towsley, “Secure wireless communication with dynamic secrets,” *In Proc. of IEEE INFOCOM 2010 - INFOCOM 2010*, pp. 1–9, Mar. 2010.

[13] U. Maurer and S. Wolf, “Secret-key agreement over unauthenticated public channels I definitions and a completeness result,” *IEEE Transactions on Information Theory*, vol. 49, no. 4, pp. 822 – 831, april 2003.

[14] H. Koorapaty, A. Hassan, and S. Chennakeshu, “Secure information transmission for mobile radio,” in *1998 Proceedings of, 1998 IEEE International Symposium on Information Theory*, Aug 1998, p. 381.

[15] Y. S. Khiabani, S. Wei, J. Yuan, and J. Wang, "Enhancement of secrecy of block ciphered systems by deliberate noise," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 5, pp. 1604–1613, Oct.

[16] Y. S. Khiabani and S. Wei, "ARQ based symmetric-key generation over correlated erasure channels," *Accepted and to be published in IEEE Transactions on Information Forensics and Security*, Apr 2013.

[17] ——, "Two-layer secrecy system with exponential security against unbounded adversary," *Submitted to IEEE Transactions on Information Forensics and Security*, March 2013.

[18] T. M. Cover and J. A. Thomas, *Elements of information theory*, 2nd ed. Wiley, 2006.

[19] A. Rényi, "On measures of entropy and information," *Proceedings of the 4th Berkeley Symposium on Mathematics, Statistics and Probability*, vol. 1, pp. 547–561, 1960.

[20] J. Aczél and Z. Daróczy, *On Measures of Information and Their Characterizations*. New York: Academic Press [Harcourt Brace Jovanovich Publishers], 1975, mathematics in Science and Engineering, Vol. 115.

[21] I. Csiszár and J. Körner, *Information theory: Coding theorem for Discrete Memoryless systems*. Academic Press, New York, 1981.

[22] C. E. Shannon, "Communication theory of secrecy systems," *Bell System Technical Journal*, vol. 28, pp. 656–715, Oct. 1949.

[23] S. K. Leung-Yan-Cheong and M. E. Hellman, "The Gaussian wire-tap channel," *IEEE Transactions on Information Theory*, vol. 24, no. 4, pp. 451 – 456, Jul 1978.

[24] M. v. Dijk, "On a special class of broadcast channels with confidential messages," *IEEE Trans. Inform. Theory*, vol. 43, pp. 712–714, Mar 1997.

[25] L. H. Ozarow and A. D. Wyner, "Wire-tap channel II," *Bell System Technical Journal*, vol. 63, no. 10, pp. 2135–2157, Dec 1984.

[26] G. Cohen and G. Zémor, "The wire-tap channel applied to biometrics," *In Proc. Int. Symp. Information Theory and its Applications, Italy*, Oct. 2004.

[27] ——, "Syndrome-coding for the wiretap channel revisited," *in Proc. IEEE Information Theory Workshop, Chengdu, China*, pp. 33–36, Oct. 2006.

[28] A. Thangaraj, S. Dihidar, A. Calderbank, S. McLaughlin, and J. Merolla, "Applications of LDPC codes to the wiretap channel," *IEEE Transactions on Information Theory*, vol. 53, no. 8, pp. 2933 –2945, Aug 2007.

[29] A. T. Suresh, A. Subramanian, A. Thangaraj, M. Bloch, and S. W. McLaughlin, "Strong secrecy for erasure wiretap channels," in *2010 IEEE Information Theory Workshop (ITW)*, Sep 2010, pp. 1 –5.

[30] H. Mahdavifar and A. Vardy, “Achieving the secrecy capacity of wiretap channels using polar codes,” *IEEE Transactions on Information Theory*, vol. 57, no. 10, pp. 6428 –6443, Oct. 2011.

[31] B. Schneier, *Applied cryptography (2nd ed.): protocols, algorithms, and source code in C*. New York, NY, USA: John Wiley & Sons, Inc., 1995.

[32] H. Delfs and H. Knebl, *Introduction to Cryptography, Principles and Applications, Second Edition*. Springer, 2007.

[33] A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone, *Handbook of Applied Cryptography*. CRC Press, 1996.

[34] V. Tilborg, C. A. Henk, Jajodia, and Sushil, *Encyclopedia of Cryptography and Security*. Springer, 2011.

[35] H. Heys and S. Tavares, “Avalanche characteristics of substitution- permutation encryption networks,” *IEEE Trans. Comput.*, vol. 44, no. 9, pp. 1131–1139, Sep 1995.

[36] P. Hamalainen, M. Hannikainen, T. Hamalainen, and J. Saarinen, “Configurable hardware implementation of triple-DES encryption algorithm for wireless local area network,” in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 2, 2001, pp. 1221 –1224.

[37] W. Zibideh and M. Matalgah, “Modified-DES encryption algorithm with improved BER performance in wireless communication,” in *Radio and Wireless Symposium (RWS), 2011 IEEE*, jan. 2011, pp. 219 –222.

[38] J. Daemen and V. Rijmen, “Aes proposal: Rijndael,” *AES Submission, version 2*, Sep. 1999.

[39] ——, “The block cipher rijndael,” *LNCS 1820, Smart-Card. Research and Applications*, pp. 288–296, Sep. 1999.

[40] M. Bellare and P. Rogaway, *Introduction to Modern Cryptography*. Course notes for UCSD cryptography course, 2005.

[41] Y. W. Law, J. Doumen, and P. Hartel, “Survey and benchmark of block ciphers for wireless sensor networks,” *ACM Trans. Sen. Netw.*, vol. 2, pp. 65–93, Feb 2006.

[42] Y. Xiao, H. Chen, X. Du, and M. Guizani, “Stream-based cipher feedback mode in wireless error channel,” *IEEE Trans. Wireless Comm.*, vol. 8, pp. 622–626, 2009.

[43] M. Matsui, “Linear cryptanalysis method for DES cipher,” *Lecture Notes in Computer Science*, vol. 765, pp. 385–397, 1994.

[44] A. Selçuk and A. Bicak, “On probability of success in linear and differential cryptanalysis,” *SCN 2002*, pp. 174–185, 2003.

[45] W. Diffie and M. E. Hellman, “New directions in cryptography,” *IEEE Transactions on Information Theory*, vol. 22, no. 6, pp. 644–654, Nov. 1976.

[46] R. L. Rivest, A. Shamir, and L. Adleman, “A method for obtaining digital signatures and public-key cryptosystems,” *Communications of the ACM*, vol. 21, no. 2, pp. 120–126, 1978.

[47] R. Ahlswede and I. Csiszár, “Common randomness in information theory and cryptography. i. secret sharing,” *Information Theory, IEEE Transactions on*, vol. 39, no. 4, pp. 1121 –1132, jul 1993.

[48] I. Csiszár and P. Narayan, “Common randomness and secret key generation with a helper,” *Information Theory, IEEE Transactions on*, vol. 46, no. 2, pp. 344 –366, mar 2000.

[49] ——, “Secrecy capacities for multiple terminals,” *Information Theory, IEEE Transactions on*, vol. 50, no. 12, pp. 3047 – 3061, dec. 2004.

[50] C. Ye, A. Reznik, G. Sternberg, and Y. Shah, “On the secrecy capabilities of itu channels,” in *Vehicular Technology Conference, 2007. VTC-2007 Fall. 2007 IEEE 66th*, Oct. 2007, pp. 2030 –2034.

[51] A. Sayeed and A. Perrig, “Secure wireless communications: Secret keys through multipath,” in *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*, April 2008, pp. 3013 –3016.

[52] M. Bloch, J. Barros, M. Rodrigues, and S. McLaughlin, “Wireless information-theoretic security,” *IEEE Transactions on Information Theory*, vol. 54, no. 6, pp. 2515 –2534, June 2008.

[53] M. A. Latif, A. Sultan, and H. E. Gamal, “ARQ-based secret key sharing,” *IEEE International Conference on Communications 2009 - ICC ’09*, pp. 1–6, June 2009.

[54] Y. Abdallah, M. A. Latif, M. Youssef, A. Sultan, and H. E. Gamal, “Keys through ARQ: Theory and practice,” *IEEE Trans. Inf. Forens. Security*, vol. 6, no. 3, pp. 737 –751, Sept. 2011.

[55] C. H. Bennett, G. Brassard, C. Crpeau, and U. M. Maurer, “Generalized privacy amplification,” *IEEE Trans. Inform. Theory, vol. 41*, pp. 1915–1923, Nov. 1995.

[56] U. M. Maurer and S. Wolf, “Information-theoretic key agreement: From weak to strong secrecy for free,” in *Advances in Cryptology, EUROCRYPT 2000 (Lecture Notes in Computer Science)*, vol. 1807. Berlin, Germany: Springer-Verlag, 2000, pp. 351–368.

[57] U. Maurer and S. Wolf, “Secret-key agreement over unauthenticated public channels II privacy amplification,” *IEEE Transactions on Information Theory*, vol. 49, no. 4, pp. 839 – 851, April 2003.

[58] M. Hayashi, “Exponential decreasing rate of leaked information in universal random privacy amplification,” *IEEE Trans. Inf. Theory*, vol. 57, no. 6, pp. 3989–4001, June 2011.

[59] Y. Dodis, R. Ostrovsky, L. Reyzin, and A. Smith, “Fuzzy extractors: How to generate strong keys from biometrics and other noisy data,” *SIAM J. Comput.*, vol. 38, no. 1, pp. 97–139, 2008.

[60] Y. Ishai, E. Kushilevitz, R. Ostrovsky, and A. Sahai, “Extracting correlations,” in *50th Annual IEEE Symposium on Foundations of Computer Science, 2009. FOCS ’09.*, Oct. 2009, pp. 261 –270.

[61] U. M. Maurer, “Information-theoretically secure secret-key agreement by not authenticated public discussion,” in *EUROCRYPT*, 1997, pp. 209–225.

[62] A. Orlitsky, N. Santhanam, and J. Zhang, “Universal compression of memoryless sources over unknown alphabets,” *IEEE Transactions on Information Theory*, vol. 50, no. 7, pp. 1469–1481, July.

[63] M. Hayashi, “Tight exponential evaluation for universal composablity with privacy amplification and its applications,” *CoRR*, vol. abs/1010.1358, 2010.

[64] M. Matsui, “The first experimental cryptanalysis of the data encryption standard,” *Lecture Notes in Computer Science*, vol. 835, pp. 1–11, 1994.

[65] E. Biham and A. Shamir, “Differential cryptanalysis of the full 16-round des,” in *CRYPTO*, 1992, pp. 487–496.

[66] S.-C. Lu, “Random ciphering bounds on a class of secrecy systems and discrete message sources,” *IEEE Transactions on Information Theory*, vol. 25, no. 4, pp. 405–414, Jul.

[67] ——, “Addition to ‘the existence of good cryptosystems for key rates greater than the message redundancy’,” *IEEE Transactions on Information Theory*, vol. 26, no. 1, p. 129, 1980.

[68] R. Ahlswede and G. Dueck, “Bad codes are good ciphers,” *Probl. Contr. Inform. Theory*, vol. 11, no. 5, pp. 337–351, July.

[69] W. F. Ehrsam, S. M. Matyas, C. H. Meyer, and W. L. Tuchman, “A cryptographic key management scheme for implementing the data encryption standard,” *IBM Systems Journal*, vol. 17, no. 2, pp. 106–125, 1978.

[70] R. Yin, S. Wei, J. Yuan, X. Shan, and X. Wang, “Tradeoff between reliability and security in block ciphering systems with physical channel errors,” *Proc. IEEE Military Commun. Conf. (MILCOM)*, pp. 2156 –2161, 2010.

[71] S. Goel and R. Negi, “Guaranteeing secrecy using artificial noise,” *Wireless Communications, IEEE Transactions on*, vol. 7, no. 6, pp. 2180 –2189, june 2008.

[72] J. Vilela, M. Bloch, J. Barros, and S. McLaughlin, “Wireless secrecy regions with friendly jamming,” *Information Forensics and Security, IEEE Transactions on*, vol. 6, no. 2, pp. 256 –266, june 2011.

[73] M. J. Mihaljevic and H. Imai, “An approach for stream ciphers design based on joint computing over random and secret data,” *Computing*, vol. 85, pp. 153–168, 2009.

[74] M. Willett, “Deliberate noise in a modern cryptographic system (corresp.),” *IEEE Transactions on Information Theory, vol.26, no.1*, pp. 102– 104, 1980.

[75] M. Mihaljevicï£¡ and F. Oggier, “A wire-tap approach to enhance security in communication systems using the encoding-encryption paradigm,” *IEEE 17th International Conference on Telecommunications (ICT)*, pp. 83–88, April 2010.

[76] P. C. Kocher, J. Jaffe, and B. Jun, “Differential power analysis,” in *Proceedings of the 19th Annual International Cryptology Conference on Advances in Cryptology*, ser. CRYPTO ’99. London, UK: Springer-Verlag, 1999, pp. 388–397.

[77] T. Roche, V. LomnÃl’, and K. Khalfallah, “Combined fault and side-channel attack on protected implementations of AES,” *CARDIS*, pp. 65–83, 2011.

[78] *RFC 791 Internet Protocol - DARPA Inernet Programm, Protocol Specification*, Internet Engineering Task Force, September 1981. [Online]. Available: http://tools.ietf.org/html/rfc791

[79] S. Ross, “Introduction to probability models,” *University of Southern California, Academic Press*, Tenth Edition, ISBN: 978-0-12-375686-2, 2010.

[80] S. Yu, Z. Liu, M. Squillante, C. Xia, and L. Zhang, “A hidden semi-$m$arkov model for web workload self-similarity,” *21st IEEE International Performance, Computing, and Communications Conference*, pp. 65–72, 2002.

[81] A. J. Goldsmith and P. P. Varaiya, “Capacity, mutual information, and coding for finite-state markov channels,” *IEEE Trans. Inform. Theory*, vol. 43, pp. 868–886, May 1996.

[82] T. Holliday, A. Goldsmith, and P. Glynn, “Capacity of finite state markov channels with general inputs,” *In Proceedings of the IEEE International Symposium on Information Theory*, p. 289, 2003.

[83] H. S. Wang and N. Moayeri, “Finite-state markov channel: A useful model for radio communication channel,” *Proc. IEEE Veh. Tech. Conf. (VTC)*, vol. 44, pp. 163–171, Feb 1995.

[84] M. Mushkin and I. Bar-David, “Capacity and coding for the Gilbert Elliot channel,” *IEEE Trans. Inform. Theory*, vol. 35, pp. 1277–1290, 1989.

[85] Y. Sankarasubramaniam, A. Thangaraj, and K. Viswanathan, “Finite-state wiretap channels: Secrecy under memory constraints,” *Information Theory Workshop, 2009. ITW 2009. IEEE*, pp. 115 –119, Oct. 2009.

[86] M. Zorzi, R. R. Rao, and L. B. Milstein, “ARQ error control for fading mobile radio channels,” *IEEE Trans. Veh. Technol.*, vol. 46, no. 2, pp. 445–455, 1997.

[87] S. R. Kim and C. K. Un, “Throughput analysis for two ARQ schemes using combined transition matrix,” *IEEE Trans. Commun.*, vol. 40, no. 11, pp. 1679 –1683, nov. 1992.

[88] W. C. Y. Lee, “Effects on correlation between two mobile radio base-station antennas,” *IEEE Trans. Veh. Technol.*, vol. 22, no. 4, pp. 130–140, Nov 1973.

[89] H. Jeon, N. Kim, J. Choi, H. Lee, and J. Ha, “Bounds on secrecy capacity over correlated ergodic fading channels at high SNR,” *IEEE Trans. Inform. Theory.*, vol. 57, no. 4, pp. 1975–1983, Apr. 2011.

[90] W. K. Harrison, J. Almeida, S. McLaughlin, and J. Barros, “Physical-layer security over correlated erasure channels,” *IEEE International Conference in Communications 2012 - ICC’12*, Ottawa, Canada, June 2012.

[91] T. A. Schonhoff and A. A. Giordano, *Detection and Estimation Theory and its Applications*, 1st ed. New Jersey: Pearson Prentice-Hall, 2006.

[92] G. Grimmett and D. Stirzaker, *Probability and Random Processes*, 3rd ed. Oxford, UK: Oxford University Press, 2001.

[93] L. Carter and M. N. Wegman, “Universal classes of hash functions,” *J. Comput. Syst. Sci.*, vol. 18, no. 2, pp. 143–154, 1979.

[94] R. G. Gallager, *Information Theory and Reliable Communication*. John Wiley & Sons, 1968.

[95] M. Cheraghchi, F. Didier, and A. Shokrollahi, “Invertible extractors and wiretap protocols,” in *IEEE International Symposium on Information Theory, 2009. ISIT 2009*, July 2009, pp. 1934 –1938.

[96] Y. Liang, H. Poor, and S. Shamai, “Information theoretic security,” *Foundations and Trends in Communications and Information Theory*, vol. 5, no. 4, pp. 355 – 580, 2008.

[97] N. Nisan and A. Ta-Shma, “Extracting randomness: A survey and new constructions,” *J. Comput. Syst. Sci.*, vol. 58, no. 1, pp. 148–173, 1999.

[98] U. Maurer and S. Wolf, “Secret-key agreement over unauthenticated public channels - part III: Privacy amplification,” *IEEE Trans. Inf. Theory*, vol. 49, no. 4, pp. 839–851, Apr 2003.

[99] M. Ben-Bassat and J. Raviv, “Renyi’s entropy and the probability of error,” *IEEE Transactions on Information Theory*, vol. 24, no. 3, pp. 324 – 331, May 1978.

[100] D. R. Stinson, “Universal hashing and authentication codes,” *Des. Codes Cryptography*, vol. 4, no. 4, pp. 369 –380, 1994.

[101] R. H. Deng, “Hybrid ARQ schemes for point-to-multipoint communication over nonstationary broadcast channels,” *IEEE Transactions on Communications*, vol. 41, no. 9, pp. 1379–1387, sep 1993.

[102] A. Neri, D. Blasi, and P. Campisi, “Secure HARQ communication protocols based on turbo codes,” *4th International Symposium on Communications, Control and Signal Processing (ISCCSP)*, pp. 1–4, march 2010.

[103] X. Tang, R. Liu, P. Spasojevic, and H. V. Poor, “On the throughput of secure hybrid-ARQ protocols for Gaussian block-fading channels,” *IEEE Trans. on Inform. Theory*, vol. 55, no. 4, pp. 1575–1591, 2009.

[104] Y. S. Khiabani, S. Wei, J. Yuan, and J. Wang, “Enhancement of secrecy of block ciphered systems by deliberate noise,” *Technical Report*, Apr 2012, Available from http://arxiv.org/abs/1204.0153.

Vita

Yahya Sowti Khiabani was born in July, 1981, in Tabriz, Iran. He received his B.S. and M.S. degrees in Electrical and Computer Engineering from the University of Tabriz, Iran, in 2003 and 2007. In 2009 he was admitted as a Ph.D. student in Louisiana state university, ECE department. He was granted Economic Development Assistantship (EDA) to work as a research assistant on security in wireless networks under advisory of Dr. Shuangqing Wei during 2009-2012. He is currently working toward the Ph.D. degree in Electrical Engineering with the School of Electrical Engineering and Computer Sciences at the Louisiana State University (LSU), Baton Rouge and is expected to graduate by August 2013.

As a Ph.D. student, Yahya's research has been focused on both information theoretic and cryptographic based security and anti-eavesdropping algorithms. During his Ph.D. work, he has been able to have one published, one accepted and one submitted journal papers to IEEE transactions on information forensics and security. He also published six conference papers on Globecom 2011 and 2012, Milcom 2011 and 2012 as well as CISS 2013 conferences. He has one submitted conference paper to CNS 2013. His current research interests include unconditional security, network tomography and security in cloud computing.