2001

Unconditionally secure schemes for distributed authentication systems

Huaxiong Wang
University of Wollongong

Recommended Citation
Unconditionally Secure Schemes for Distributed Authentication Systems

A thesis submitted in fulfillment of the requirements for the award of the degree

Doctor of Philosophy

from

UNIVERSITY OF WOLLONGONG

by

Huaxiong Wang

Computer Science Department
June 2001
Dedicated to

Qun and Anna
Declaration

This is to certify that the work reported in this thesis was done by the author, unless specified otherwise, and that no part of it has been submitted in a thesis to any other university or similar institution.

________________________________________
Huaxiong Wang
June 26, 2001
Abstract

One of the main goals of a cryptographic system is to provide authentication, which simply means providing assurance about the content and origin of communicated message.

Historically, cryptography began with secret writing and this remained the main area of development until very recently. With the rapid progress in data communication, the need for providing message integrity and authenticity has escalated to the extent that currently authentication is seen as the more urgent goal of cryptographic systems.

Traditionally, it was assumed that a secrecy system provides authentication by the virtue of the secret key being only known by the intended communicants; this would prevent an enemy from constructing a fraudulent message. It is a fairly recent realization that secrecy and authenticity are quite distinct goals. In classical or private cryptography these two concepts were closely intertwined. While secrecy depended on the message being unintelligible to any receiver who did not know the secret key, authenticity depended on the inability of anyone without knowledge of the secret key to produce a ciphertext that would decipher to an intelligible message. Simmons argued that the two goals of cryptography are independent. He shows that a system that provides perfect secrecy might not provide any protection against authentication threats. Similarly, a system can provide perfect authentication without concealing the message.

This thesis will deal with authentication theory.

Chapter 1 reviews the basic concepts of authentication theory, some fundamental results and constructions of authentication schemes.

In Chapter 2, we study the application of algebraic curves over finite fields to the constructions of universal hash functions and unconditionally secure authentication codes.

The theory of algebraic curves over finite fields has many applications in coding
theory and it has been observed by several authors that coding theory, universal hash families and authentication codes are closely related. This has led researchers to naturally consider the algebraic-geometric codes based on some specified algebraic curves to construct universal hash families and authentication codes. We pursue this direction of research. However, unlike the earlier approach, which composes a family of efficient geometric codes to obtain a strongly universal hash family, known as "composition method", we instead provide a direct construction of strongly universal hash families from the algebraic curves over finite fields without using the composition method. We show that our constructions result in new classes of strongly universal hash families and authentication codes which have the best performance known so far.

In Chapter 3, we study multireceiver authentication codes (MRA-codes). MRA-codes were introduced by Desmedt, Frankel and Yung in 1992 as an extension of Simmons' model of unconditionally secure authentication. In an MRA-code, a sender wants to authenticate a message for a group of receivers such that each receiver can individually verify authenticity of the received message. The receivers are not trusted and may try to construct fraudulent messages on behalf of the sender. If the fraudulent message is acceptable by even one receiver the attackers have succeeded. This is a useful extension of traditional authentication codes and has numerous applications.

In this chapter we start by giving a formal definition of an MRA-code and use it to derive information theoretic bounds on the probability of success in impersonation and substitution attack against a single receiver for a general MRA-code. These bounds are used to obtain lower bounds on the the number of keys of the sender and receivers, and also lower bound on the length of the transmitted codeword in terms of deception probability of the system. This is followed by a review of the known constructions of MRA-codes, pointing out their shortcomings and giving constructions that alleviate these shortcomings.

In Chapter 4, we introduce and investigate multireceiver authentication codes with dynamic senders (DMRA-code). MRA-codes can be seen as the first attempt in providing authentication in group communication. However in this model the sender is fixed. We remove this limitation by introducing the new model of DMRA-codes to allow the sender to be dynamic. The extended model of DMRA-codes captures the essential aspect of authentication problem in group communication with many applications such as dynamic conference key distributions.

In this chapter we introduce the model and give the formal definition of DMRA-codes. We prove the lower bounds on size of the key and codeword, and give an
optimal construction that meets the bounds. We also describe efficient constructions from key distribution patterns and perfect hash families. We consider systems with multiple senders and present two constructions. As one application, we propose a secure dynamic conference authenticated system based on our new model. Finally, computationally secure group authentication systems are also discussed.

Chapter 5 deals with shared generation of authenticator systems (SGA-systems). SGA-systems are a generalisation of Simmons' traditional model of authentication where authenticating a message (also called a source state) requires collaboration of a group of senders. In a \((t, n)\) threshold SGA-system any group of \(t\) senders is an authorised group and no coalition of \(t - 1\) (or less) senders can produce a valid authenticator for a source state (impersonation attack) or, after seeing a valid codeword can produce a valid authenticator for a different source state (substitution attack). This means that in a \((t, n)\) threshold SGA-systems receiving an authenticated codeword implies authorisation of at least \(t\) senders. SGA-systems are an important cryptographic primitive in distributed systems where a number of parties located at distant geographic location are to collaboratively authorise an action, sign a document or share responsibility.

Threshold SGA-systems are part of a general approach to cryptography, known as threshold cryptography which has received considerable attention in recent years. While computationally secure threshold authentication, e.g. threshold RSA and DSS signature have been studied extensively, the information-theoretic secure model, i.e. threshold SGA-systems has received much less attention. Unconditionally secure threshold SGA-systems were first introduced by Desmedt, Frankel and Yung in 1992. The basic idea behind their schemes is to combine a traditional A-code and a secret sharing scheme to construct a threshold SGA-system. We note that a naive application of a secret sharing scheme to share the secret authentication key among the senders is not acceptable simply because it requires full trust on the combiner and in fact allows him to authenticate any message of his choice and without requiring any collaboration from senders, after receiving partial authenticators from an authorised group. The key assumption in an SGA-system is that the combiner is not trusted and the combining operation does not require any secret information.

In this chapter we formally define the model of threshold SGA-systems and their important parameters, and then we derive information-theoretic and combinatorial lower bounds on \((t, n)\) SGA-systems. We present two efficient \((t, n)\) SGA-systems and show how to build new threshold SGA-systems for large groups from old ones (small groups) by using perfect hash families. We also discuss the robustness in SGA-systems.
and propose two constructions.

Chapter 6 studies multiplicative secret sharing schemes. Secret sharing schemes, introduced independently by Shamir and Blakely in 1979, are one of the main building blocks of secure distributed computation. A secret sharing scheme gives a method of breaking a secret into shares that are distributed among members of a group $\mathcal{P}$, called participants, such that only authorised subgroups of $\mathcal{P}$ can recover the secret. To generate the shares of a secret, a dealer uses a random input to select a distribution rule that determines the share given to each participant. In a perfect secret sharing scheme unauthorised subgroups do not learn anything about the secret. In a $(t,n)$ threshold scheme, any group of at least $t$ out of $n$ users can recover the secret.

Multiplicative secret sharing schemes play an important role in the constructions of SGA systems and MRA-codes. In this chapter, we introduce a new approach to constructing new threshold schemes from old ones. The basic idea is to start with the set of shares generated for an instance of the original scheme and then form the shares of the new scheme as subsets of the shares of the old one. To form the subsets, we introduce a new combinatorial structure, called a strong cover free family. Cover-free families are extensively studied by various authors and have found numerous applications in cryptography. Strong cover-free families are a special case of cover-free families that can be constructed in a number of ways and in particular using universal classes of hash functions and error correcting codes. Distributing shares to the new participants according to the blocks of a strong cover-free family results in a threshold scheme which inherits security and structural properties of the original scheme, that is perfectness, homomorphic and multiplicative property, with the same randomness of the original scheme.

We also give bounds on parameters and numerous constructions of the newly introduced concept, SCFF.
Cryptography is the science and study of secret writing. A cipher is a secret method of writing, whereby a plaintext (or cleartext) is transformed into a ciphertext. The process of transforming the plaintext into the ciphertext is called encryption; the reverse process of transforming the ciphertext into the plaintext is called decryption. Both encryption and decryption are controlled by a cryptographic key or a number of cryptographic keys.

The traditional goal of a communications engineer is to ensure that the message delivered to the destination is the same as that sent by the sender. The enemy is noise. Cryptography by contrast has two goals: secrecy and/or authenticity. A cryptographer may seek to ensure that the message is intelligible only to the intended receiver – the enemy is the “eavesdropper” who overhears the transmitted signals. The cryptographer may seek instead (or also) to ensure that the identity of the sender and the integrity of the message can be unmistakably verified by the receiver – the enemy is the “spoof” who can originate, or tamper with, transmitted signals.

It is a fairly recent realization that secrecy and authenticity are quite distinct goals. In classical or private cryptography these two concepts were closely intertwined. While secrecy depended on the message being unintelligible to any receiver who did not know the secret key, authenticity depended on the inability of anyone without knowledge of the secret key to produce a ciphertext that would decipher to an intelligible message.

It was only with the introduction of public-key cryptography by Diffie and Hellman [43] in 1976 that it became clear that secrecy and authenticity did not always go hand-in-hand.

In 1949, Shannon [98] provided a theoretical foundation for cryptography based on his fundamental work on information theory [97]. He measured the protection of a cipher by the uncertainty about the plaintext given the received ciphertext. If, no matter how much ciphertext is intercepted, nothing can be learned about the plaintext, the cipher achieves perfect secrecy.
We shall follow Shannon’s lead in distinguishing between the two types of cryptographic security: *theoretical security* and *practical security*. “Theoretical security” means that security is unconditional, that is, the cryptographic system provides against an enemy who has unlimited computational resources available to him. “Practical security” means, the system is computationally secure, it provides against an enemy with finite computational resources. A system is theoretically secure, or unconditionally secure, if it is impossible to break regardless of how much effort the enemy cryptanalyst expends. A system is practically secure, or computationally secure if its breaking requires a computational effort beyond the enemy’s means.

The theory of unconditionally secure authentication is still very young. The initial study appears to be Gilbert, MacWilliams and Sloane’s landmark paper “Codes which detect deception” [56] in 1974. In the beginning of 80’s Simmons placed this subject in a more general setting and developed a more systematic theory. Because of its young age, many researchers feel that the theory of unconditionally secure authentication is not so well developed and established.

A completely different solution to the same problem appeared in 1976 when Diffie and Hellman introduced the concept of a *digital signature*. Here the solution depends on the assumption that certain difficult problems cannot be solved efficiently [43]. For example, it is hard to factor large numbers.

In this thesis, we will be mainly concerned with the theoretical security of cryptographic systems. We will study unconditionally secure authentication schemes. The unconditionally secure authentication theory draws heavily from mathematics and computer science, it has been studied from many different areas, for example in design theory, finite geometry, finite field, coding theory and information theory. In this thesis, we will focus on authentication systems in distributed environments. We have been interested in both theoretical and practical aspects.

Here is a brief summary of the chapters:

- *Chapter 1. Introduction*, introduces the basic concepts of authentication codes, briefly reviews some fundamental results in authentication theory.

- *Chapter 2. Authentication Codes and Universal Hashing from Algebraic Curves*, studies the application of algebraic curves over finite fields to the constructions of universal hash functions and unconditionally secure authentication codes. We show that the construction from Garcia-Stichtenoth curves yields new classes of authentication code and universal hash families which are substantially better than those previously known.
• Chapter 3. Multireceiver Authentication Codes (MRA-codes), studies authentication in multireceiver environments. We extend the mathematical model of multireceiver authentication codes and derive the information-theoretic and combinatorial bounds for such systems. We also give some new optimal MRA-codes from coding theory.

• Chapter 4. Multireceiver Authentication Codes with Dynamic Senders (DMRA-codes), continues the study of MRA-codes. We introduce the model of MRA-codes with dynamic senders, called DMRA-codes, we derive the combinatorial bounds and present several methods for constructions of DMRA-codes. We also address the application of DMRA-codes in secure group communications.

• Chapter 5. Shared Generation of Authenticator systems (SGA-systems), deals with threshold authentication systems. We derive the information-theoretic and combinatorial bounds for \((t,n)\) threshold authentication codes and present new constructions. We also introduce the model of robust threshold authentication and give two efficient constructions.

• Chapter 6. Multiplicative Secret Sharing and Threshold Authentication Codes, explores the connection between threshold authentication codes and secret sharing schemes. We show how a threshold authentication code can be constructed by a multiplicative secret sharing scheme and a linear authentication. We then study the efficient construction for multiplicative secret sharing schemes.

Part of this thesis appear in the following papers.


Multiplicative secret sharing schemes and strong cover-free families, to be submitted.

During my PhD studies, I also wrote and published the following papers which I have not included in this thesis.


Acknowledgements

I am most grateful to my supervisor Rei Safavi-Naini. Her advice, guidance and support have been invaluable to me. I am also grateful to my co-supervisor Josef Pieprzyk for many advice and many interesting discussion. Working with Rei and Josef has been the most exciting part of my PhD studies.

Thanks to Jonathan Golan for his various advice and continue encouragement during this project.

Thanks to Cunsheng Ding and Chaoping Xing for their various helps during my 10 months stays at National University of Singapore.

Thanks to all my co-authors who cooperated with me during this research: Lynn Batten, Chris Charnes, Yvo Desmedt, Hossein Ghodosi, Kwok-Yan Lam, Keith Martin, Josef Pieprzyk, Rei Safavi-Naini, Igor Shparlinski, Guozheng Xiao and Chaoping Xing.

I am also grateful for the help and assistance that the staff, visitors and students in the Centre for Computer Security Research provided to me.
I am indebted to my wife Qun, and my daughter Anna.
## Contents

Abstract \hspace{1cm} v

Preface \hspace{1cm} ix

1 Introduction to Authentication Theory \hspace{1cm} 1

1.1 Authentication Model \hspace{1cm} 2

1.2 Bounds on the performance of the A-codes \hspace{1cm} 5

1.2.1 Information-Theoretic Bounds for A-codes \hspace{1cm} 5

1.2.2 Combinatorial Bounds of A-codes \hspace{1cm} 6

1.3 Other Issues \hspace{1cm} 10

1.3.1 Other types of attacks \hspace{1cm} 10

1.3.2 Authentication codes and error-correcting codes \hspace{1cm} 10

1.3.3 Authentication with arbiter \hspace{1cm} 11

1.3.4 Shared generation of authenticators \hspace{1cm} 12

1.3.5 Broadcast authentication codes \hspace{1cm} 12

1.3.6 Multiple authentication \hspace{1cm} 13

1.4 Computationally Secure Authentication Systems \hspace{1cm} 13

1.4.1 Unconditionally secure MACs \hspace{1cm} 14

1.4.2 Wegman and Carter construction \hspace{1cm} 15

1.4.3 Computational security \hspace{1cm} 17

1.4.4 Security analysis of computationally secure MACs \hspace{1cm} 18

1.4.5 Formal security analysis \hspace{1cm} 19

2 Authentication Codes and Universal Hashing \hspace{1cm} 22

2.1 Universal hashing \hspace{1cm} 23

2.2 Constructions \hspace{1cm} 25

2.3 Comparisons with previous constructions \hspace{1cm} 30

2.4 Conclusion \hspace{1cm} 33
3 Multireceiver Authentication Codes

3.1 The Model ....................................................... 35
3.2 Information-Theoretic Bounds ................................. 37
3.3 Combinatorial Bounds ............................................. 43
3.4 Comparison with Kurosawa et al Bounds ...................... 45
3.5 Constructions ..................................................... 47
3.5.1 DFY Polynomial Construction .......................... 47
3.5.2 A Construction based on $(n, m, w)$-Cover-Free Family 48
3.5.3 MRA-codes for Multiple Message Transmissions ....... 50
3.5.4 A Construction from Error-Correcting Codes ......... 54
3.5.5 Construction II ................................................. 57

4 Multireceiver Authentication Codes with Dynamic Senders 60

4.1 The Model ....................................................... 63
4.2 DMRA-codes with a Single Sender ............................ 66
4.2.1 Bounds .......................................................... 66
4.2.2 An optimal construction ..................................... 66
4.2.3 A General Construction ..................................... 70
4.3 DMRA-codes with Multiple Senders .......................... 73
4.3.1 A polynomial construction for $t$DMRA-codes .......... 73
4.3.2 A general construction from perfect hash families .... 75
4.4 A Secure Dynamic Conference System .................... 79
4.5 Computationally secure $t$DMRA-codes ..................... 82
4.6 Conclusions ..................................................... 83

5 Shared Generation of Authenticator Systems (SGA-systems) 85

5.1 Model and Bounds for SGA-systems .......................... 88
5.2 Efficient constructions for SGA-system ....................... 92
5.2.1 $(t, n)$ SGA-system based on modified den Boer A-code 93
5.2.2 $(t, n)$ SGA-systems from linear error-correcting codes 95
5.3 Recursive Constructions ....................................... 97
5.4 Robust SGA-systems .......................................... 100
5.4.1 Combiner Verifiable Scheme (CVS) .................... 102
5.4.2 Public Verification Scheme (PVS) ...................... 105
5.5 Multireceivers SGA-system .................................... 106
5.6 Conclusions ..................................................... 110
6 Multiplicative Secret Sharing

6.1 Secret Sharing Schemes ................................................................. 111
  6.1.1 Constructing new threshold schemes from old ones .............. 112
  6.1.2 Our results ........................................................................... 113

6.2 Preliminaries ................................................................................ 114
  6.2.1 Blackburn-Burmester-Desmedt-Wild's scheme ...................... 116

6.3 Our approach ............................................................................. 117
  6.3.1 An example .......................................................................... 118

6.4 Bounds ....................................................................................... 119

6.5 Constructions ............................................................................. 120
  6.5.1 Constructions from combinatorial designs ......................... 121
  6.5.2 Constructions from universal hashing families .................... 122
  6.5.3 Construction based on exponential sums ......................... 124

6.6 Evaluation .................................................................................. 125

6.7 Concluding Remarks .................................................................. 126

Bibliography .................................................................................. 128
List of Tables
List of Figures
Chapter 1

Introduction to Authentication Theory

One of the main goals of a cryptographic system is to provide authentication, which simply means providing assurance about the content and origin of communicated message.

Historically, cryptography began with secret writing and this remained the main area of development until very recently. With the rapid progress in data communication, the need for providing message integrity and authenticity has escalated to the extent that currently authentication is seen as the more urgent goal of cryptographic systems.

Traditionally, it was assumed that a secrecy system provides authentication by the virtue of the secret key being only known by the intended communicants; this would prevent an enemy from constructing a fraudulent message. Simmons [99] argued that the two goals of cryptography are independent. He shows that a system that provides perfect secrecy might not provide any protection against authentication threats. Similarly, a system can provide perfect authentication without concealing the message.

In this thesis, we use the term *communication system* to encompass message transmission as well as storage. The system consists of one or more *transmitters* (or *senders*) who want to send a message, one or more *receivers* who are the intended recipients of the message, and an *enemy* (or *opponent*) who attempts to construct a fraudulent message with the aim of getting it accepted by the receiver unwittingly. The communication is assumed to be over a public channel, and hence the communicated message can be seen by all the principals. An authentication threat is an attempt by an enemy in the system to modify a communicated message or inject a fraudulent message into the channel. In a secrecy system the attacker is passive, while in an authentication system the enemy is active and not only observes the communicated message and gathers information such as plaintext and ciphertext, but also actively interacts with the system to achieve its goal. This view of the system clearly explains Simmons' motivation for basing authentication systems on game theory.
The most important criteria that can be used to classify authentication systems are:

- the relation between authenticity and secrecy;
- the framework for the security analysis.

The first criterion divides authentication systems into those that provide authentication with and without secrecy. The second criterion divides systems into systems with unconditional security and systems with computational security. Unconditional security implies that the enemy has unlimited resources while in systems with computational security, the security relies on the required computation exceeding the enemy's computational power.

These two classifications are orthogonal and produce four subclass. Below we review the basic concepts of authentication theory, some known bounds and constructions in unconditional security. We then also briefly review computationally secure authentication systems, also called message authentication codes (or MACs).

1.1 Authentication Model

In Simmons' model of unconditionally secure authentication, there are three participants: a transmitter (sender), a receiver and an opponent. The transmission from the sender to receiver takes place over an insecure channel. The opponent has access to the channel and can insert a message into the channel, or observe a transmitted message and replace it with another message.

The information that the sender wants to send is called a source state, denoted by \( s \) and taken from the finite set \( S \) of possible source states. The source state is mapped into a (channel) message, denoted by \( m \) and taken from the set of possible messages.

Exactly how this mapping is performed is determined by the secret key, called "the encoding rule", which is denoted by \( e \) and taken from the set \( E \) of possible encoding rules. The encoding rule is secretly shared between the sender and receiver.

**Definition 1.1** An authentication code (A-code) is a 4-tuple \((S, E, M, f)\), where \( f \) is a mapping from \( S \times E \) to \( M \)

\[
f : S \times E \rightarrow M
\]

such that \( f(s, e) = m \) and \( f(s', e) = m \) implies \( s = s' \).
In the above definition, an important property is that \( f \) satisfies the condition \( f(s, e) = m \) and \( f(s', e) = m \) implies \( s = s' \). It follows for each \( e \in \mathcal{E} \), \( f(., e) \) induces an injective mapping from \( S \) to \( \mathcal{M} \). In other words, two different source states cannot be mapped into the same message for a given encoding rule. In general, the mapping \( f \) can be a probabilistic mapping, i.e., \( f(s, e) \) may take on one of several possible values determined by some probabilistic law. In authentication theory this is called splitting. In this thesis, we will only deal with non-splitting A-code.

Given an A-code \( (S, \mathcal{E}, \mathcal{M}, f) \), in order to authenticate a source state, the sender and receiver follow the following protocol. First, they agree on a key \( e \in \mathcal{E} \) which is randomly selected, which can be through a trusted way. At a later time, if the sender wants to communicate a source state \( s \in S \) to the receiver over an insure channel, he computes \( m = f(s, e) \) and sends \( m \) to the receiver. When the receiver receives the message \( m \), he checks whether a source \( s \) such that \( f(s, e) = m \) exists. If such an \( s \) exists, the message \( m \) is accepted as authentic (\( m \) is called valid). Otherwise, \( m \) is not authentic and thus rejected.

We will study two different types of attacks that the opponent might carry out. These attacks are described as follows:

- **Impersonation**: the opponent introduces a message \( m \) into the channel, hoping to have it accepted as authentic by the receiver.

- **Substitution**: the opponent observes a message \( m \) in the channel, and then changes it to \( m' \), hoping for \( m' \) to be valid. We demand that message \( m \) and \( m' \) correspond to different source states.

Associated with each of these attacks is a **deception probability**, which represents the probability that the opponent will successfully deceive the receiver. We assume that the opponent chooses the message that maximises his chance of success. These probability are denoted by \( P_I \) for impersonation attack and \( P_S \) for substitution attack. They are formally defined as follows

\[
P_I \overset{\text{def}}{=} \max_m P(m \text{ is valid })
\]

\[
P_S \overset{\text{def}}{=} \max_{m, m' \in \mathcal{M}} P(m' \text{ is valid } \mid m \text{ is valid } ).
\]

In order to complete \( P_I \) and \( P_S \), we need to specify a probability distribution on \( S \) and \( \mathcal{E} \). This will induce a probability distribution on \( \mathcal{M} \). We shall adopt the
Kerckhoff’s principle that everything in the system except the actual key is public. That is, we assume that the authentication code and the probability distributions on $S$ and $E$ are known to the opponent. The only information that the sender and receiver possess that is unknown to the opponent is the value of the key $e$.

We define $E(m)$ as the set of keys for which the message $m$ is valid, i.e.,

$$E(m) = \{ e \in E : \exists s \in S, f(s, e) = m \}.$$ 

Then $P_I$ can be expressed as

$$P_I = \max_m P(E(m)).$$

Furthermore, if we assume that the probability distribution on $S$ and $E$ are uniform, the deception probability can be expressed as

$$P_I = \max_m \frac{|E(m)|}{|E|}.$$

$$P_S = \max_{m, m' \in M} \frac{|E(m) \cap E(m')|}{E(m)}.$$

Since the opponent can choose between the two attacks, we define the overall deception probability of an A-code, defined by $P_D$, as

$$P_D = \max\{P_I, P_S\}.$$ 

In an A-code without secrecy, the key is used to compute an authentication tag and concatenate to the source state as a message to be transmitted to the receiver. We will use the notation $(S, E, T, f)$ to denote A-code without secrecy.

**Definition 1.2** An A-code $(S, E, M, f)$ for which the mapping $f : S \times E \rightarrow M$ can be written as

$$f : S \times E \rightarrow S \times T, \quad f((s, e)) = (s, t),$$

where $s \in S, t \in T$, is called a systematic Cartesian A-code (or A-code without secrecy). The second part $t$ in the message is called the tag (or authenticator).

In an A-code without secrecy, the key is used to compute an authentication tag which is concatenated to the source state to form the message that is to be transmitted to the receiver. We will use the notation $(S, E, T, f)$ to denote A-code without secrecy.
In authentication theory, we sometimes consider multiple transmissions: that is, a key is used to authenticate multiple messages. An attack is said to be spoofing of order $r$ if the opponent has seen $r$ communicated messages and tries to construct a fraudulent message under a single key. The opponent's chance of success in this case is denoted by $P_r$. In particular, $P_0 = P_I$ and $P_1 = P_S$.

1.2 Bounds on the performance of the A-codes

1.2.1 Information-Theoretic Bounds for A-codes

In this subsection, we review some fundamental lower bounds on A-codes which are obtained by using information theory. The security and efficiency of an A-code ($S, E, M, f$) (or $(S, E, T, f)$ for A-code without secrecy) can be measured by a number of parameters: the deception probabilities $P_I$ and $P_S$, and the size of key space $|E|$, the size of the message spaces $|M|$ (or authentication tag $|T|$). The goal of authentication theory is to examine the relationships among these parameters and give constructions that for a given source and deception probabilities, have the shortest possible length for the key and transmitted message.

We assume that the reader is familiar with the basic concepts of information theory. A brief review is given in Appendix A. We use $X$ to denote a set and $X$ a random variable defined on $X$. Let $H(X)$ denote the entropy of the random variable $X$, and let $I(X;Y)$ denote the mutual information between $X$ and $Y$. We state Simmons' information-theoretic bounds.

**Theorem 1.1 (Simmons's bound [100, 26])** For any A-code $(S, E, M, f)$, we have

(i) $P_I \geq 2^{-I(M;E)}$;

(ii) $P_S \geq 2^{-I(M';E|M)}$,

where $E$ is the random variable defined on $E$, and $M', M$ are random variables on $M$ such that $m' \neq m, m, m' \in M$. In other words, $M' \times M$ are random variables on $M' \times M = \{(m', m) ; m' \neq m \in M\}$.

From Theorem 1.1, we have the following corollary.

**Corollary 1.2** In an A-code

$$P_S \geq 2^{-H(E|M)}.$$
1.2. Bounds on the performance of the A-codes

These bounds show how authentication codes provide protection. For the impersonation attack, we see that $P_I$ is lower bounded by the mutual information between the transmitted message and the key. This means that in order to have a good protection against impersonation attack, i.e., $P_I$ be small, we must give away a lot of information about the key. On the other hand, from Corollary 1.2 we know that in the substitution attack, $P_S$ is lower bounded by the uncertainty about the key when a message has been observed. Thus we cannot waste all the key entropy for protection against the impersonation attack, but some certainty about the key must remain for protection against the substitution attack.

A general form of Simmons' bounds for protection against spoofing of order $r$, proved independently by Rosenbaum [90] and Pei [80], is

$$P_r > 2^{-I(E;M'|M^r)},$$

where $I(E;M'|M^r)$ is the mutual information between a string of $r$ transmitted messages and the key.

1.2.2 Combinatorial Bounds of A-codes

In this subsection, we review some combinatorial lower bounds for A-codes.

In our model of A-code, each source state $s$ maps to a message $m$. We see among all the messages in $\mathcal{M}$, at least $|S|$ must be authentic, since every source state maps to a different message in $\mathcal{M}$. Similarly, for the substitution attack, after the observation of one legal message, at least $|S|-1$ of the remaining $|\mathcal{M}|-1$ messages must be authentic. Thus we have

**Theorem 1.3 ([64])** For any A-code $(S, \mathcal{E}, \mathcal{M}, f)$,

\[
\begin{align*}
P_I &\geq \frac{|S|}{|\mathcal{M}|}, \\
P_S &\geq \frac{|S|-1}{|\mathcal{M}|-1}.
\end{align*}
\]

The above theorem shows that in order to have good protection, i.e., $P_D$ be small, $|\mathcal{M}|$ must be chosen much larger than $|S|$. For a fixed source space, an increase in the authentication protection implies an increase in message size.

Next, by multiplying the two bounds in Theorem 1.1 and the bound in Corollary 1.2 together, we have

$$P_D^2 \geq P_I P_S \geq 2^{-I(M;E)-H(E|M)} = 2^{-H(E)}.$$ 

Since $H(E) \leq \log |\mathcal{E}|$, we obtain the famous square root bound.
1.2. Bounds on the performance of the A-codes

Theorem 1.4 (Square root bound [56]) For any A-code

\[ P_D \geq \frac{1}{\sqrt{\mathcal{E}}}. \]

Moreover, the above bound can be tight only if \(|S| \leq \sqrt{|\mathcal{E}|} + 1\).

The square root bound gives a direct relation between the key size and the protection that we can expect to obtain.

The following theorem follows immediately.

Theorem 1.5 In an A-code \((S, \mathcal{E}, \mathcal{M}, f)\), assume that \(P_D = 1/q\), then

(i) \(|\mathcal{E}| \geq q^2\);

(ii) \(|\mathcal{M}| \geq q|S|\).

We call an A-code optimal if the bounds (i), (ii) of Theorem 1.5 can be met with equality.

In the remainder of this subsection, we give some characterisations of A-codes from some combinatorial objects.

Given an A-code \((S, \mathcal{E}, \mathcal{M}, f)\), we can associate an \(|\mathcal{E}| \times |\mathcal{M}|\) matrix \(A\), called the incidence matrix, where \(A\) is a binary matrix whose rows are labelled by encoding rules (i.e., key) and columns by codewords (i.e., message), such that \(A(e, m) = 1\) if \(m\) is a valid codeword under \(e\), and \(A(e, m) = 0\), otherwise.

An authentication matrix \(B\) of an A-code without secrecy is a matrix of size \(|\mathcal{E}| \times |S|\) whose rows are labelled by the encoding rules, columns by the source states, and \(B(e, s) = t\) if \(t\) is the tag for the source state \(s\) under the encoding rule \(e\).

The following combinatorial bounds are the extensions of Theorem 1.3 for A-codes to protect against spoofing of order \(r\).

Theorem 1.6 (i) In an A-code with secrecy

\[ P_i \geq \frac{|S| - i}{|\mathcal{M}| - i}, \quad i = 1, 2, \ldots \]

(ii) In an A-code without secrecy

\[ P_i \geq \frac{|S|}{|\mathcal{M}|}, \quad i = 1, 2, \ldots \]
An A-code that satisfies (i) and (ii) of Theorem 1.6 with equality, that is with $P_i = \frac{|\mathcal{S}| - i}{|\mathcal{M}| - i}$ for A-code with secrecy and $P_i = \frac{|\mathcal{S}|}{|\mathcal{M}|}$ for A-code without secrecy, is said to provide **perfect protection for spoofing of order $i$**. The opponent's best strategy in spoofing of order $i$ for such an A-code is to randomly select one of the remaining codewords.

A-codes that provide perfect protection for all orders of spoofing up to $r$ are said to be **$r$-fold secure**. These codes can be characterised by using combinatorial structures such as orthogonal arrays and $t$-designs.

An **orthogonal array** $OA(t, k, v)$ is an array with $\lambda v^t$ rows, each row of size $k$, from the elements of set $X$ of $v$ symbols, such that in any $t$ columns of the array every $t$-tuple of elements of $X$ occurs in exactly $\lambda$ rows. Usually $t$ is referred to as the **strength** of the OA.

**Example 1.1** *The following table gives a $OA_2(2, 5, 2)$ on the set $\{0, 1\}$:*

$$
\begin{array}{cccccc}
0 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 1 \\
1 & 1 & 0 & 1 & 1 \\
1 & 0 & 1 & 0 & 1 \\
0 & 1 & 1 & 0 & 1 \\
1 & 0 & 1 & 1 & 0 \\
0 & 1 & 1 & 1 & 0 \\
\end{array}
$$

A $t - (v, k, \lambda)$ **design** is a collection of $b$ subsets, each of size $k$, of a set, $X$, of size $v$ where every subset of size $t$ occurs exactly $\lambda$ times.

The **incidence matrix** of a $t - (v, k, \lambda)$ design is a binary matrix, $A = (a_{ij})$, of size $b \times v$ such that $a_{ij} = 1$ if element $j$ is in block $i$ and 0 otherwise.
1.2. Bounds on the performance of the A-codes

Example 1.2 The following table gives a $3-(8, 4, 1)$ design on the set $\{0, 1, 2, 3, 4, 5, 6, 7\}$:

\[
\begin{array}{cccc}
7 & 0 & 1 & 3 \\
7 & 1 & 2 & 4 \\
7 & 2 & 3 & 5 \\
7 & 3 & 4 & 6 \\
7 & 4 & 5 & 0 \\
7 & 5 & 6 & 1 \\
7 & 6 & 0 & 2 \\
2 & 4 & 5 & 6 \\
3 & 5 & 6 & 0 \\
4 & 6 & 0 & 1 \\
5 & 0 & 1 & 2 \\
6 & 1 & 2 & 3 \\
0 & 2 & 3 & 4 \\
1 & 3 & 4 & 5 \\
\end{array}
\]

with incidence matrix:

\[
\begin{array}{cccccccccccc}
1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \\
0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 \\
0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 \\
0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \\
1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \\
0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\
1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\
0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 \\
1 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\
1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \\
1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\
0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 \\
1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\
0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 \\
\end{array}
\]

The main theorem relating A-codes with $r$-fold security and combinatorial structures are due to a number of authors, including Stinson [105], and Tombak and Safavi-Naini [114]. The following are the most general forms of these theorems.

**Theorem 1.7 ([114])** Let the source be $r$-fold uniform. Then an A-code provides
r-fold security against spoofing if and only if the incidence matrix of the code is the incidence matrix of a \((r + 1) - (|\mathcal{M}|, |\mathcal{S}|, \lambda)\) design.

In the above theorem, an \(r\)-fold uniform source is a source for which every string of \(r\) distinct source states has probability \(\frac{1}{|\mathcal{S}|(|\mathcal{S}|-1)\cdots(|\mathcal{S}|-r+1)}\).

**Theorem 1.8** Let \(P_0 = P_1 = \cdots = P_r = |\mathcal{S}|/|\mathcal{M}|\). Then the authentication matrix is an \(OA(r + 1, |\mathcal{S}|, \ell)\), where \(\ell = |\mathcal{M}|/|\mathcal{S}|\).

1.3 Other Issues

1.3.1 Other types of attacks

Tombak and Safavi-Naini [114] consider other types of attacks, similar to those for secrecy systems. In a plaintext attack against A-codes with secrecy, the enemy not only knows the codeword but also knows the corresponding plaintext. In chosen content attack the enemy wants to succeed with a codeword that has a prescribed plaintext. It is shown that by applying some transformation on the A-code it is possible to provide immunity against the above attacks.

A-codes with secrecy are generally more difficult to analyse than Cartesian A-codes. Moreover, the verification process for the former is not as efficient. In the case of Cartesian A-codes, verification of a received codeword, \((s, t)\) amounts to recalculating the tag using the secret key and the source state \(s\) to obtain \(t'\) and comparing it with the received tag \(t\). For an authentic codeword we have \(t = t'\). In the case of A-codes with secrecy, when \(m\) is received, the receiver must compare \(m\) with all authentic codewords under his secret key to see if \(m\) is valid. Otherwise there must be an inverse algorithm that allows the receiver to find and verify the source state. The former process is costly and the later does not exist for a general A-code. For these reasons, the majority of research has been concentrated on Cartesian A-codes.

1.3.2 Authentication codes and error-correcting codes

An error correcting code (EC-code) provides protection against random channel error. Study of error correcting code was motivated by Shannon's channel capacity theorem and has been an active research area since early 1950s. Error correcting codes add
redundancy to a message in such a way that a codeword corrupted by the channel noise can be detected and/or corrected. The main different between an A-code and EC-code is that in the former redundancy depends on a secret key while in the latter it only depends on the message being coded. There exists a duality between authentication codes and error correcting codes. In words of Simmons [99], "...one (coding theory) is concerned with clustering the most likely alterations as closely about the original code as possible and the other (authentication theory) with spreading the optimal (to the opponent) alterations as uniformly as possible".

The relationship between EC-code and A-codes is explored in the work of Johansson et al [62], who showed that it is possible to construct EC-codes from A-codes and vice versa. Their work uses a worst case analysis approach in analysing the security of A-codes. That is, in the case of substitution attack, they consider the best chance of success an enemy has when the transmitted message ranges over all possible messages. This contrasts with the information theoretic (or game theory) approach in which the average success probability of the enemy over all possible codewords is calculated.

The work of Johansson et al [62] is especially useful as it allows bounds and asymptotic results from the theory of error correcting codes to be employed to derive upper and lower bounds on the size of the source for A-codes with given $E$, $T$, and $P_D$.

### 1.3.3 Authentication with arbiter

In the basic model of authentication discussed above, the enemy is an outsider and we assume that the transmitter and the receiver are trustworthy. Moreover, because the key is shared by the transmitter and the receiver, the two principals are cryptographically indistinguishable. In an attempt to model authentication systems in which the transmitter and the receiver are distinguishable and to remove assumptions about the trustworthiness of the two, Simmons [99] introduced a fourth principal called the arbiter. The transmitter and receiver have different keys and the arbiter has access to all or part of the key information. The system has a *key distribution phase* during which the transmitter uses its key to produce a codeword and finally a *distribution phase* during which disputes are resolved with the aid of the arbiter. The arbiter in Simmons’ model is active during the transmission phase and is assumed to be trustworthy. Yung and Desmedt [122] removed this assumption and considered a model in which the arbiter is only trusted to resolve disputes. Johansson [63] and Kurosawa [70] derived lower bounds on the probability of deception in such codes. Johansson [65] and Taylor [113] proposed constructions.
1.3.4 Shared generation of authenticators

Many applications require the power to generate an authentic message and/or to verify the authenticity of a message to be distributed among a number of principals. An example of such a situation is multiple signatures required for a bank account or in a court room. Desmedt and Frankel [39] introduced systems with shared generation of authenticators (SGA-systems). Safavi-Naini [91], van Dijk, Gehrmann and Smeets [44], and Martin and Safavi-Naini [73] further studied such systems. In a SGA-system there is a group $V$ of transmitters endowed with an access structure $T$ that determines authorised subsets of $V$. Each principal has a secret key which is used to generate a partial tag. The system has two phases. In the key distribution phase, a trusted authority generates keys for transmitters and the receiver and securely delivers the keys to them. In the communication phase, the trusted authority is not active. When an authorised group of transmitters want to construct an authentic codeword, using their key information, each group member generates a partial tag for the source state $s$ which needs to be authenticated and sends it to a combiner. The combiner is a fixed algorithm with no secret input that combines codeword using its secret key. Martin and Safavi-Naini [73] give a general construction for SGA-systems by combining A-codes and secret sharing schemes. van Dijk et al [44] propose an efficient construction for SGA-systems, based on maximum rank distance separable codes.

1.3.5 Broadcast authentication codes

Another interesting extension of Simmons' traditional authentication code is broadcast authentication codes, or multireceiver authentication codes (MRA-codes for short). In an MRA-code, a sender wants to authenticate a message for a group of receivers such that each receiver can verify authenticity of the received message. The receiver are not all trusted and may collude to construct fraudulent messages on behalf of the transmitter. This is a useful extension of traditional point-to-point authentication code and has numerous applications. For example, a director wants to give instructions to employees in an organisation such that each employee is able to verify authenticity of the received message. Providing such service using digital signature implies that the security is based on unproved assumptions.

Unconditionally secure multireceiver authentication codes are introduced by Desmedt, Frankel and Yung in [41]. They have been further studied by other authors (see [52, 71, 66, 111, 93, 94, 95]).
1.3.6 Multiple authentication

As noted before, in the theory of A-codes possible attacks by the enemy are limited to impersonation and substitution. This means that the security of the system in only for one message and after that the key must be changed. To extend protection over more than one message transmission, a number of alternatives exist. The most obvious one is to use A-codes that provide protection against spoofing of order \( \ell \). However, little is known about construction of codes for multiple messages. Vanroose, Smeets and Wan [119] suggested key strategies in which the communicants change their key after each transmitted codeword, using some pre-specified strategy. In this case the key information shared by the communicants is the sequence of keys to be used for consecutive transmission slots. The resulting bounds on the probability of deformation generalise the bounds given by Pei [80], Rosenbaum [90] and Walker [120]. Another successful approach proposed by Wegman and Carter [121] uses a special class of hash functions together with one time pad of random numbers.

1.4 Computationally Secure Authentication Systems

A message authentication code (MAC) is a symmetric key cryptographic primitive that provides computational security against authentication attacks. Although, in this thesis, we will only consider unconditional secure A-codes, a brief introduction to MACs is included for the sake of completeness.

Message authentication codes (MACs) provide message integrity and are one of the most important security primitives in current distributed information systems. Informally, A MAC consists of two algorithms. A MAC generation algorithm, \( G = \{G_k; k = 1, \ldots, N\} \) takes an arbitrary message, \( s \), from a given collection \( S \) of messages and produces a tag, \( t = G_k(s) \), which is appended to the message to produce an authentic message \( m = (s, t) \). A MAC verification algorithm, \( V = \{V_k : k = 1, \ldots, N\} \), takes authenticated messages of the form \( (s, t) \) and produces a true or false value, depending on whether the message is authentic or not. The security of a MAC is measured by the best chance of an active spoofer to successfully substitute a received message \( (s, G_k(s)) \) with a fraudulent one, \( m' = (s', t) \), such that \( V_k(m') \) produces a true result. In MAC systems, the communicants share a secret key, and are therefore not distinguishable cryptographically.

Security of MACs can be studied from the point of view of unconditional or computational security.
Unconditionally secure MACs are equivalent to Cartesian authentication codes. However, in MAC systems only multiple communications are of interest. Computationally secure MACs have arisen from the needs of the banking community; see, for example, Preneel, Chaum, Fumy, Jansen, Landrock and Roelofsen [82]. They are also studied under other names, such as 
\textit{keyed hash functions} and \textit{keying hash functions} (see, for example, [5]).

\subsection{Unconditionally secure MACs}

When the enemy has unlimited computational resources, attacks against MAC systems and the analysis of security are similar to that of Cartesian A-codes. The enemy observes \( n \) codewords of the form \((s_i, t_i), i = 1, \ldots, n\), in the channel and attempts to construct a fraudulent codeword \((s, t)\) which is accepted by the receiver. (This is the same as spoofing of order \( n \) in an A-code). If the communicants want to limit the enemy’s chance of success to \( p \) after \( n \) message transmissions, the number of authentication functions (number of keys) must be greater than a lower bound which depends on \( p \). If the enemy’s chance of success in spoofing of order \( i, i = 1, \ldots, n \), is \( p_i \), then at least \( 1/p_1 p_2 \cdots p_n \) keys are required [48], [121]. For \( p_i = p, i = 1, \ldots, n \), the required number of key bits is \(-n \log_2 p\). That is, for every message, \(- \log_2 p\) key bits are required. This is the absolute minimum for the required number of key bits.

Perfect protection is obtained when the enemy’s best strategy is random choice of a tag and appending it to the message; this strategy succeeds with probability \( p = 2^{-k} \), if the size of the tag is \( k \) bits. In this case the number of required key bits for every extra message is \( k \).

Wegman and Carter [121] gave a general construction for unconditionally secure MACs that can be used for providing protection for an arbitrary number of messages.

Their construction uses \textit{universal classes of hash functions}. Traditionally, a hash function is used to achieve fast and good average performance over all inputs in various applications. By using a universal class of hash functions it is possible to achieve provable good average performance without restricting the input distribution.

Let \( h : A \rightarrow B \) be a hash function mapping the elements of a set \( A \) to a set \( B \). A \textit{strongly universal} \( n \) class of hash functions is a class of hash functions with the property that for \( n \) distinct elements \( a_1, \ldots, a_n \) of \( A \) and \( n \) elements \( b_1, \ldots, b_n \) of \( B \), exactly \( |H|/b^n \) functions map \( a_i \) to \( b_i \), for \( i = 1, \ldots, n \). Strongly universal \( n \) hash functions give perfect protection for multiple messages as follows. The transmitter and the receiver use a publicly known class of strongly universal \( n \) hash functions, and a shared secret key
1.4. Computationally Secure Authentication Systems

determines a particular member of the class used by the transmitter and the receiver. Stinson [107] shows that a class of strongly universal\(\lambda\) that maps a set of \(a\) elements to a set of \(b\) elements is equivalent to an orthogonal array \(OA_{\lambda}(2, a, b)\) with \(\lambda = |H|/b^2\). Similar results can be proved for strongly universal\(n\) classes of hash functions.

1.4.2 Wegman and Carter construction

Wegman and Carter [121] show that, instead of strongly universal\(n\) family of hash functions one can always use a strongly universal\(\lambda\) family of hash functions, together with a one time pad of random numbers. The system works as follows. Let \(B\) denote the set of tags consisting of the sequences of \(k\) bit strings. Let \(\mathcal{H}\) denote a strongly universal\(\lambda\) class of hash functions mapping \(S\) to \(B\). Two communicants share a key that specifies a function \(h \in \mathcal{H}\) together with a pad containing \(k\)-bit random numbers. The tag for the \(j\)th message is \(s_j \oplus r_j\), where \(r_j\) is the \(j\)th number on the pad. It can be proved that this system limits the enemy’s chance of success to \(2^{-k}\) as long as the pad is random and not used repeatedly. The system requires \(nk + K\) bits of key, where \(K\) is the number of bits required to specify an element of \(\mathcal{H}\), \(n\) is the number of messages to be authenticated, and \(k\) is the size of the tags.

This construction has a number of remarkable properties. Firstly, for large \(n\) the key requirement for the system approaches the theoretical minimum of \(k\) bits per message. This is because for large \(n\) the number of key bits is effectively determined by \(nk\). Secondly, the construction of MAC for multiple communications is effectively reduced to the construction of a better studied primitive, that is, strongly universal\(\lambda\) class of hash functions. Finally, by replacing the one-time pad with a pseudorandom sequence generator, unconditional security is replaced by computational security.

Wegman and Carter’s important observation is as follows. By not insisting on the minimum value for the probability of success in spoofing of order one, it is possible to reduce the number of functions and thus the required number of keys. This observation leads to the notion of almost strongly universal class.

An \(\epsilon\) - almost universal (or \(\epsilon\)-AU) class of hash functions has the following property. For any pair \(x, y \in A, x \neq y\), the number of hash functions \(h\) with \(h(x) = h(y)\) is at most equal to \(\epsilon|\mathcal{H}|\). The \(\epsilon\)-almost strongly universal (or \(\epsilon\)-ASU) hash functions have the additional property that for any \(x \in A, y \in B\) the number of functions with \(h(x) = y\) is \(|H|/|B|\). Using an \(\epsilon\)-almost strongly universal class of functions in the Wegman and Carter construction results in MAC systems for which the probability of success for an intruder is \(\epsilon\).
Stinson [107] gives several methods for constructing AU and ASU hash functions. Johansson et al [62] establish a relationship between ASU hash functions and error correcting codes. They use geometric error correcting code to construct new classes of $\epsilon$-ASU hash function of small size. This reduces the key size.

Krawczyk [69] shows that in the Wegman-carter construction, $\epsilon$-ASU hash functions can be replaced with a less demanding class of hash functions, called $\epsilon$-otp-secure. The definition of this class differs from other classes of hash functions, in that it is directly related to MAC constructions and their security; in particular, to the Wegman-Carter construction.

Let $s \in S$ denote a message that is to be authenticated by a $k$ bit tag $h(s) \oplus r$, using by Wegman and Carter’s method. An enemy succeeds in a spoofing attack if he can find $s' \neq s, t' = h(s') \oplus r$, assuming that he knows $H$ but does not know $h$ and $r$. A class $H$ of hash functions is $\epsilon$-otp-secure if for any message no adversary succeeds in the above attack scenario with probability greater than $\epsilon$. In [69], Krawczyk shows that a necessary and sufficient condition for a family $H$ of hash functions to be $\epsilon$-otp-secure is

$$\forall a_1 \neq a_2, c, Pr_h(h(a_1) \oplus h(a_2) = c) \leq \epsilon,$$

where $a_1, a_2 \in A$ and $c \in B$ are in the form of bit strings. The need for high speed MACs has increased with the progress in high speed data communication. A successful approach to the construction of such MACs uses hash function families in which the message is hashed by multiplying it by a binary matrix. Because hashing is achieved with exclusive-or operations, it can be efficiently implemented in software. An obvious candidates for such a class of hash functions, originally proposed by Wegman and Carter [29, 121], is the set of linear transformations from $A$ to $B$. It is shown that this forms an $\epsilon$-AU class of hash functions. However the size of the key - the number of entries in the matrix - is too large, and too many operations are required for hashing. Later proposals by Krawczyk [69] and by Rogaway [89] are aimed at alleviating these problems, and obtains a fast software implementation. The former uses Toeplitz matrices, while the latter uses binary matrices with only three non-zero entries per column. In both cases, the resulting family is $\epsilon$-AU.

The design of a complete MAC usually involves combining a number of hash functions following results proved by Stinson [107]. The role of some of the hash functions is to produce high compression (small $b$), while others produce the desired spread and uniformity (see Rogaway [89]).

Reducing the key size of the hash function is especially important in practical applications, because the one-time pad is replaced by the output of a pseudorandom
1.4. Computationally Secure Authentication Systems

Generator with a short key (of the order of 128 bits). Hence it is desirable to have the key size of the hash function of similar order.

1.4.3 Computational security

In the computationally secure approach, the protection is achieved because excessive computation is required for a successful forgery. Although a hash value can be used as a checksum to detect random changes in the data, a secret key must be used to provide protection against active tampering. Methods for constructing MACs from hash functions have traditionally followed one of the following approaches: the so-called hash-then-encrypt and keying a hash function.

Hash-then-encrypt

To construct a MAC for a message \( x \) with this method, the hash value of \( x \) is calculated and the result is encrypted using an encryption algorithm. This is similar to signature generation, where a public key algorithm is replaced by a private key encryption function.

There are a number of drawbacks to this method. First, the overall scheme is slow. This is because the two primitives used in the construction, i.e., the cryptographic hash functions and encryption functions, are designed for other purposes and have extra security properties which are not strictly required in the construction. Although this construction can produce a secure MAC, the speed of the MAC is bounded by the speed of its constituent algorithms. For example, cryptographic hash functions are designed to be one-way. It is not clear whether this is a required property in the hash-then-encrypt construction, where the output of the hash function is encrypted and one-wayness is effectively obtained through the difficulty of finding the plaintext from the ciphertext.

A serious shortcoming of this method is that existing export restrictions, which usually apply to encryption functions, are inherited by MACs constructed using this method.

Keying a hash function

In the second approach a secret key is incorporated into a hashing algorithm. This operation is sometimes called keying a hash function (see Bellare et al [9]. This method is attractive, because of the availability of hashing algorithms and their relative speed in software implementation. Moreover the algorithms are not subject to export restrictions.
Although this scheme can be implemented more efficiently in software than the previous scheme, the objection to the superfluous properties of the hash functions remains.

The keying method depends on the structure of the hash function. Tsudik [117] proposes three methods of incorporation the key into the data. In the secret prefix method, the key is prepended to the data, $G_k(s) = H(k \parallel s)$, while in the secret suffix method it is appended to this data and we have $G_k(s) = H(s \parallel k)$. Finally, the envelope method combines the previous two methods with $G_k(s) = H(k_1 \parallel s \parallel k_2)$ and $k = k_1 \parallel k_2$.

Instead of including the key into the data, the key information can be included into the hashing algorithm. In iterative hash functions such as MD5 and SHA, the key can be incorporated into the initial vector, compression function or into the output transformation.

There have also been some attempts at defining and constructing secure keyed hash functions as independent primitives, namely by Berson, Gong and Lomas [14], and Bakhtiari, Safavi-Naini and Pieprzyk [5]. The former proposes a set of criteria for secure keyed hash functions and give constructions using one-way hash functions. The latter argues that the suggested criteria for security is in most cases excessive and relaxing them allows constructions of more efficient secure keyed hash functions. Bakhtiari et al also give the design of a keyed hash function from scratch. Their design is based mostly on intuition principles and lacks a rigorous proof of security. A similar approach is taken in the design of MDx-MAC by Preneel and van Oorschot [83], which is a scheme for constructing a MAC from an MD5-type hash function. It is conjectured that MDx-MAC is a secure MAC.

1.4.4 Security analysis of computationally secure MACs

The security analysis of computationally secure MACs has followed two different approaches. In the first approach, the security assessment is based on an analysis of some possible attacks. In the second approach, a security model is developed and used to examine the proposed MAC.

Security analysis through attacks

Consider a MAC algorithm that produces MACs of length $m$ using a $k$ bit key. In general an attack might result in a successful forgery, or in the recovery of the key. According to the classification given by Preneel and van Oorschot [84], a forgery in a MAC can be either existential - the opponent can construct a valid message and
MAC pair without the knowledge of the key, or selective where the opponent can determine the MAC for a message of his choice. Protection against the former type of attack imposes more stringent conditions than the latter type of attack. A forgery is verifiable if the attacker can determine with a high probability whether the attack is successful. In a chosen text attack the attacker is given the MACs for the messages of his own choice. In an adaptive attack the attacker chooses text for which he can see the result of his previous request before forming his next request. In a key recovery attack the aim of the attacker is to find the key. If the attacker is successful, he can perform selective forgery on any message of his choice and the security of the system is totally compromised.

For an ideal MAC any method to find the key is as expensive as an exhaustive search of $O(2^k)$ operations. If $m < k$, the attacker may randomly choose the MAC for a message with the probability of success equal to $1/2^m$. However, in this attack the attacker cannot verify whether his attack has been successful.

The complexity of various attacks is discussed by several authors: Tsudik [117], Bakhtiari et al [6], Bellare et al [10]. Preneel and van Oorschot [83, 84] propose constructions resistant to such attacks. Some attacks can be applied to all MACs obtained using a specific construction method while other attacks are limited to particular instances of the method.

1.4.5 Formal security analysis

The main attempts at formalising the security analysis of computationally secure MACs are due to Bellare et al [9], and Bellare and Rogaway [11]. In both papers, an attack model is carefully defined and the security of a MAC with respect to that model is evaluated. Bellare et al [9] use their model to prove the security of a generic construction based on pseudorandom functions, while Bellare and Rogaway [11] use their model to prove the security of a generic construction based on hash functions.

MAC from pseudorandom functions

The formal definition of security given by Bellare et al [11] assumes that the enemy can ask the transmitter to construct tags for messages of his choice, and also ask the receiver to verify chosen message and tag pairs of his choice. The number of these requests is limited, and a limited time $t$ can be spent on the attack. Security of the MAC is expressed as an upper bound on the enemy's chance of succeeding in its best attack.

The construction proposed by Bellare et al applies to any pseudorandom functions.
1.4. Computationally Secure Authentication Systems

Their proposal, called XOR-MAC, basically breaks a message into blocks. For each block the output of the pseudorandom function is calculated on the black, and the outputs are finally XORed. Two schemes based on this approach are proposed: the randomised XOR scheme and the counter based scheme.

The pseudorandom function used in the above construction can be an encryption function, like DES, or a hash function, like MD5. It is proved that the counter based scheme is more secure than the randomised scheme, and if DES is used, both schemes are more secure than CBC MAC. Some of the desirable features of this construction are parallelizability and incrementality. The former means that message blocks can be fed into the pseudorandom function in parallel. The latter refers to the feature of calculating incrementally the value of the MAC for a message $s'$ which differs from $s$ in only a few blocks.

**MAC from hash functions**

The model used by Bellare and Rogaway [11] is similar to the above one. The enemy can obtain information by asking queries; however, in this case queries are only addressed to the transmitter.

A family of functions $\{F_k\}$ is an $(\epsilon, t, q, L)$-secure MAC [10] if any adversary that is not given the key $k$, is limited to spend total time $t$, and sees the values of the function $F_k$ computed on $q$ messages $s_1, s_2, \ldots, s_q$ of its choice, each of length at most $L$, cannot find a message and tag pair $(s, t), s \neq s_i, i = 1, \ldots, q$, such that $t = F_k(s)$ with probability better than $\epsilon$.

Two general constructions for MAC from hash functions, the so-called NMAC, the Nested construction; and HMAC, the Hash based MAC, are given and their security is formally proved [11].

Weak collision-resistance is a much weaker notion than the collision resistance of (unkeyed) hash functions because the enemy does not know the secret key and finding collision is much more difficult. More precisely, a family of keyed hash functions $\{F_k\}$ is $(\epsilon, t, q, L)$-weakly collision-resistant if any adversary that is not given the key $k$, is limited to spend total time $t$, and sees the values of the function $F_k$ computed on $q$ messages $m_1, m_2, \ldots, m_q$ of its choice, each of length at most $L$, cannot find messages $m$ and $m'$ for which $F_k(m) = F_k(m')$ with probability better than $\epsilon$.

With some extra assumptions similar results are proved for the HMAC construction.

A related construction is the collisionful keyed hash function proposed by Gong [57]. In his construction, the collisions are selectable and the resulting function is claimed to provide security against password guessing attacks. Bakhtiari, Safavi-Naini and
Pieprzyk [7, 8] question the security of Gong's function and a key exchange protocol based on collisionful hash functions and show attacks on their systems.
Chapter 2

Authentication Codes and Universal Hashing

In this chapter, we study the application of algebraic curves over finite fields to the constructions of universal hash functions and unconditionally secure authentication codes.

The concept of universal hash family was introduced by Carter and Wegman [29] in 1979, and has found numerous applications in computer science, including cryptography, complexity theory, search algorithms and information retrieval, etc. In cryptography, one of the most interesting classes of universal hash functions is called strongly universal hash family, also known as two-point based sampling, or pairwise independent random variable [109]. It is known [121, 107] that the strongly universal hash families provide a very powerful tool for constructing unconditionally secure authentication codes when the number of authenticators is exponentially small compared to the number of possible source states.

The theory of algebraic curves over finite fields has many applications in coding theory ([116]) and it has been observed by several authors that coding theory, universal hash families and authentication codes are closely related. (More details will be given in Section 2.1) This has led researchers to naturally consider the algebraic-geometric codes based on some specified algebraic curves to construct universal hash families and authentication codes [16, 17]. In this chapter, we will pursue this direction of research. However, unlike the earlier approach, which composes a family of efficient geometric codes to obtain an strongly universal hash family, known as "composition method", and whose detail will be given in Section 2.1, we will instead provide a direct construction of strongly universal hash families from the algebraic curves over finite fields without using the composition method. We also note that direct construction of universal hash families without using composition method has been also proposed by Helleseth and Johansson [59]. They used exponential sums over finite fields and obtained strongly
universal hash families and authentication codes that had the best performance compared to those previously known. We show that our constructions result in new classes of strongly universal hash families and authentication codes which are even better than those of Helleseth and Johansson [59], and so yield new authentication codes with the best performance known so far.

In Section 2.1, we review some basic definitions of universal hash families, and their connections with authentication codes. In Section 2.2 we present our new constructions from algebraic curves. We compare various parameters of the constructions with previously known results in Section 2.3, and finally conclude the chapter in Section 2.4.

2.1 Universal hashing

Universal families of hash functions were introduced by Carter and Wegman [29], and were further studied by many authors, we refer to [109] for a through account of recent developments in this field. We are interested in the application of universal hashing to authentication codes.

Consider a hash family \( \mathcal{H} \), which is a set of \( N \) functions such that \( h : A \rightarrow B \) for each \( h \in \mathcal{H} \), where \(|A| = k\) and \(|B| = \ell\). There will be no loss of generality in assuming \( k \geq \ell \) and we call \( \mathcal{H} \) an \((N; k, \ell)\) hash family. We first review the relevant definitions and results as follows.

**Definition 2.1** An \((N; k, \ell)\) hash family is called \(\epsilon\)-almost universal (\(\epsilon\)-AU for short) if for any two distinct elements \(a_1, a_2 \in A\), there are at most \(\epsilon N\) functions \(h \in \mathcal{H}\) such that \(h(a_1) = h(a_2)\).

The following lemma, due to Bierbrauer, Johansson, Kabatianskii and Smeets [17], establishes the equivalence between \(\epsilon\)-AU and error-correcting codes.

**Lemma 2.1** If there exists a \(q\)-ary code with codeword length \(N\), cardinality \(M\), and minimum Hamming distance \(d\), then there exists an \(\epsilon\)-AU \((N; M, q)\) hash family, where \(\epsilon = 1 - d/N\). Conversely, if there exists an \(\epsilon\)-AU hash family, then there exists a code with parameters as above.

**Definition 2.2** An \((N; k, \ell)\) hash family is called \(\epsilon\)-almost strongly universal (\(\epsilon\)-ASU for short) if,

1. for any element \(a \in A\) and any element \(b \in B\), there exist exactly \(N/\ell\) functions \(h \in \mathcal{H}\) such that \(h(a) = b\).
2. for any two distinct elements \( a_1, a_2 \in A \) and for any two (not necessarily distinct) elements \( b_1, b_2 \in B \), there exist at most \( \epsilon N / \ell \) functions \( h \in \mathcal{H} \) such that \( h(a_i) = b_i, i = 1, 2 \).

\( \epsilon \)-ASU are strictly related to A-codes. An A-code, \((S, E, T)\), with (authentication) mapping \( f : S \times E \rightarrow T \), \( P_l = 1/|T| \) and \( P_S \), associates a unique function \( h_e \) from \( S \) to \( T \) for each key \( e \in E \), defined by \( h_e(s) = f(s, e) \). It is straightforward to verify that \( \mathcal{H} = \{ h_e : e \in E \} \) is an \( \epsilon \)-ASU hash family from \( S \) to \( T \), where \( \epsilon = P_S \). Conversely, given an \( \epsilon \)-ASU \((N; k, \ell)\) hash family \( \mathcal{H} \) from \( A \) to \( B \), we can associate an A-code \((S, E, T)\), where \( S = A \), \( T = B \) and \( |E| = |\mathcal{H}| \), and each key \( e \in E \) corresponds to a unique hash function \( h_e \in \mathcal{H} \) indexed by \( e \). The (authentication) mapping \( f : S \times E \rightarrow T \) is defined by \( f(s, e) = h_e(s) \). It has been observed by several authors \([17, 107]\) that the resulting A-code has \( P_l = 1/\ell \) and \( P_S = \epsilon \). In summary, we have the following result.

**Lemma 2.2** If there exists an A-code \((S, E, T)\) with \( P_l = 1/|T| \) and \( P_S \), then there exists an \( \epsilon \)-ASU \((N; k, \ell)\) hash family, where \( \epsilon = P_S \), \( N = |E| \), \( k = |S| \) and \( \ell = |T| \). Conversely, if there exists an \( \epsilon \)-ASU hash family, then there exists an A-code with the above parameters.

Earlier research on A-codes had been mainly devoted to the constructions which ensure that the opponent’s deception probabilities are bounded by \( 1/\ell \). In terms of \( \epsilon \)-ASU hash families, \( \epsilon = 1/\ell \). Such codes were shown to be equivalent to orthogonal arrays, and from the results in \([107]\) we know that \( |E| \geq k(\ell - 1) + 1 \). This means that for fixed security (i.e. \( 1/\ell \)), the key size increases linearly with the size of the set of source states—a similar situation with the “one-time pad”. Thus for the large size sources, many bits of keys are to be stored and “secretly” exchanged.

The significance of \( \epsilon \)-ASU hash families in the construction of A-codes that was first observed by Wegman and Carter \([121]\), is that by not requiring the deception probability to be the theoretical minimum, that is allowing \( \epsilon > 1/\ell \), we can expect to significantly reduce the key size. As shown in \([17, 107, 121]\), by allowing \( P_S > P_l \) (i.e. \( \epsilon > 1/\ell \)), it is possible to have the size of source states to grow exponentially with the key size. This observation is very important from practical point of view, we may deal with the scenarios where we are satisfied with deception probability slightly larger than \( 1/\ell \), but have limited key storage.

A very useful method of constructing an \( \epsilon \)-ASU hash family is to compose an \( AU \) hash family and an \( ASU \) hash family with appropriate parameters. The following
lemma is due to Stinson [107], and independently proved by Bierbrauer, Johansson, Kabatianskii and Smeets [17].

**Lemma 2.3 (Composition)** Let $\mathcal{H}_1$ be an $\epsilon_1$-AU hash family from $A_1$ to $B_1$ and let $\mathcal{H}_2$ be an $\epsilon_2$-ASU hash family from $B_1$ to $B_2$. Then

$$\mathcal{H} = \{h_2h_1 : h_1 \in \mathcal{H}_1, h_2 \in \mathcal{H}_2\}$$

is an $\epsilon$-ASU hash family from $A_1$ to $B_2$ with $\epsilon < \epsilon_1 + \epsilon_2$.

It is worth noting that previous constructions of $\epsilon$-ASU hash families for large $|A|$ were based on the above composition construction. The constructions giving good performance use Reed-Solomon codes [17, 107], or more general geometric codes [16] as the $\epsilon$-AU hash family in the above composition construction. Helleseth and Johansson [59] gave the first direct construction of $\epsilon$-ASU hash family without using the above composition construction, by using exponential sums. For large $|A|$, the construction results in $\epsilon$-ASU hash family whose performance was better than all the previously known constructions. The goal of this chapter is to give another direct construction (without using composition) of $\epsilon$-ASU hash families and authentication codes by using algebraic curves over finite fields. The constructions result in new classes of A-codes with better performance than those previously known.

### 2.2 Constructions

In this section, we describe a construction of $\epsilon$-ASU hash families based on algebraic curves over finite fields which can later be used to construct A-codes using Lemma 2.2.

Before giving the construction we need some concepts and notations. For further results on algebraic curves over finite fields, we refer to [103, 116].

We fix some notations for this Chapter.

- $\ell$ – power of a prime;
- $\mathbb{F}_\ell$ – the finite field of $\ell$ elements;
- $\mathcal{X}$ – a projective, absolutely irreducible, complete algebraic curve defined over $\mathbb{F}_q$. We simply say that $\mathcal{X}$ is an algebraic curve;
- $g = g(\mathcal{X})$ – the genus of $\mathcal{X}$;
- $\mathbb{F}_\ell(\mathcal{X})$ – the function field of $\mathcal{X}$;
- $\mathcal{X}(\mathbb{F}_\ell)$ – the set of all $\mathbb{F}_\ell$- rational points on $\mathcal{X}$ with all coordinates belonging to $\mathbb{F}_\ell$. 
A divisor $G$ of $\mathcal{X}$ is called *rational* if

$$G^\sigma = G$$

for any automorphism $\sigma \in \text{Gal}(\overline{F}/F)$, where $\overline{F}$ is a fixed algebraic closure of $F$ and $\text{Gal}(\overline{F}/F)$ is the Galois group of $\overline{F}/F$. In this paper we always mean a rational divisor whenever a divisor is mentioned.

We write $\nu_P$ for the normalised discrete valuation corresponding to the point $P$ of $\mathcal{X}$. Let $x \in F^\ell(\mathcal{X})\setminus\{0\}$ and denote by $Z(x)$ and $N(x)$, the set of zeros and poles respectively, of $x$. We define the *zero divisor* of $x$ by

$$(x)_0 = \sum_{P \in Z(x)} \nu_P(x)P \quad (2.1)$$

and the *pole divisor* of $x$ by

$$(x)_\infty = \sum_{P \in N(x)} (-\nu_P(x))P. \quad (2.2)$$

Then $(x)_0$ and $(x)_\infty$ are both rational divisors. Furthermore, the *principal divisor* of $x$ is given by

$$\text{div}(x) = (x)_0 - (x)_\infty. \quad (2.3)$$

The degree of $\text{div}(x)$ is equal to zero, i.e.,

$$\deg((x)_0) = \sum_{P \in Z(x)} \nu_P(x) = \sum_{P \in N(x)} (-\nu_P(x)) = \deg((x)_\infty). \quad (2.4)$$

For an arbitrary divisor $G = \sum m_P P$ of $\mathcal{X}$, we denote by $\nu_P(G)$ the coefficient $m_P$ of $P$. Then

$$G = \sum \nu_P(G) P.$$

For such a divisor $G$ we form the vector space

$$L(G) = \{ x \in F^\ell(\mathcal{X})\setminus\{0\} : \text{div}(x) + G \geq 0 \} \cup \{0\}. $$

Then $L(G)$ is a finite-dimensional vector space over $F^\ell$, and we denote its dimension by $l(G)$. By the Riemann-Roch theorem (see [103, 116]), we have

$$l(G) \geq \deg(G) + 1 - g, \quad (2.5)$$

and equality holds if $\deg(G) \geq 2g - 1$.

Now we are ready to describe the construction.
Let $T$ be a subset of $\mathbb{F}_\ell(X)$, i.e., $T$ is a set of $\mathbb{F}_\ell$-rational points of $X$. Let $D$ be a positive divisor with $T \cap \text{Supp}(D) = \emptyset$. Choose an $\mathbb{F}_\ell$-rational point $R$ in $T$ and put $G = D - R$. Then $\deg(G) = \deg(D) - 1$, $L(G) \subseteq L(D)$ and $\mathbb{F}_\ell \cap L(G) = \{0\}$. Moreover we have

$$L(D) = \mathbb{F}_\ell \oplus L(G) = \{\alpha + f | f \in L(G)\}.$$ 

Each element $(P, \alpha) \in T \times \mathbb{F}_\ell$ can be associated with a map $h_{(P, \alpha)}$ from $L(G)$ to $\mathbb{F}_\ell$ defined by

$$h_{(P, \alpha)}(f) = f(P) + \alpha.$$ 

**Lemma 2.4** Let $\mathcal{H} = \{h_{(P, \alpha)} | (P, \alpha) \in T \times \mathbb{F}_\ell\}$. If $\deg(D) \geq 2g + 1$, then the cardinality of $\mathcal{H}$ is equal to $\ell |T|$.

**Proof.** It is sufficient to prove that $\{h_{(P, \alpha)} | (P, \alpha) \in T \times \mathbb{F}_\ell\}$ are pairwise distinct.

Assume that $h_{(P, \alpha)} = h_{(Q, \beta)}$ for $(P, \alpha)$ and $(Q, \beta)$ in $T \times \mathbb{F}_\ell$, i.e.,

$$h_{(P, \alpha)}(f) = h_{(Q, \beta)}(f)$$  \hspace{1cm} (2.6)

for all $f \in L(G)$. In particular,

$$\alpha = h_{(P, \alpha)}(0) = h_{(Q, \beta)}(0) = \beta.$$  \hspace{1cm} (2.7)

It follows that,

$$f(P) = f(Q)$$

for all $f \in L(G)$. This yields that

$$e(P) = e(Q)$$  \hspace{1cm} (2.8)

for all $e \in L(D)$ since $L(D) = \mathbb{F}_\ell \oplus L(G)$.

Suppose that $P$ is different from $Q$. As $\deg(D - P) \geq \deg(D - P - Q) \geq 2g - 1$, we obtain by the Riemann-Roch theorem

$$l(D - P) = \deg(D) - g, \quad l(D - P - Q) = \deg(D) - g - 1.$$ 

By the above results on dimensions, we can choose a function $u$ from the set $L(D - P) - L(D - P - Q)$. Then it is clear that $u(P) = 0$ and $u(Q) \neq 0$. This contradicts $e(P) = e(Q)$. Hence $P = Q$. The proof is complete.
Theorem 2.5 Let $X$ be an algebraic curve and $T$ a set of $\mathbb{F}_\ell$-rational points on $X$. Suppose that $D$ is a positive divisor with $\text{deg}(D) \geq 2g + 1$ and $T \cap \text{Supp}(D) = \emptyset$. Then there exists an $\epsilon - \text{ASU} \ (N ; k, \ell)$ hash family with

$$N = \ell |T|, \quad k = \ell^{\text{deg}(D)-1} = \ell^{\deg(D) - g}, \quad \epsilon = \frac{\deg(D)}{|T|}.$$  

Proof. Let $R \in T$ be an $\mathbb{F}_\ell$-rational points on $X$ and put $G = D - R$. Define

$$A := L(G), \quad B := \mathbb{F}_\ell$$

and

$$\mathcal{H} := \{ h_{(P, \alpha)} | (P, \alpha) \in T \times \mathbb{F}_\ell \}.$$

It is easy to verify that for any element $a \in A = L(G)$ and $b \in B = \mathbb{F}_\ell$, there exist exactly $|T| = N/\ell$ pairs $(P, \alpha) \in T \times \mathbb{F}_\ell$ such that

$$h_{(P, \alpha)}(a) = a(P) + \alpha = b,$$

i.e., there exist exactly $N/\ell$ functions $h_{(P, \alpha)} \in \mathcal{H}$ such that $h_{(P, \alpha)}(a) = b$.

Now let $a_1, a_2$ be two distinct elements of $A$ and $b_1, b_2$ two elements of $B$. We consider

$$m := \max_{\begin{subarray}{c} a_1 \neq a_2 \in A \\ b_1, b_2 \in B \end{subarray}} \left| \left\{ h_{(P, \alpha)} \in \mathcal{H} | h_{(P, \alpha)}(a_1) = b_1; h_{(P, \alpha)}(a_2) = b_2 \right\} \right|$$

$$= \max_{\begin{subarray}{c} a_1 \neq a_2 \in A \\ b_1, b_2 \in B \end{subarray}} \left| \left\{ (P, \alpha) \in T \times \mathbb{F}_\ell | a_1(P) + \alpha = b_1; a_2(P) + \alpha = b_2 \right\} \right|$$

$$= \max_{\begin{subarray}{c} a_1 \neq a_2 \in A \\ b_1, b_2 \in B \end{subarray}} \left| \left\{ (P, \alpha) \in T \times \mathbb{F}_\ell | (a_1 - a_2 - b_1 + b_2)(P) = 0; a_2(P) + \alpha = b_2 \right\} \right|.$$  

As $a_1 - a_2 \in L(G) - \{0\}$ and $b_1 - b_2 \in \mathbb{F}_\ell$, we know that $0 \neq a_1 - a_2 - b_1 + b_2 \in L(D)$. Thus there are at most $\deg(D)$ distinct zeros of $a_1 - a_2 - b_1 + b_2$ in $T$. Since $\alpha$ is uniquely determined by $P$ from the equality $a_2(P) + \alpha = b_2$, we have that at most $\deg(D)$ pairs $(P, \alpha) \in T \times \mathbb{F}_\ell$ satisfy

$$(a_1 - a_2 - b_1 + b_2)(P) = 0 \quad \text{and} \quad a_2(P) + \alpha = b_2,$$

i.e.,

$$m \leq \deg(D) = \frac{\deg(D) N}{|T|}.$$  

Hence we can take $\epsilon = \deg(D)/|T|$. This completes the proof.

The above theorem gives a construction of $\epsilon$-ASU hash families based on general algebraic curves over finite fields. In the examples below, we apply the theorem to some special curves to obtain $\epsilon$-ASU families with nice parameters.
Example 2.1 Consider the projective line $\mathcal{X}$. Then $g = g(\mathcal{X}) = 0$.

(a) Let $d$ be an integer between 1 and $\ell$, and $P$ an $\mathbb{F}_\ell$-rational point of $\mathcal{X}$. Put

$$D = dP, \quad T = \mathcal{X}(\mathbb{F}_\ell) - \{P\}.$$ 

Then $\deg(D) = d \geq 2g + 1$, $|T| = \ell$ and $T \cap \text{Supp}(D) = \emptyset$. By Theorem 4.1, we obtain an $\epsilon$-ASU $(N; k, \ell)$ hash family with

$$N = \ell^2, \quad k = \ell^d, \quad \epsilon = \frac{d}{\ell}.$$ 

The $\epsilon$-ASU hash family with the above parameters can also be found in [17].

(b) Let $d$ be an integer between 2 and $\ell$. Put $T = \mathcal{X}(\mathbb{F}_\ell)$. As there always exists an irreducible polynomial of degree $d$ over $\mathbb{F}_\ell$, we can find a positive divisor $D$ such that $\deg(D) = d$ and $T \cap \text{Supp}(D) = \emptyset$. Then $\deg(D) = d \geq 2g + 1$, $|T| = \ell + 1$. By Theorem 2.5, we obtain an $\epsilon$-ASU $(N; k, \ell)$ hash family with

$$N = \ell(\ell + 1), \quad k = \ell^d, \quad \epsilon = \frac{d}{\ell + 1}.$$ 

Example 2.2 Consider an elliptic curve $\mathcal{X}$ with $\ell + \lfloor 2\sqrt{\ell} \rfloor$ $\mathbb{F}_\ell$-rational points (such an elliptic curves exists), where $\lfloor \cdot \rfloor$ denotes the integral part of a real number. Let $d$ be an integer between 3 and $\ell + \lfloor 2\sqrt{\ell} \rfloor - 1$, and $P$ an $\mathbb{F}_\ell$-rational point of $\mathcal{X}$. Put

$$D = dP, \quad T = \mathcal{X}(\mathbb{F}_\ell) - \{P\}.$$ 

Then $\deg(D) = d \geq 2g + 1$, $|T| = \ell + \lfloor 2\sqrt{\ell} \rfloor - 1$ and $T \cap \text{Supp}(D) = \emptyset$. By Theorem 2.5 we obtain an $\epsilon$-ASU $(N; k, \ell)$ hash family with

$$N = \ell(\ell + \lfloor 2\sqrt{\ell} \rfloor - 1), \quad k = \ell^d - 1, \quad \epsilon = \frac{d}{\ell + \lfloor 2\sqrt{\ell} \rfloor - 1}.$$ 

Example 2.3 Let $\ell$ be a square and put $r = \sqrt{\ell}$. Consider the Hermitian curve $\mathcal{X}$ defined by

$$y^r + y = x^{r+1}.$$ 

Then the number of $\mathbb{F}_\ell$-rational points of $\mathcal{X}$ is equal to $r^3 + 1 = \ell\sqrt{\ell} + 1$ and the genus of $\mathcal{X}$ is $g = \sqrt{\ell}(\sqrt{\ell} - 1)/2$. Choose an $\mathbb{F}_\ell$-rational point $P$ and put $D = dP$ for an integer $d$ between $2g+1 = \ell - \sqrt{\ell} + 1$ and $r^3 = \ell\sqrt{\ell}$. Define $T$ to be the set $\mathcal{X}(\mathbb{F}_\ell) - \{P\}$. Then $\deg(D) = d \geq 2g + 1$, $|T| = \ell\sqrt{\ell}$ and $T \cap \text{Supp}(D) = \emptyset$. By Theorem 4.1, we obtain an $\epsilon$-ASU $(N; k, \ell)$ hash family with

$$N = \ell^2\sqrt{\ell}, \quad k = \ell^{d - \sqrt{\ell}/2}, \quad \epsilon = \frac{d}{\ell\sqrt{\ell}}.$$
Example 2.4 Let \( \ell \) be a square and put \( r = \sqrt{\ell} \). Consider a sequence of algebraic curves \( X_m \) given in [53] as follows. Let \( X_1 \) be the projective line with the function field \( F_\ell(X) = F_\ell(x_1) \). Let \( X_m \) be obtained by adjoining a new equation:

\[
x_m^r + x_m = \frac{x_{m-1}^r - 1}{x_m^{r-1} + 1},
\]
for all \( m \geq 2 \). Then the number of \( F_\ell \)-rational points of \( X_m \) is more than \( (r^2 - r)r^{m-1} \), and the genus \( g_m \) of \( X_m \) is less than \( r^m \) for all \( m \geq 1 \). Choose an integer \( c \) between 2 and \( \sqrt{\ell} - 1 \) (\( c \) is independent of \( m \)), an \( F_\ell \)-rational point \( P_m \) of \( X_m \) and put \( D_m = c\ell^{m/2}P_m \). Let \( T_m \) be a subset of \( X_m(F_\ell) - \{ P_m \} \) with

\[
|T_m| = (r^2 - r)r^{m-1} = \ell^{m/2}(\sqrt{\ell} - 1).
\]

By Theorem 4.1, we obtain a sequence of \( \epsilon \)-ASU \((N_m; k_m, \ell)\) hash families with

\[
N_m = \ell^{m/2}(\ell \sqrt{\ell} - \ell), \quad k_m = \ell^{c\ell^{m/2}-g_m} \geq \ell^{(c-1)\ell^{m/2}}, \quad \epsilon = \frac{\deg(D_m)}{|T_m|} = \frac{c}{\sqrt{\ell} - 1}.
\]

2.3 Comparisons with previous constructions

From Lemma 2.2, one can easily translate the \( \epsilon \)-ASU hash family from Theorem 2.5 and Examples 2.1–2.4 to A-codes in a straightforward manner. In particular, we phrase the construction of Example 2.4 in terms of A-code as follows.

Corollary 2.6 The construction in Example 2.4 results in an A-code \( C = (S, E, T) \) with the following parameters

\[
|S| = \ell^{(c-1)\ell^{m/2}}, \quad |E| = \ell^{m/2}(\ell \sqrt{\ell} - \ell), \quad |T| = \ell,
\]

and with deception probabilities

\[
P_T = \frac{1}{\ell}, \quad P_S = \frac{c}{\sqrt{\ell} - 1},
\]

where \( \ell \) is a prime power and a square, \( c \) and \( m \) are integers satisfying \( 2 \leq c \leq \sqrt{\ell} - 1 \) and \( m \geq 1 \).

In the sequel, we make some comparisons between various parameters of the above A-code with those in [16] and [59], which are known to have the best performance among all the previously known construction.
2.3. Comparisons with previous constructions

Compare with Helleseth and Johansson's construction In [59] Helleseth and Johansson applied exponential sums over finite fields to construct an A-code $C_0 = (S_0, E_0, T_0)$ with the parameters,

$$|S_0| = q^{r(D - \lfloor D/p \rfloor)}, \quad |E_0| = q^{r+1}, \quad |T_0| = q$$

and with deception probabilities

$$\hat{P}_I = \frac{1}{q}, \quad \hat{P}_S = \frac{1}{q} + \frac{D - 1}{\sqrt{q^r}}$$

where $p, q$ are prime powers and $q = p^e$ for some integer $e$, and $D$ and $r$ are integers satisfying $0 \leq D \leq \sqrt{q^r}$ and $r \geq 1$.

To compare the performance of our construction and that of Helleseth and Johansson [59], we will choose two A-codes $C = (S, E, T)$ and $C_0 = (S_0, E_0, T_0)$ such that the sizes of keys, the lengths of authenticators, and the success probabilities in impersonation and substitution attacks for them are the same. We then compare their sizes of source states.

Suppose $q = \ell$ and $2r = m + 1$. Let $D = \lceil 2(cq^{r/2} - 1 + 1) \rceil = \lceil 2(c\ell^{(m-3)/4} + 1) \rceil$. Since $2 \leq c \leq \sqrt{\ell}$, it follows that $D \leq 2(\sqrt{q^r}^{(m-3)/4} + 1) = 2(q^{(m-1)/4} + 1) \leq \sqrt{q^m}$, as desired by Helleseth and Johansson construction. From these parameters, we have

- $P_I = \hat{P}_I = 1/\ell$;
- $P_S = c/(\sqrt{\ell} - 1) < 2c/\sqrt{\ell} + 1/\ell \leq 1/q + (D - 1)/(\sqrt{q^r}^{(m-3)/4}) = \hat{P}_S$;
- $|T| = |T_0| = \ell$; and
- $|E| = \ell^{m/2}(\ell - \ell) \leq \ell^{(m+3)/2} = q^{(m+3)/2 + 1} = q^{r+1} = |E_0|$.

On the other hand,

$$|S_0| = q^{r(D - \lfloor D/p \rfloor)} \leq q^{D(m+1)/2} \leq \ell^{((m+1)/2)2(c\ell^{(m-3)/4} + 1)} \leq \ell^{c(m+1)(\ell^{(m-3)/4} + 1)}.$$  

It follows that

$$\frac{\log_\ell |S|}{\log_\ell |S_0|} \geq \frac{(c - 1)\ell^{m \over 2}}{c(m + 1)(\ell^{m - 3 \over 4} + 1)}.$$

For given $c$, we know $\log_\ell |S|/\log_\ell |S_0|$ is at least $O(\ell^{(m+2)/4}/m)$. As $m$ increases, the size of source states of A-code $C$ significantly exceeds that of $C_0$. 
2.3. Comparisons with previous constructions

Next, we show that if we allow the size of authenticators in our construction to be slightly larger while keeping the size of key and deception probabilities for two codes to be the same, then the increase in the size of the source size in our construction is even more significant. To this end, we first want to have the probabilities of substitution attacks (i.e. the overall security level) for two A-codes are the same. Let $\ell = q^4$ and $c = [(q^2 - 1)/q + (D - 1)/\sqrt{q^2(q^2 - 1)}] + 1$. Then $P_S \approx \hat{P}_S$.

Next, let $r = 2m + 5$. Then we have

$$|\mathcal{E}| = q^{2m}(q^6 - q^4) \leq q^{2m+5} = q^{r+1} = |\mathcal{E}_0|.$$  

It yields that the key size of $C_0$ is at least as large as that of $C$.

On the other hand,

$$|\mathcal{S}| \geq \ell^{(c-1)\ell^m/2}$$

$$= q^{\ell^{(c-1)q^2m}}$$

$$\geq q^{((q^2-1)/q + (D-1)/\sqrt{q^2(q^2-1)})q^{2m}}$$

$$\geq q^{(D-1)q^{2m-r/2}}$$

$$= q^{(D-1)q^{m-5/2}}.$$  

Without loss of generality, we may further assume that $D \geq p$. It follows that $(D-1) \geq (D - \lfloor D/p \rfloor)$, and we have $|S_0| = q^{r(D-\lfloor D/p \rfloor)} \leq q^{2m+5}(D-1)$. As we have seen $|\mathcal{S}| \geq q^{(D-1)q^{m-5/2}}$. It follows that the size of the source of A-code $C_0$ linearly increases with $m$ while the size of source for $C$ exponentially increases with $m$. By choosing large $m$ the size of source in $C$ significantly exceeds that of $C_0$. We note that the authenticator length of $C$ is four times that of $C_0$, and the success probability in impersonation has reduced from $1/q$ (of $C_0$) to $1/q^4$ (of $C$).

**Compare with Bierbrauer’s construction** In [16] Bierbrauer applied the composition method to geometric code and obtained an A-code $C' = (S', \mathcal{E}', T')$ with the parameters

$$|S'| = q^{s(1+s-t)}, \quad |\mathcal{E}'| = q^{2s+t}, \quad |T'| = q^t$$

and with deception probabilities,

$$P'_I = \frac{1}{q^t}, \quad P'_S = \frac{2}{q^t},$$

where $q$ is a prime power, and $s \geq t$ are natural numbers.

Now consider A-codes $C$ and $C'$. We again want the success probabilities of substitution attacks for the two codes to be the same. Let $\ell = q^{2t}$ and $c = 2$. Then we
have $P_s = 2/(q^t - 1)$, and so $P_s \approx P'_s$. Next, let $m = [(2s - t)/t]$. Then

$$|E| = \ell^{m/2}(\ell \sqrt{\ell} - \ell) = q^{2t(m/2)}q^{2t}(q^t - 1) \leq q^{2s-t}q^{2t} = q^{2s+t} = |E'_0|.$$

On the other hand,

$$|S| \geq \ell^{(c-1)\ell^{m/2}} \geq q^{2tq^{t((2s-t)/t-1)}} \geq q^{2tq^{2s-t}}.$$

It follows that

$$\frac{\log_q |S|}{\log_q |S'|} = \frac{2tq^{(s-t)}q^{(s-t)}}{s(1+q^{s-t})} \geq \frac{t(1+q^{s-t})q^{s-t}}{s(1+q^{s-t})} = \frac{tq^{s-t}}{s}.$$

If we fix $t$, which determines the deception probability of substitution attack, then the ratio for the sizes of source states of the two A-codes is at least $O(q^s/s)$. It follows that as $s$ increases the size of source in our construction can be significantly increased.

### 2.4 Conclusion

We have given constructions of universal hash families and authentication codes from general algebraic curves over finite fields. The construction is different from the previous ones in two respects: firstly, it is the first direct construction based on algebraic curves without using “composition method”; secondly, the construction yields new classes of authentication codes and universal hash families which are substantially better than those previously known.
Multireceiver authentication codes (MRA-codes) are introduced by Desmedt, Frankel and Yung (DFY) [41] as an extension of Simmons’ model of unconditionally secure authentication. In an MRA-code, a sender wants to authenticate a message for a group of receivers such that each receiver can verify authenticity of the received message. The receivers are not trusted and may try to construct fraudulent messages on behalf of the transmitter. If the fraudulent message is acceptable by even one receiver the attackers have succeeded. This is a useful extension of traditional authentication codes and has numerous applications. For example a director wants to give instructions to employees in an organisation such that each employee is able to verify authenticity of the message. Providing such service using digital signature implies that security is based on unproven assumptions and the attackers have finite amount of computational resources. We will be only concerned with the unconditionally secure model, that is, there is no computational assumptions or limitations on the attackers’ resources.

A multireceiver A-code can be trivially constructed using traditional A-codes: the sender shares a common key with each receiver and to send an authenticated message, constructs $n$ codewords, one for each receiver, concatenates them and broadcasts the result. Now each receiver can verify its own codeword and so authenticate the message. In this construction collaboration of even $n - 1$ receivers does not enable them to construct a message that is acceptable by the $n^{th}$ receiver simply because the $n$ codewords are independently constructed. If we assume that the size of the malicious groups cannot be too large, for example the biggest number of collaborators is $w - 1$ ($w < n$), then we can expect to save on the size of the key and the length of the codeword because the tags can have dependencies. This is the basic motivation of studying MRA-codes that are more efficient than the trivial one described above. DFY gave two constructions for $(w, n)$ MRA-codes based on polynomials over finite fields and finite geometries. DFY description of MRA-code is basically an operational description of the system: that is the way the system works. Kurosawa and Obana (KO)
[71] studied \((w, n)\) MRA-code, again using the operational description of these codes, derived combinatorial lower bounds on the probability of success in impersonation and substitution attacks, and characterised Cartesian MRA-codes that satisfy the bounds with equality. They showed that DFY polynomial construction is in fact an optimal (smallest sizes of transmitter and receiver keys) construction.

In this chapter we start by giving a formal definition of an MRA-code and use it to derive information theoretic bounds on the probability of success in impersonation and substitution attack against a single receiver for a general MRA-code. These bounds are used to obtain lower bounds on the number of keys of the transmitter and receivers, and also lower bound on the length of the transmitted codeword in terms of deception probability of the system. This is followed by a review of the known constructions of MRA-codes, pointing out their shortcomings and giving constructions that alleviate these shortcomings.

### 3.1 The Model

An extension of the conventional model of authentication, proposed by Desmedt, Frankel and Yung (DFY) [41], is when there are multiple receivers. The system works as follows. First the key distribution centre (KDC) securely distributes secret keys to the transmitter and each receiver. Next the transmitter broadcasts a message to all the receivers who can individually verify authenticity of the message using their secret key. There are malicious groups of receivers who use their secret keys and all the previous communications in the system to construct fraudulent messages. They succeed in their attack even if a single receiver accepts the message as being authentic.

An MRA-System has three phases:

1. **Key distribution**: The KDC (key distribution centre) privately transmits the key information to the sender and each receiver (the sender can also be the KDC).

2. **Broadcast**: For a source state, the sender generates the authenticated message using his/her key and broadcasts the authenticated message.

3. **Verification**: Each receiver can verify the authenticity of the broadcast message.

Denote by \(X_1 \times \cdots \times X_n\) the direct product of sets \(X_1, \ldots, X_n\), and by \(p_i\) be the projection mapping of \(X_1 \times \cdots \times X_n\) on \(X_i\). That is, \(p_i : X_1 \times \cdots \times X_n \rightarrow X_i\), be defined by \(p_i(x_1, x_2, \ldots, x_n) = x_i\). Let \(g_1 : X_1 \rightarrow Y_1\) and \(g_2 : X_2 \rightarrow Y_2\) be two mappings. We
denote the direct product of $g_1$ and $g_2$ by $g_1 \times g_2$, where $g_1 \times g_2 : X_1 \times X_2 \rightarrow Y_1 \times Y_2$ is defined by $(g_1 \times g_2)(x_1, x_2) = (g_1(x_1), g_2(x_2))$. The identity mapping on a set $X$ is denoted by $1_X$.

**Definition 3.1** Let $C = (S, M, \mathcal{E}, f)$ and $C_i = (S, M_i, \mathcal{E}_i, f_i)$, $i = 1, 2, \ldots, n$, be authentication codes. We call $(C; C_1, C_2, \ldots, C_n)$ a multireceiver authentication code (MRA-code) if there exist two mappings $\tau : \mathcal{E} \rightarrow \mathcal{E}_1 \times \cdots \times \mathcal{E}_n$ and $\pi : M \rightarrow M_1 \times \cdots \times M_n$ such that for any $(s, e) \in S \times \mathcal{E}$ and any $1 \leq i \leq n$, the following identity holds:

$$p_i(f(s, e)) = f_i((1_S \times p_i\tau)(s, e)).$$

Let $\tau_i = p_i\tau$ and $\pi_i = p_i\pi$. Then for each $(s, e) \in S \times \mathcal{E}$, we have

$$\pi_if(s, e) = f_i(1_S \times \tau_i)(s, e).$$

We assume that for each $i$ the mappings $\tau_i : \mathcal{E} \rightarrow \mathcal{E}_i$ and $\pi_i : M \rightarrow M_i$ are surjective. We also assume that for each code $C_i$ the probability distribution on the source states of $C_i$ is the same with that in the $A$-code $C$, and the probability distribution on $\mathcal{E}_i$ is derived from that of $\mathcal{E}$ and the mapping $\tau_i$.

Let $T$ denote the sender and $R_1, \ldots, R_n$ denote the $n$ receivers. In order to authenticate a message, the sender and receivers follow the following protocol.

1. The KDC (or the sender) randomly chooses a key $e \in \mathcal{E}$ and privately transmits $e$ to $T$ and $e_i = \pi_i(e)$ to the receiver $R_i$, $1 \leq i \leq n$.

2. If $T$ wants to send a source state $s \in S$ to all the receivers, $T$ computes $m = f(s, e) \in M$ and broadcasts it to all receivers.

3. Receiver $R_i$ checks whether a source state $s$ such that $f_i(s, e_i) = \pi_i(m)$ exists. If such an $s$ exists, the message $m$ is accepted as authentic. Otherwise $m$ is rejected.

We adopt the Kerckhoff’s principle that everything in the system except the actual keys of the sender and receivers is public. This includes the probability distribution of the source states and the sender’s keys. From Definition 3.1 we know that the probability distribution of the sender’s key induces a probability distribution on each receiver’s key.

Attackers could be outsiders who do not have access to any key information, or insiders who have some key information. We only need to consider the latter group as it is at least as powerful as the former. We consider the systems that protect against
3.2 Information-Theoretic Bounds

the coalition of groups of receivers (up to a maximum size) and study impersonation and substitution attacks.

Assume there are \( n \) receivers \( R_1, \ldots, R_n \). Let \( L = \{i_1, \ldots, i_t\} \subseteq \{1, \ldots, n\} \), \( \mathcal{E}_i = \mathcal{E}_{i_1} \times \cdots \times \mathcal{E}_{i_t} \) and \( R_L = \{R_{i_1}, \ldots, R_{i_t}\} \). We consider the attack from \( R_L \) on a receiver \( R_i \), where \( i \notin L \).

**Impersonation attack:** \( R_L \), after receiving their secret keys, send a message \( m \) to \( R_i \). \( R_L \) is successful if \( m \) is accepted by \( R_i \) as authentic. We denote by \( P_I[i, L] \) the success probability of \( R_L \) in performing an impersonation attack on \( R_i \). This can be expressed as

\[
P_I[i, L] = \max_{e_L \in \mathcal{E}_L} \max_{m \in M} P(m \text{ is accepted by } R_i \mid e_L)
\]

where \( i \notin L \).

**Substitution attack:** \( R_L \), after observing a message \( m \) that is transmitted by the sender, replace \( m \) with another message \( m' \). \( R_L \) is successful if \( m' \) is accepted by \( R_i \) as authentic. We denote by \( P_S[i, L] \), the success probability of \( R_L \) in performing a substitution attack on \( R_i \). We have,

\[
P_S[i, L] = \max_{e_L \in \mathcal{E}_L} \max_{m \in M} \max_{m' \in M} P(R_i \text{ accepts } m' \mid m, e_L)
\]

3.2 Information-Theoretic Bounds

In the following Theorem 3.1 we derive bounds on deception probability of a group of insiders who have access to part of the key information. The bounds generalise Simmons' [100] and Brickell's bounds [26].

**Theorem 3.1** Let \( P_I[i, L] \) and \( P_S[i, L] \) be defined as in equation (3.1) and (3.2). Assume that \( M' \times M = \{(m', m); m' \neq m, m', m \in M\} \), then

1. \( P_I[i, L] \geq 2^{-I(M'; E_i \mid E_L)} \).

2. \( P_S[i, L] \geq 2^{-I(M'; E_i \mid M, E_L)} \).

**Proof.** 1. We define an impersonation characteristic function \( \chi_I \) on \( M \times \mathcal{E}_i \times \mathcal{E}_L \) by,

\[
\chi_I(m, e_i, e_L) = \begin{cases} 
1 & \text{if } m \text{ is a valid for } e \in \mathcal{E} \text{ in } C \\
& \text{such that } \tau_i(e) = e_i \text{ and } \tau_L(e) = e_L; \\
0 & \text{otherwise.}
\end{cases}
\]
3.2. Information-Theoretic Bounds

From the definition of the impersonation attack we can express $P_{I}[i, I]$ as,

$$\begin{align*}
P_{I}[i, L] &= \max_{m \in M} P(\pi_i(m) \text{ is valid in } C_i | e_L \in \mathcal{E}_L) \\
&= \max_{m \in M} \sum_{e_i \in \mathcal{E}_i} \chi_I(m, e_i, e_L) P(e_i | e_L)).
\end{align*}$$

For given $L \subseteq \{1, \ldots, n\}$ and $i \notin L$, let $P(m, e_i, e_L)$ be the joint probability distribution induced by the system. If $\chi_I(m, e_i, e_L) = 0$ then $P(m, e_i, e_L) = 0$. Indeed, if $P(m, e_i, e_L) \neq 0$ then $m$ is a valid message for $e$ with $\tau_i(e) = e_i$ and $\tau_L(e) = e_L$, which contradicts the definition of $\chi_I(m, e_i, e_L)$.

$$\begin{align*}
I(M; E_i | E_L) &= E_{P(m,e_i,e_L)} \frac{P(M,E_i | E_L)}{P(M | E_L)P(E_i | E_L)} \\
&= \sum_{m \in M, e_i \in \mathcal{E}_i, e_L \in \mathcal{E}_L} P(m, e_i, e_L) \log \frac{P(m,e_i | e_L)}{P(m | e_L)P(e_i | e_L)} \\
&= \sum_{m \in M, e_i \in \mathcal{E}_i, e_L \in \mathcal{E}_L} P(m, e_i, e_L) \log \frac{P(e_i | m, e_L)P(e_i)}{P(e_i | e_L)} \\
&= \sum_{m \in M, e_L \in \mathcal{E}_L} P(m, e_L) \left( \sum_{e_i \in \mathcal{E}_i} P(e_i | m, e_L) \log \frac{P(e_i | m, e_L)}{P(e_i | e_L)} \right).
\end{align*}$$

For each pair $(m, e_L)$ with $P(m, e_L) \neq 0$, if $\chi_I(m, e_i, e_L) = 0$ then $P(e_i | m, e_L) = 0$. In this case, $P(e_i | m, e_L) \log \frac{P(e_i | m, e_L)}{P(e_i | e_L)} = 0$. It follows that the summation taking over $\mathcal{E}_i$ in the above identity is restricted to all $e_i$ for which $\chi_I(m, e_i, e_L) = 1$. Thus we have,

$$\begin{align*}
I(M; E_i | E_L) &= \sum_{m \in M, e_L \in \mathcal{E}_L} P(m, e_L) \left( \sum_{e_i \in \mathcal{E}_i} P(e_i | m, e_L) \chi_I(m, e_i, e_L) \right) \\
&\quad \cdot \log \frac{P(e_i | m, e_L)\chi_I(m, e_i, e_L)}{P(e_i | e_L)\chi_I(m, e_i, e_L)}.
\end{align*}$$

By log-sum inequality, we have,

$$\begin{align*}
I(M; E_i | E_L) \geq \sum_{m \in M, e_L \in \mathcal{E}_L} P(m, e_L) \left( \sum_{e_i \in \mathcal{E}_i} P(e_i | m, e_L) \chi_I(m, e_i, e_L) \right) \\
&\quad \cdot \log \frac{\sum_{e_i \in \mathcal{E}_i} P(e_i | m, e_L)\chi_I(m, e_i, e_L)}{\sum_{e_i \in \mathcal{E}_i} P(e_i | e_L)\chi_I(m, e_i, e_L)}.
\end{align*}$$
3.2. Information-Theoretic Bounds

For each pair \((m, e_L)\), as we have noted before, if \(P(m, e_L) \neq 0\) and \(\chi_I(m, e_i, e_L) = 0\), then \(P(e_i|m, e_L) = 0\). It follows that

\[
\sum_{e_i \in E_i} P(e_i|m, e_L) \chi_I(m, e_i, e_L) = 1,
\]

and

\[
\sum_{e_i \in E_i} P(e_i|e_L) \chi_I(m, e_i, e_L) = P(\pi_i(m) \text{ is valid in } C_i|e_L).
\]

We obtain,

\[
I(M; E_i|E_L) \geq - \sum_{m \in \mathcal{M}, e_L \in E_L} P(m, e_L) \log P(\pi_i(m) \text{ is valid in } C_i|e_L)
\]

\[
= - \sum_{e_L \in E_L} P(e_L) \sum_{m \in \mathcal{M}} P(m|e_L) \log P(\pi_i(m) \text{ is valid in } C_i|e_L).
\]

Since,

\[
P_{i}[i, L] \geq \sum_{e_L \in E_L} P(e_L) \left[ \max_{m \in \mathcal{M}} P(\pi_i(m) \text{ is valid in } C_i|e_L) \right]
\]

\[
\geq \sum_{e_L \in E_L} P(e_L) \left[ \sum_{m \in \mathcal{M}} P(m|e_L) P(\pi_i(m) \text{ is valid in } C_i|e_L) \right],
\]

by Jensen inequality, it follows that,

\[
\log P_{i}[i, L] \geq \sum_{e_L \in E_L} P(e_L) \sum_{m \in \mathcal{M}} P(m|e_L) \log P(\pi_i(m) \text{ is valid in } C_i|e_L)
\]

\[
\geq -I(M; E_i|E_L).
\]

Therefore, \(P_{i}[i, L] \geq 2^{-I(M; E_i|E_L)}\).

2. In the substitution attack \(R_L\) receives the key information from the sender, observe a message \(m\) that is transmitted by \(T\) and construct a fraudulent message \(m\). \(R_L\) succeed if \(m\) is accepted by \(R_i\) as authentic. We denote by \(P_{S}[i, L]\) the successful probability that \(R_L\) perform substitution attack on \(R_i\). We have,

\[
P_S[i, L] = \max_{e_L \in E_L} \max_{m \in \mathcal{M}} \max_{m' \neq m \in \mathcal{M}} P(\pi_i(m) \text{ is valid in } C_i|m, e_L)
\]

Now we define a substitution characteristic function \(\chi_S(m', m, e_i, e_L)\) by

\[
\chi_S(m', m, e_i, e_L) = \begin{cases} 
1 & \text{if } \chi_I(m', e_i, e_L) = 1 \text{ and } \chi_I(m, e_i, e_L) = 1, \text{ and } m' \neq m, \\
0 & \text{otherwise}.
\end{cases}
\]
We introduce a random variable $M'$ which only takes values when

$$\chi_S(m', m, e_i, e_L) = 1$$

It follows that there is a joint probability distribution $P(m', m, e_i, e_L)$ such that $P(m, e_i, e_L)$ is the probability distribution given in the system and such that if $\chi_S(m', m, e_i, e_L) = 0$ and $P(m, e_i, e_L) \neq 0$ then $P(m', e_i, e_L) = 0$.

$$I(M'; E_i|M, E_L)$$

$$= \sum_{m' \in M', m \in M} P(m', m, e_i, e_L) \log \frac{P(m', e_i|m, e_L)}{P(m'|m, e_L)P(e_i|m, e_L)}$$

$$= \sum_{m' \in M', m \in M} P(m', m, e_L)P(e_i|m', m, e_L) \log \frac{P(m'|m, e_L)P(e_i|m', m, e_L)}{P(m'|m, e_L)P(e_i|m, e_L)}$$

$$= \sum_{m' \in M', m \in M} P(m', m, e_L) \sum_{e_i \in E_i} P(e_i|m', m, e_L) \log \frac{P(m'|m, e_L)P(e_i|m', m, e_L)}{P(m'|m, e_L)P(e_i|m, e_L)}$$

If $P(m', m, e_L) \neq 0$ then $\chi_S(m', m, e_i, e_L) = 0$ implies $P(e_i|m', m, e_L) = 0$, and so

$$P(e_i|m', m, e_L) \log \frac{P(e_i|m', m, e_L)}{P(e_i|m, e_L)} = 0.$$
Thus the summation taken over \( \mathcal{E}_i \) in the above identity is restricted to all \( e_i \) for which \( \chi_{S}(m',m,e_i,e_L) = 1 \). By log-sum inequality, we have

\[
I(M'; E_i | M, E_L) = \sum_{m' \in M, m \in M} P(m', m, e_L) \sum_{e_i \in \mathcal{E}_i} P(e_i | m', m, e_L) \chi_{S}(m', m, e_i, e_L)
\]

\[
\geq \sum_{m' \in M, m \in M} P(m', m, e_L) \sum_{e_i \in \mathcal{E}_i} P(e_i | m', m, e_L) \chi_{S}(m', m, e_i, e_L)
\]

\[
\cdot \left( \log \frac{\sum_{e_i \in \mathcal{E}_i} P(e_i | m', m, e_L) \chi_{S}(m', m, e_i, e_L)}{\sum_{e_i \in \mathcal{E}_i} P(e_i | m, m, e_L) \chi_{S}(m', m, e_i, e_L)} \right)
\]

Again, if \( P(m', m, e_L) \neq 0 \) and \( \chi_{S}(m', m, e_i, e_L) = 0 \) then \( P(e_i | m', m, e_L) = 0 \). It follows that,

\[
\sum_{e_i \in \mathcal{E}_i} P(e_i | m', m, e_L) \chi_{S}(m', m, e_i, e_L) = 1,
\]

and,

\[
\sum_{e_i \in \mathcal{E}_i} P(e_i | m, m, e_L) \chi_{S}(m', m, e_i, e_L) = P(\pi_i(m') \text{ is valid in } C_i|m, e_L).
\]

So we have,

\[
I(M'; E_i | M, E_L) \geq \sum_{m' \in M, m \in M, e_L \in \mathcal{E}_L} P(m', m, e_L) \log P(\pi_i(m') \text{ is valid in } C_i|m, e_L)
\]

\[
= \sum_{m \in M, e_L \in \mathcal{E}_L} P(m, e_L) \sum_{m' \in M'} P(m'|e_L, m) \log P(\pi_i(m') \text{ is valid in } C_i|m, e_L)
\]

Since,

\[
P_{S}[i, L]
\]

\[
\geq \sum_{e_L \in \mathcal{E}_L} P(e_L) \sum_{m \in M} P(m | e_L) \sum_{m' \in M'} P(m'|m, e_L) P(\pi_i(m') \text{ is valid in } C_i|m, e_L)
\]

\[
\geq \sum_{e_L \in \mathcal{E}_L, m \in M} P(e_L, m) \sum_{m' \in M'} P(m'|m, e_L) P(\pi_i(m') \text{ is valid in } C_i|m, e_L).
\]
By Jensen's inequality, it follows that,

\[
\log P_S[i, L] 
\geq \sum_{e_L, m \in M} P(e_L, m) \sum_{m' \in M'} P(m' | m, e_L) \log P(\pi_i(m')) \text{ is valid in } C_i | m, e_L
\]

\[
\geq -I(M'; E_i | M, E_L).
\]

We obtain

\[P_S[i, L] \geq 2^{-I(M'; E_i | M, E_L)}.\]

**Corollary 3.2**

\[P_S[i, L] \geq 2^{-H(E_i | M, E_L)}.\]

**Proof.** The corollary follows from Theorem 3.1 by noting that

\[I(M'; E_i | M, E_L) = H(E_i | M, E_L) - H(E_i | M', M, E_L).\]

A \((w, n)\) MRA-code is an MRA-code in which there are \(n\) receivers such that no subset of \(w - 1\) receivers can construct a fraudulent codeword acceptable by another receiver. We note that in this definition, the only requirement is that the chance of success of the attackers is less than one but it is possible that some coalition of attackers can have a better chance of success than an outsider.

A \((w, n)\) MRA-code is perfect against impersonation attack if the chance of success of any group of up to \(w - 1\) receivers in an impersonation attack is the same as an outsider. Similarly, a \((w, n)\) MRA-code is perfect against substitution attack if the chance of success for any group of up to \(w - 1\) receivers in a substitution attack is the same as an outsider.

**Lemma 3.3** A sufficient condition for a \((w, n)\) MRA-code to be perfect against impersonation attack is that \(P(e_i | e_L) = P(e_i)\) for all \(w\)-subsets \(L \cup \{i\}, i \notin L\) of \(\{1, \ldots, n\}\).

**Proof.** Consider the A-code \(C_i = (S, M_i, E_i)\). We define an authentication function \(\chi(m_i, e_i)\) on \(M_i \times S_i\) as,

\[\chi_S(m_i, e_i) = \begin{cases} 1 & \text{if } m_i \text{ is authentic for the key } e_i \\ 0 & \text{otherwise.} \end{cases}\]
We have \( P(\pi_i(m) \text{ is valid in } C_i) = \sum_{e_i \in e_i \chi(\pi_i(m), e_i)P(e_i).} \) Because of the definition of \( \chi_I(m, e_i, e_L) \), we know that for any \( e_L \) such that \( \tau_L(e) = e_L \) and \( \tau_i = e_i \), we have \( \chi(\pi_i(m), e_i) = \chi_I(m, e_i, e_L) \). Thus, it follows that,

\[
P_I[i, L] = \max_{m \in M} \{ P(m \text{ is accepted by } R_i | e_L) 
= \max_{m \in M} \sum_{e_i \in e_i \chi_I(m, e_i, e_L)P(e_i | e_L) 
= \max_{m \in M} \sum_{e_i \in e_i \chi(\pi_i(m), e_i)P(e_i | e_L) 
= P_I[i].
\]

In the above lemma, \( P_I[i] \) is the success probability of an outsider in impersonation attack and is given by,

\[
P_I[i] = \max_{m \in M} P(R_i \text{ accepts } m) = \max_{m \in M} P(\pi_i(m) \text{ is valid in } C_i).
\]

It should also be noted that a \((w, n)\) MRA-code which is perfect against impersonation attack is not necessarily perfect against substitution attack.

### 3.3 Combinatorial Bounds

Let \((C; C_1, \ldots, C_n)\) be an MRA-code. Define \( P_I \) and \( P_S \) as follows.

\[
P_I = \max_{L \cup \{i\}} \{ P_I[i, L] \}
\]
\[
P_S = \max_{L \cup \{i\}} \{ P_S[i, L] \}
\]

where maximum is taken over all possible \( w \)-subsets \( L \cup \{i\} \) \((i \notin L)\) of \( \{1, 2, \ldots, n\} \). In other words, \( P_I \) and \( P_S \) are the best chances of a group of \( w - 1 \) receivers to succeed in impersonation and substitution attacks against a single receiver, respectively. We define the deception probability of a \((w, n)\) MRA-system as \( P_D = \max\{P_I, P_S\} \).

**Theorem 3.4** Let \((C; C_1, \ldots, C_n)\) be a \((w, n)\) MRA-code. Assume that

\[
P_D \leq 1/q,
\]

and there is a uniform probability distribution on the source \( S \). Then,

(i) \(|e_i| \geq q^2\), for each \( i \in \{1, \ldots, n\}\);

(ii) \(|e| \geq q^{2w}\);

(iii) \(|M| \geq q^w|S|\).
The bounds are tight and there exists a system that satisfies the bounds with equality.

Proof. (i) For each \((w - 1)\)-subset \(L\) of \(\{1, \ldots, n\}\) and any \(i \in \{1, \ldots, n\}\) where \(i \notin L\), using Theorem 3.1 and Corollary 3.2 we have,

\[
\left(\frac{1}{q}\right)^2 \geq \frac{P_D^2}{q} \geq P_I[i, L]P_S[i, L] \geq 2^{-(I(M;E_i|E_L)+H(E_i|E_L,M))} = 2^{-H(E_i|E_L)} \\
\geq 2^{-H(E_i)} \geq 2^{-\log |E_i|} = \frac{1}{|E_i|}.
\]

It follows that \(|E_i| \geq q^2\).

(ii) Assume that \(L_i = \{1, \ldots, i - 1, i + 1, \ldots, w\}, i = 1, \ldots, w\). We have,

\[
\left(\frac{1}{q}\right)^{2w} \geq \prod_{i=1}^{w} P_I[i, L_i]P_S[i, L_i] \geq 2^{\sum_{i=1}^{w} H(E_i|E_{L_i})} \\
\geq 2^{-\sum_{i=1}^{w} H(E_i|E_{1, \ldots, E_{i-1}})} = 2^{-H(E_1, \ldots, E_w)} \\
\geq 2^{-H(E)} \geq 2^{-\log |E|} = \frac{1}{|E|}.
\]

Therefore, \(|E| \geq q^{2w}\).

(iii) Since \(\tau : E \rightarrow E_1 \times \cdots \times E_n\) induces a mapping from \(E\) to \(E_1 \times \cdots \times E_w\), we have \(I(M; E) \geq I(M; E_1, \ldots, E_w)\). It follows that

\[
2^{-I(M; E)} \leq 2^{-I(M; E_1, \ldots, E_w)} = 2^{-\sum_{i=1}^{w} I(M; E_i|E_1, \ldots, E_w)} \\
= 2^{-\sum_{i=1}^{w} I(M; E_i|E_{1, \ldots, E_{i-1}})} \\
= \prod_{i=1}^{w} 2^{-I(M; E_i|E_{1, \ldots, E_{i-1}})} \leq \prod_{i=1}^{w} P_I[i, Q_i],
\]

where \(Q_i = \{1, \ldots, i - 1\}\). Since for each \(1 \leq i \leq w\), we have,

\[
P_I[i, Q_i] \leq P_I[i, L_i] \leq \frac{1}{q},
\]

it follows that,

\[
2^{-I(M; E)} = 2^{-\left(H(M)-H(M|E)\right)} = 2^{-H(M)}2^{H(M|E)} \leq \left(\frac{1}{q}\right)^w.
\]

Since \(S\) is assumed to be uniformly distributed, we know that

\[
H(M|E) = H(S) = \log |S|.
\]

Hence \(|M| = 2^{\log |M|} \geq 2^{H(M)} \geq q^w|S|\), which proves (iii).

The bounds are tight as it is easy to verify that they are satisfied by the polynomial construction by Desmedt, Frankel and Yung (described in section 3.5.1), in which we have, \(P_D = 1/q\), \(|E_i| = q^2\), for all \(1 \leq i \leq n\), \(|E| = q^{2w}\) and \(|M| = q^w|S|\) and so the lower bounds are satisfied with equality.
3.4 Comparison with Kurosawa et al Bounds

Kurosawa and Obana [71] formalise \((w, n)\) MRA-codes as follows. Let \(\mathcal{E}_1, \mathcal{E}_2, \ldots, \mathcal{E}_n\) denote the set of decoding rules of receivers \(R_1, \ldots, R_n\), respectively, and let \(\mathcal{S}\) and \(\mathcal{M}\) denote the set of source states and senders codewords, respectively.

**Definition 3.2** ([71]) We say that \((\mathcal{S}, \mathcal{M}, \mathcal{E}_1, \ldots, \mathcal{E}_n)\) is a \((w, n)\) multireceiver A-code if for all \((\mathcal{E}_i, \ldots, \mathcal{E}_i)\) and all \((e_1, \ldots, e_w)\),

\[
P(E_{iw} = e_w | E_{i1} = e_1, \ldots, E_{iw-1} = e_{w-1}) = P(E_{iw} = e_w).
\]

Probabilities of success in impersonation and substitution attacks, \(P_I\) and \(P_S\), for \((w, n)\) MRA-codes are then defined as

\[
P_I = \max_{R_i} \max_m P(R_i \text{ accepts } m)
\]

\[
P_S = \sum_m P(m) \max_{R_i} \max_{m'} P(R_i \text{ accepts } m' | R_i \text{ accepts } m)
\]

where maximum is taken over \(m'\) such that the source state of \(m'\) is different from that of \(m\). With these definitions, they derived the following bounds. Assume \(\ell = |\mathcal{M}|/|\mathcal{S}|\).

**Theorem 3.5** (Theorem 9 [71]) In a \((w, n)\) MRA-code, \(P_I \geq 1/\sqrt{\ell}\). Equality holds if and only if \(P(R_{i_1}, \ldots, R_{iw} \text{ accept } m) = 1/\ell\) and \(P(R_j \text{ accepts } m) = 1/\sqrt{\ell}\) for any \(m\) and any \(R_j\).

**Theorem 3.6** (Theorem 10 [71]) In a \((w, n)\) MRA-code without secrecy, if \(P_I = 1/\sqrt{\ell}\), then \(P_S \geq 1/\sqrt{\ell}\). Equality holds if and only if

\[
P(R_{i_1}, \ldots, R_{ik} \text{ accept } m' | R_{i_1}, \ldots, R_{ik} \text{ accept } m) = 1/\ell
\]

\[
P(R_j \text{ accepts } m' | R_j \text{ accepts } m) = 1/\sqrt{\ell}
\]

for all \(R_j, \forall m\) and \(\forall m'\) such that the source state of \(m\) is different from that of \(m'\).

**Theorem 3.7** (Theorem 11 [71]) In a \((w, n)\) MRA-code without secrecy, if \(P_I = P_S = 1/\sqrt{\ell}\), then \(|\mathcal{E}_j| \geq (\sqrt{\ell})^2\) for all \(j\). If equality holds, then each rule of \(\mathcal{E}_j\) is used with equal probability.

Kurosawa and Obana characterised Cartesian MRA-codes that satisfy \(P_I = P_S = 1/\sqrt{\ell}\) and observed that DFY polynomial construction is in fact an optimal construction and has the least number of keys for the transmitter and the receivers, and the smallest size of the authenticator.
Definition 3.2 does not specify the relationship between the encoding functions of the transmitter and the receivers and only requires the independence of receivers’ keys for any set of \( w \) receivers. This independence, as shown in Lemma 3.3, is sufficient to ensure that the success probability of \( w - 1 \) receivers against another receiver in impersonation attack by any is the same as that of an (outside) opponent. We give a general definition of MRA-codes in terms of commutative mappings, and for \((w,n)\) MRA-codes only require the success probability of attackers in impersonation and/or substitution attacks to be less than one. However we do allow coalition of insiders to have higher chance of success compared to an outsider. KO’s definition of \((w,n)\) MRA-codes corresponds to our definition of \((w,n)\) MRA-codes that are perfect against impersonation attack (see Lemma 3.3).

In the following we give a comparison between bounds obtained in Theorem 3.4 and the bounds derived by Kurosawa and Obana in [71]. Let \( \ell = \frac{|M|}{|S|} \).

1. In [71] the first part of Theorem 9 proves that,

\[
P_I \geq \frac{1}{\sqrt{\ell}}.
\]

We show that our Theorem 3.4 (iii) implies that,

\[
P_D = \max\{P_I, P_S\} \geq \frac{1}{\sqrt{\ell}}.
\]

This is because assuming \( P_D = \max\{P_I, P_S\} = 1/q \) and using Theorem 3.4 (iii), we have

\[
|M| \geq q^w |S| \implies P_D = \frac{1}{q} \geq \sqrt{\frac{|S|}{|M|}} = \frac{1}{\sqrt{\ell}}.
\]

Our result applies to general MRA-codes. Kurosawa et al result is stronger as \( P_S \geq 1/q \) implies \( P_D \geq 1/q \), but only applies to MRA-codes that are perfect against impersonation attack.

2. Theorem 10 and 11 in [71] in fact prove the following result (see also the introduction in [71]).

**Theorem 3.8** (KO [71]) For a \((w,n)\) MRA-code without secrecy, if \( P_I = P_S = \frac{1}{\sqrt{\ell}} \), then \( |E| \geq \ell^2 \) and \( |E_i| \geq (\sqrt{\ell})^2 \) for all \( 1 \leq i \leq n \).

This result can be also obtained from Theorem 3.4. Indeed, since \( P_I = P_S = \frac{1}{\sqrt{\ell}} \), we have \( P_D = \frac{1}{q} = \frac{1}{\sqrt{\ell}} \) where \( q = \sqrt{\ell} \). By our Theorem 3.4 (i) and (ii) it follows that,

\[
|E_i| \geq q^2 = (\sqrt{\ell})^2.
\]
3.5 Constructions

47

\[ |\mathcal{E}| \geq q^{2w} = (\sqrt{\ell})^{2w} = (\ell)^2, \]

proving the desired result.

This result applies to all \((w,n)\) MRA-codes and does not require the code to be perfect against impersonation attack. It also applies to codes with secrecy.

3. The second part of Theorems 9, 10 and 11 in [71] do not have any counterpart in this paper.

3.5 Constructions

3.5.1 DFY Polynomial Construction

In [41], Desmedt, Frankel and Yung gave two constructions for MRA-codes: one is based on polynomials and the other based on finite geometries. We briefly review DFY's polynomial construction because generalisations of this scheme will be discussed later. Details of the geometric construction can be found in [41].

Assume there is a sender \(T\), and \(n\) receivers \(R_1, \ldots, R_n\). DFY polynomial scheme works as follows. The key for \(T\) consists of two random polynomials \(P_0(x)\) and \(P_1(x)\), each of degree at most \(w - 1\), with coefficients in \(GF(q)\), where \(q \geq \max\{\lvert S\rvert, n\}\). The key for \(R_i\) consists of \(P_0(i)\) and \(P_1(i)\). To authenticate a source state \(s \in GF(q)\), \(T\) broadcasts \((s, A(x))\) where \(A(x) = P_0(x) + sP_1(x)\). \(R_i\) accepts \((s, A(x))\) as authentic if \(A(i) = P_0(i) + sP_1(i)\). It is proved [41] that the construction results in a MRA-code with \(P = 1/q\) and the following parameters:

\[
\begin{align*}
|S| &= q, \\
|\mathcal{E}_i| &= q^2, \forall i \in \{1, \ldots, n\}, \\
|\mathcal{E}| &= 2^{2w}, \text{ and } |\mathcal{M}| &= q^w |S|,
\end{align*}
\]

and so the bounds in Theorem 3.4 can be achieved with equality.

The trivial construction for MRA-codes (as noted before), requires the sender to store many key bits and produces a long tag for the authenticated message. DFY scheme significantly reduces the size of the key storage and the length of the authentication tag. However the order of the field \(GF(q)\) must be chosen bigger than the size of the source and the number of the receivers. In fact \(q\), which can be thought of as the security parameter of the system, \((P_I = P_S = 1/q)\), determines the size of the key storage and the length of the authentication tag. This makes the construction very restrictive because although it is acceptable to have the key storage, and length of the tag, a function of the security parameter of the system, but having the number of receivers and the size of the source bounded by it, is not reasonable. In particular
when the size of the source or the number of the receivers are very large, \( P_l \) and \( P_s \) will be unnecessarily small and the key storage of the sender and the receivers, together with the length of the authentication tag will become prohibitively large.

In practice, we might be satisfied with deception probabilities higher than \( 1/q \), but have limitation on key storage or communication bandwidth. So it is desirable to look for constructions that accommodate this situation. In Section 3.5.2 we will give such a construction.

### 3.5.2 A Construction based on \((n, m, w)\)-Cover-Free Family

In this section we present a general construction for \((w, n)\) MRA-codes by combining an arbitrary A-code with an \((n, m, w)\)-cover-free Family.

**Definition 3.3** Let \( X = \{x_1, \ldots, x_m\} \) and \( B = \{B_1, \ldots, B_n\} \) be a family of subsets of \( X \). We call \((X, F)\) an \((n, m, w)\) Cover-Free Family (CFF) if \( B_i \not\subseteq B_{i_1} \cup \cdots \cup B_{i_{w-1}} \) for all \( B_{i_0}, B_{i_1}, \ldots, B_{i_{w-1}} \in B \), where \( B_{i_j} \neq B_{i_k} \) if \( j \neq k \).

CFFs were introduced by Erdös et al in [46] and [47], and also implicitly studied by Fujii, Kachen and Kurosawa in [52] in connection with MRA-codes. An \((n, w, 2)\) CFF is exactly a Sperner family. A trivial CFF is the family consisting of single element subsets, in which case \( n = m \). Non-trivial CFFs are those with \( n > m \). A good CFF is one that for given \( m \) and \( w, n \) is large. Finding good CFFs with the largest possible \( n \) is believed to be a hard combinatorial problem [45]. Constructions of CFFs rely on various areas of mathematics such as finite geometries, design theory and probability theory [46, 47].

Assume that \((X, B)\) is an \((n, m, w)\) CFF and \((S, T, E, f)\) is an A-code without secrecy. We construct a \((w, n)\) MRA-code as follows.

1. **Key Distribution:** The KDC randomly chooses an \( m \)-tuple of keys \((e_1, \ldots, e_m) \in E^m\), and privately sends \((e_1, \ldots, e_m)\) to the sender \( T \) and \( e_i \) to every receiver \( R_j \) for all \( j \) with \( x_i \in B_j, 1 \leq i \leq m \).

2. **Broadcast:** For a source state \( s \in S \), the sender calculates \( a_i = f(s, e_i) \) for all \( 1 \leq i \leq m \) and broadcast \((s, a_1, \ldots, a_m)\).

3. **Verification:** Since the receiver \( R_i \) holds the keys \( \{e_j \mid x_j \in B_i\} \), \( R_i \) accepts \((s, a_1, \ldots, a_m)\) as authentic if for all \( j \) satisfying \( x_j \in B_i, a_j = f(s, e_j) \).
Assume that the probabilities of impersonation and substitution attacks for the underlying A-code, $C$, is $P_I$ and $P_S$, respectively, and let

$$\alpha = \min\{ |B_{i_0} \setminus B_{i_1} \cup \cdots \cup B_{i_{w-1}}|; \text{ for all } B_{i_0}, \ldots, B_{i_{w-1}} \in \mathcal{B} \}.$$

**Theorem 3.9** The above scheme is a $(w, n)$ MRA-code and the probabilities of impersonation and substitution attacks are $(P_I)^\alpha$ and $(P_S)^\alpha$, respectively.

The proof of the theorem is straightforward. In this scheme the sender is required to store $m[\log |\mathcal{E}|]$ bits, and the receiver $R_i$ to store $|B_i|[\log |\mathcal{E}|]$ bits. The authentication tag is of size $m[\log |\mathcal{T}|]$.

In [52], Fujii, Kachen and Kurosawa gave a definition of broadcast authentication which can be seen as a special case of DFY definition of MRA systems. Fujii et al also gave a construction for their broadcast authentication system which is a special case of the above construction, when the cover-free family has constant block size; that is $|B_i| = c, i = 1, \ldots, n$.

An important property of this construction is that it allows a complex system, such as a $(w, n)$ MRA-code, to be constructed from two simpler ones, an A-code and a cover-free family, such that the security of the former can be described in terms of the properties and parameters of the latter. Another advantage of this construction is its flexibility in choosing system parameters. That is $w$ and $n$ are determined by the cover-free family while $P_I$ and $P_S$ are determined by the A-code and the cover-free family and so it is possible to fix $w$ and $n$ but change the A-code to obtain MRA-codes that provide the required protection. The following examples compare this construction with DFY polynomial scheme.

**Example 3.1** Assume that the size of the source state is only one bit (for example, yes and no) and we need a $(2, 70)$ MRA-code with the probabilities of impersonation and substitution attacks not greater than $1/2$. Using DFY polynomial scheme we need a finite field $GF(q)$ with $q \geq 70$; it follows that $[\log q] \geq 7$, and so the sender must store at least 28 bits and each receiver must store at least 14 bits. The length of the authentication tag is at least 14 bits, and the probabilities of impersonation and substitution attacks are $(\frac{1}{2})^7$. Now we use our construction. It is easy to see that the Sperner family consisting of all 4-subsets of a set of 8 elements gives a $(70, 8, 2)$ CFF. We define the underlying A-code $C = (S, T, \mathcal{E}, f)$ as follows. Let $S = T = GF(2), \mathcal{E} = GF(2)^2$, and $f : S \times \mathcal{E} \rightarrow T$ be given by $f(s, (e, e')) = e + se'$. Then $C$ is an A-code with $P_I = P_S = \frac{1}{2}$. Applying our scheme, the sender and each receiver need to store
only 16 bits and 8 bits, respectively. The length of authentication tag is of 8 bits and the probabilities of impersonation and substitution attacks are both 1/2.

**Example 3.2** Assume that the size of the source state is very large, for example $2^{20}$ bits (i.e. $|S| = 2^{20}$). A direct computation shows that the DFY polynomial scheme for (2,70) MRA-code requires that the sender and each receiver to store $2^{22}$ and $2^{21}$ bits, respectively. The length of authentication tag is $2^{21}$ bits while the probability of success in impersonation and substitution attacks is not greater than $1/2^{220}$. In many applications the deception probability of around $1/2^{20}$ is acceptable. Consider an A-code that is constructed from a universal hashing family (see [107]) with the following parameter: $2^{20}$ bits of source state, 445 bits of authentication key, 20 bits of authentication tag and the probability of impersonation and substitution attacks no greater than $1/2^{19}$. Combining with the (70,8,2) CFF, our construction results in a (2,70) MRA-code in which the key storages for the sender and each receiver are 3560 bits and 1780 bits, respectively. The length of the authentication tag is 160 bits and the deception probability is bounded by $1/2^{19}$.

We note that this construction is only suitable for the case when the number of malicious receivers compared with the total number of the receivers is not very large. This is due to the following result.

**Lemma 3.10** ([47]) In a non-trivial $(n,m,w)$ CFF, $\frac{w(w-1)}{2} \leq n$.

In [45], using probabilistic methods the authors proved that $(n, O(\log n), w)$ CFFs exist for small $w$. Finally, we point out that MRA-code constructions that are based on CFFs are not perfect against impersonation or substitution attacks.

### 3.5.3 MRA-codes for Multiple Message Transmissions

In the basic model of MRA-codes, security analysis is for a single message transmission (only impersonation and substitution attacks are considered) and for a second message no protection is guaranteed. To provide protection for multiple message transmission one possibility is to use a new key after each broadcasted message. This is very inefficient both in terms of going through a key distribution phase after each message and the amount of key information required for each message. In the following we propose systems that use a single key distribution phase for multiple message transmission and compared with using a new key for each message require less key information per communicated message.
3.5. Constructions

Generalised DFY scheme for multiple messages
Assume messages are all distinct and \( t < |\mathcal{S}| \). The scheme consists of the following steps:

1. **Key distribution:** KDC randomly generates \( t+1 \) polynomials \( P_0(x), P_1(x), \ldots, P_t(x) \) of degree at most \( w - 1 \) and chooses \( n \) distinct elements \( x_1, x_2, \ldots, x_n \) of \( GF(q) \). KDC makes \( x_i \)'s public and privately sends \( (P_0(x), P_1(x), \ldots, P_t(x)) \) to the sender \( T \), and \( (P_0(x_i), P_1(x_i), \ldots, P_t(x_i)) \) to the receiver \( R_i \).

2. **Broadcast:** For a source state \( s \), \( T \) computes \( A_s(x) = P_0(x) + sP_1(x) + \cdots + s^tP_t(x) \) and broadcasts \( (s, A_s(x)) \).

3. **Verification:** \( R_i \) accepts \( (s, A_s(x)) \) as authentic if

\[
A_s(x_i) = P_0(x_i) + sP_1(x_i) + \cdots + s^tP_t(x_i).
\]

The above scheme is a multi-receiver authentication code in which each key can be used to authenticate up to \( t \) messages. To prove the security of the scheme, we consider the scenario where for a given key \( (P_0(x), P_1(x), \ldots, P_t(x)) \), \( t \) source states \( s_1, s_2, \ldots, s_t \) have been authenticated and there are \( w - 1 \) receivers who want to construct a fraudulent codeword that is acceptable by one of the other receivers. Without loss of generality, we may assume that the malicious receivers are \( R_1, R_2, \ldots, R_{w-1} \).

Let \( P_i(x) = a_{i0} + a_{i1}x + \cdots + a_{iw}x^w \), \( 0 \leq i \leq t \). Since \( s_1, s_2, \ldots, s_t \) have been sent, \( A_{s_1}(x), A_{s_2}(x), \ldots, A_{s_t}(x) \) given by

\[
A_{s_j}(x) = b_{j0} + b_{j1}x + \cdots + b_{jw-1}x^{w-1}, \text{ for all } 1 \leq j \leq t,
\]

are publicly known, and the \( w - 1 \) receivers \( R_1, R_2, \ldots, R_{w-1} \) know their keys

\[
(P_0(x_1), P_1(x_1), \ldots, P_t(x_1)), \ldots, (P_0(x_{w-1}), P_1(x_{w-1}), \ldots, P_t(x_{w-1}))
\]

It follows that the malicious receivers know the following two matrix equations,

\[
\begin{bmatrix}
    a_{00} & a_{01} & \cdots & a_{0t} \\
a_{01} & a_{11} & \cdots & a_{1t} \\
    \vdots & \vdots & \ddots & \vdots \\
a_{0w-1} & a_{1w-1} & \cdots & a_{tw-1}
\end{bmatrix}
\begin{bmatrix}
    1 & 1 & \cdots & 1 \\
s_1 & s_2 & \cdots & s_t \\
    \vdots & \vdots & \ddots & \vdots \\
s_{10} & s_{20} & \cdots & s_{t0}
\end{bmatrix}
= 
\begin{bmatrix}
    b_{10} & b_{11} & \cdots & b_{1w-1} \\
b_{20} & b_{21} & \cdots & b_{2w-1} \\
    \vdots & \vdots & \ddots & \vdots \\
b_{t0} & b_{t1} & \cdots & b_{tw-1}
\end{bmatrix}
\]
and,

\[
\begin{bmatrix}
1 & x_1 & \cdots & x_{w-1} \\
1 & x_2 & \cdots & x_{w-1} \\
\vdots & \vdots & \ddots & \vdots \\
1 & x_{w-1} & \cdots & x_{w-1}
\end{bmatrix}
\begin{bmatrix}
a_{00} & a_{10} & \cdots & a_{t0} \\
a_{01} & a_{11} & \cdots & a_{t1} \\
\vdots & \vdots & \ddots & \vdots \\
a_{0w-1} & a_{1w-1} & \cdots & a_{tw-1}
\end{bmatrix}
\begin{bmatrix}
1 \\
x_1 \\
x_2 \\
\vdots \\
x_{w-1}
\end{bmatrix}
= 
\begin{bmatrix}
P_0(x_1) \\
P_0(x_2) \\
\vdots \\
P_0(x_{w-1})
\end{bmatrix}
\begin{bmatrix}
P_t(x_1) \\
P_t(x_2) \\
\vdots \\
P_t(x_{w-1})
\end{bmatrix}
\]

The matrix equations can be rewritten as,

\[
AM_t = B, \tag{3.3}
\]
\[
X_{w-1}A = C, \tag{3.4}
\]

where \(A, M_t, B, X_{w-1}\) and \(C\) denote the corresponding matrices. We first give a lemma, which says that knowing \(M_t, X_{w-1}, B\) and \(C\) cannot determine \(A\). In other words, the matrix satisfying (1) and (2) is not unique.

**Lemma 3.11** There exist \(q\) different matrices \(D\) such that \(DM_t = B\) and \(X_{w-1}D = C\).

**Proof.** It is sufficient to prove that there exist \(q\) different matrices \(D\) such that \(DM_t = 0\) and \(X_{w-1}D = 0\). First, we observe that given an \(n \times m\) matrix \(D_0 = (d_{ij})\), we can associate it with a polynomial in \(x, y\),

\[
F(x, y) = (1, x, \cdots, x^{n-1})D_0
\]

Conversely, a polynomial \(F(x, y)\) can be written in the form (3) for some \(n \times m\) matrix \(D_0\). Now consider the polynomial,

\[
F(x, y) = (x - x_1)(x - x_2)\cdots(x - x_{w-1})(y - s_1)(y - s_2)\cdots(y - s_t).
\]

Let

\[
F(x, y) = (1, x, \cdots, x^{w-1})D
\]

\[
\begin{pmatrix}
1 \\
y \\
\vdots \\
y^t
\end{pmatrix}
\]

52
where $D$ is a $w \times (t + 1)$ matrix and $D \neq 0$. Clearly, $F(x_1, y) = F(x_2, y) = \cdots = F(x_{w-1}, y) = 0$ for all $y$. It follows that,

$$
\begin{bmatrix}
1 & x_1 & \cdots & x_1^{w-1} \\
1 & x_2 & \cdots & x_2^{w-1} \\
\vdots & \vdots & \ddots & \vdots \\
1 & x_{w-1} & \cdots & x_{w-1}^{w-1}
\end{bmatrix} D = 0.
$$

Now, we may choose $(t + 1)$ distinct elements $y_1, y_2, \ldots, y_{t+1}$ in $GF(q)$ such that,

$$
F(x_1, y_i) = F(x_2, y_i) = \cdots = F(x_{w-1}, y_i) = 0, \text{ for all } 1 \leq i \leq t+1.
$$

Thus we have,

$$
\begin{bmatrix}
1 & x_1 & \cdots & x_1^{w-1} \\
1 & x_2 & \cdots & x_2^{w-1} \\
\vdots & \vdots & \ddots & \vdots \\
1 & x_t & \cdots & x_t^{w-1}
\end{bmatrix} D
\begin{bmatrix}
1 & 1 & \cdots & 1 \\
y_1 & y_2 & \cdots & y_{t+1} \\
y_1^t & y_2^t & \cdots & y_{t+1}^t
\end{bmatrix} = 0.
$$

Since

$$
\begin{bmatrix}
1 & 1 & \cdots & 1 \\
y_1 & y_2 & \cdots & y_{t+1} \\
y_1^t & y_2^t & \cdots & y_{t+1}^t
\end{bmatrix}
$$
is a Vandermonde matrix, the desired result follows. Similarly, we have

$$
D
\begin{bmatrix}
1 & 1 & \cdots & 1 \\
s_1 & s_2 & \cdots & s_t \\
s_1^t & s_2^t & \cdots & s_t^t
\end{bmatrix} = 0.
$$

For each $r \in GF(q)$, we also have $(rD)M_t = 0$ and $X_{w-1}(rD) = 0$. Thus there are $q$ different matrices $\{rD | r \in GF(q)\}$ with the desired property. So we complete the proof of the lemma.

**Theorem 3.12** [111] The above scheme is a $(w, n)$ MRA-code in which every key can be used to authenticate up to $t$ messages.

To authenticate $t$ consecutive messages, using basic DFY scheme, $2t$ polynomials are required while in the above scheme we only need $t + 1$ polynomials. So the key storages for the sender and receivers are $(t + 1)w[\log q]$ bits and $(w + 1)[\log q]$ bits, respectively, which is reduced to nearly half of that of the DFY scheme. The length of the authentication tag for both constructions are the same and equal to $tw[\log q]$ bits.
Using Cover-Free Family Construction

To extend the construction of Section 3.5.2 to support multiple messages it is only required to replace the underlying A-code by an A-code that provides protection against spoofing of order $t$, $t > 1$. In an spoofing of order $t$ attack on an A-code, the enemy has access to $t$ authenticated codewords and wants to construct a fraudulent one. An A-code provides perfect protection against spoofing of order $t$ if the enemy's best strategy is randomly selecting one of the remaining codewords. It is straightforward to see that in the construction given in Section 3.5.2, using an A-code that provides protection against spoofing of order $t$ ensures that probability of success in spoofing of order $t$ (which can be defined similar to A-codes) is equal to $(P_t)^n$, where $P_t$ is the probability of success in spoofing of order $t$ for the A-code used in the construction.

By replacing the underlying A-code with a Wegman-Carter type construction [4] one can obtain an MRA-code for multiple authentication using universal hash functions.

3.5.4 A Construction from Error-Correcting Codes

We first give a definition for optimal MRA-codes.

**Definition 3.4** An MRA-code is called optimal if the bounds in Theorem 3.4 are met with equality.

The important property of optimal MRA-codes is that, for given cheating probability, the number of keys for transmitter and receivers is minimum and the tag length for codewords is the smallest.

DFY [41] gave two constructions for MRA-codes: one based on polynomials and the other based on finite geometries. Kurosawa and Obana [71] showed that the polynomial construction is optimal. No other optimal construction was known so far. In this section we use error-correcting codes (E-codes) to construct MRA-codes. First, we present two constructions which can be used to derive an MRA-code from an arbitrary E-code and then show that the constructions result in new optimal MRA-codes.

A linear $[n, k]$ code $C$ over $GF(q)$ is a linear $k$ dimensional subspace of $GF(q)^n$. The dual code is denoted by $C^\perp$ and is the collection of vectors that are orthogonal to all vectors of $C$. Minimum distances of $C$ and $C^\perp$ are denoted by $d$ and $d'$, respectively.

**Construction I**

Let $C$ be a linear $[n, k]$ error-correcting code (E-code) over $GF(q)$ with a generator matrix $G \in (GF(q))^{k \times n}$. We construct an MRA-code with $n$ receivers from $C$ as follows. Assume that $S = GF(q)$ is the set of source states and $G$ is publicly known.
1. **Key distribution**: $T$ randomly chooses $(\alpha, \beta) \in GF(q)^k \times GF(q)^k$. $T$ then calculates the codewords $\alpha G = u = (u_1, \ldots, u_n)$ and $\beta G = v = (v_1, \ldots, v_n)$, and privately transmits $(u_i, v_i)$ to the receiver $R_i$ for each $1 \leq i \leq n$. That constitutes the secret key of $R_i$.

2. **Broadcast**: To authenticate a message $s \in \mathcal{S}$, the sender $T$ computes $\gamma = \alpha + s\beta$ and broadcasts $(s, \gamma)$ to all the receivers.

3. **Verification**: For each $i$, $R_i$ accepts $(s, \gamma)$ as authentic if $y_i = u_i + s v_i$, where $y = (y_1, \ldots, y_n) = \gamma G$.

**Lemma 3.13** In the above construction, let the probability distribution on the source and sender’s key space be uniform. Let $L = \{i_1, \ldots, i_t\} \subseteq \{1, \ldots, n\}$ and $i \not\in L$. Then $P_L[i, L] = P_S[i, L] = \frac{1}{q}$ if and only if there exists a codeword $c = (c_1, \ldots, c_n) \in C$ such that $c_{i_1} = \cdots = c_{i_t} = 0$ and $c_i = 1$.

**Proof.**

**Sufficiency**: Assume that there exists a codeword $c \in C$ satisfying the required property of the theorem. Let $u = \alpha G, v = \beta G$, where $(\alpha, \beta)$ is the key chosen by the sender $T$. Because of the linearity of the E-code, we know that for any $t, t' \in GF(q)$ we have $u + tc, v + t'c \in C$. Since $R_L$ has the key information $((u_i, \ldots, u_{i_t})$ and $(v_i, \ldots, v_{i_t}))$, then for all $t, t' \in GF(q)$, $(u_{i_1}, \ldots, u_{i_t}, u_i + t)$ and $(v_{i_1}, \ldots, v_{i_t}, v_i + t')$ produce all possible keys of $R_{L \cup \{i\}}$. It follows that $R_L$ have no information about $R_i$’s key, and hence $P_L[i, L] = P_S[i, L] = \frac{1}{q}$.

**Necessity**: Assume that there is no codeword $c$ in $C$ satisfying the required property of the theorem. We prove that $(u_i, \ldots, u_{i_t})$ and $(v_i, \ldots, v_{i_t})$ uniquely determine $u_i$ and $v_i$. Clearly, there exist $u_i$ and $v_i$ such that $(u_i, \ldots, u_{i_t}, u_i)$ and $(v_i, \ldots, v_{i_t}, v_i)$ are subcodewords of $C$. We only need to show that such $u_i$ and $v_i$ are unique. Indeed, if there exist two subcodewords $(u_i, \ldots, u_{i_t}, u_i), (u_i, \ldots, u_{i_t}, u'_i)$ in $C$, it follows that $(u_i, \ldots, u_{i_t}, u_i) - (u_i, \ldots, u_{i_t}, u'_i) = (0, \ldots, 0, u_i - u'_i)$ is also a subcodeword in $C$, and so is $(0, \ldots, 0, 1)$, which is a contradiction. In this case we have $P_L[i, L] = P_S[i, L] = 1$, proving the necessity.

**Theorem 3.14** Let $C$ be a linear $[n, k]$ code over $GF(q)$ with $d'$ the minimum distance of its dual code, $C^\perp$. Then Construction I results in a $(w, n)$ MRA-code with $P_L = P_S = 1/q, w = d' - 1 \leq k$, and the following parameters,

$$|\mathcal{S}| = q, |\mathcal{M}| = q^k|\mathcal{S}|, |\mathcal{E}| = q^{2k} \text{ and } |\mathcal{E}_i| = q^2.$$
3.5. Constructions

Proof. We show that the resulting MRA-code is a \((d' - 1, n)\) MRA-code, but not a \((d', n)\) MRA-code. Let \(G\) be a generator matrix of \(C\). Recall [72] that \(C^\perp\) has the minimum distance \(d'\) if and only if every \(d' - 1\) columns of \(G\) are linearly independent and some \(d'\) columns of \(G\) are linearly dependent. For each \(d' - 1\) columns, indexed by \(\{i_1, \ldots, i_{d' - 2}, i\}\), the restriction of \(G\) to these \(d' - 1\) columns results in a \(k \times (d' - 1)\) matrix \(G_{\{i_1, \ldots, i_{d' - 2}, i\}}\). It follows that \(e_i \in GF(q)^{d' - 1}\) can be expressed as a linear combination of \(k\) rows of \(G_{\{i_1, \ldots, i_{d' - 2}, i\}}\), where \(e_i \in GF(q)^{d' - 1}\) is the vector with the \(i\)th entry being 1 and other entries being 0. This implies that there exits a codeword \(c = (c_1, \ldots, c_n) \in C\) such that \(c_1 = \ldots = c_{i_{d' - 2}} = 0\) and \(c_i = 1\). Thus, by Lemma 3.13, we have \(P_I[i, L] = P_S[i, L] = 1/q\) for any \(d' - 1\) subset \(\{i\} \cup L\) of \(\{1, \ldots, n\}\) with \(i \notin L\), and so the MRA-code is a \((d' - 1, n)\) MRA-code with \(P_I = P_S = 1/q\). In a similar manner, we can prove that there exists a \(d'\)-subset \(L \cup \{i\}\) of \(\{1, \ldots, n\}\) such that \(P_I[i, L] = P_S[i, L] = 1\), so the code is not a \((d', n)\) MRA-code.

In general the MRA-code derived from an E-code is not optimal and does not satisfy bounds of Theorem 3.4. In the following we will show that for a well-known class of E-codes the construction results in optimal MRA-codes.

A maximum distance separable (MDS) E-code has maximum possible minimum distance and its parameters satisfy \(d = n - k + 1\). We are only interested in linear MDS codes. An important property of MDS codes is that its dual code is an MDS code too (page 318 in [72]).

This means that for an MDS code \(d' = n - (n - k) + 1 = k\), or \(k = d' - 1\). That is, the resulting \((w, n)\) MRA-code can protect against the largest size set of cheaters. Using this result and theorem 3.14 it is straightforward to prove the following.

Corollary 3.15 If the linear code \(C\) in Construction I is an \([n, k]\) MDS code over \(GF(q)\), then Construction I results in an optimal \((k, n)\) MRA-code with \(P_I = P_S = 1/q\). The MRA-code has the following parameters,

\[|S| = q, \ |\mathcal{M}| = q^k|S|, \ |\mathcal{E}| = q^{2k} \text{ and } |\mathcal{E}_i| = q^2.\]

A special class of MDS codes are Reed-Solomon code with the following generator matrix,

\[
G = \begin{bmatrix}
1 & 1 & \cdots & 1 \\
x_1 & x_2 & \cdots & x_n \\
\vdots & \vdots & \ddots & \vdots \\
x_1^{w-1} & x_2^{w-1} & \cdots & x_n^{w-1}
\end{bmatrix},
\]

where \(x_i's\) are \(n\) distinct elements in \(GF(q)\).
Corollary 3.16 If the linear code $C$ in Construction I is an $[n,k]$ Reed-Solomon code, then Construction I coincides with DFY's construction.

3.5.5 Construction II

Construction I can be seen as a generalisation of DFY's construction. Construction II is based on the properties of the dual code and can be used for large size sources which makes it of practical interest. We first describe the construction and then discuss its properties.

The basic idea is to use vectors of dual code for verification process. The sender's secret key is an $\ell \times w$ matrix $U$ which defines the generator matrix $G = [I_\ell \mid U]$ of a linear code. To authenticate a source state $s \in S$ the sender generates the codeword $c = sG$ and broadcasts it to the receivers. Each receiver $R_i$ has a codeword $d_i$ of the dual code. To verify authenticity of a broadcasted vector $x$, receiver $R_i$ calculates $x \cdot d_i$ (\textit{\cdot} denotes vector inner product) and if it is zero, it accepts the codeword as authentic.

Let $S \subseteq GF(q)^\ell$ denote the set of source states obtained by defining an equivalence relation $\sim$ over $GF(q)^\ell \setminus \{0\}$ as follows: $s \sim s'$ if and only if $s = rs'$ for some $0 \neq r \in GF(q)$. It is easy to verify that this relation is in fact an equivalence relation. We define $S$ as the set of equivalence classes obtained from $\sim$. It follows that $|S| = \frac{q^\ell - 1}{q-1} = q^{\ell-1} = \cdots + q + 1$.

The three phases of Construction II are as follows.

1. Key distribution: The sender $T$ randomly chooses an $\ell \times w$ matrix $G \in GF(q)^{\ell \times w}$ (and so $[I_\ell \mid U]$ is the generator matrix of a linear $[\ell + w,\ell]$ code in its systematic form). Assume that $q \geq n$ (this assumption is not necessary$^1$). $T$ chooses $n$ distinct elements $x_1, \ldots, x_n \in GF(q)$ (these elements are public and are used as the identities of the receivers), and then calculates and secretly transmits $U(1,x_1,\ldots,x_n) = \alpha_i \in GF(q)^{\ell \times 1}$ to $R_i$, which consists of the secret key of $R_i$, $i = 1, \ldots, n$.

2. Broadcast: To authenticate a source state $s = (s_1, \ldots, s_\ell) \in S$, $T$ computes $sU = t = (t_1, \ldots, t_w) \in GF(q)^w$ and broadcasts $(s, t)$.

3. Verification: For each $i$, $R_i$ accepts $(s, t)$ as authentic if $s\alpha_i = t(1,x_i,\ldots,x_i^{w-1})^T$.

Theorem 3.17 Construction II results in a $(w,n)$ multireceiver A-code with $P_I =$

$^1$Instead, the sender may choose a $w \times n$ matrix $M = (M_1, \ldots, M_n)$ over $GF(q)$ such that any $w$ columns of $M$ are linearly independent and the secret key of $R_i$ is $(-U)^T M_i$. 


$P_S = 1/q$. It has the following parameters,

$$|S| = \frac{q^l - 1}{q - 1}, \quad |M| = q^w|S|, \quad |E| = q^{tw} \quad \text{and} \quad |E_i| = q^t.$$

**Proof.** First, we prove that $P_I = P_S = 1/q$. It is sufficient to show that for each $L \subseteq \{1, \ldots, n\}$ with $|L| = w - 1$ and $i \notin L$, $P_I[i, L] = P_S[i, L] = 1/q$. Without loss of generality, assume that $L = \{1, \ldots, w - 1\}$ and $i = w$, and that after the key distribution, $R_L$ hold the keys

$$U(1, x_1, \ldots, x_{w-1})^T = \alpha_1, \quad \ldots, \quad U(1, x_{w-1}, \ldots, x_1)^T = \alpha_{w-1}.$$

Let $F = \{U \in GF(q)^{\ell \times w}, U(1, x_1, \ldots, 1)^T = \alpha_1, \quad \ldots, \quad U(1, x_{w-1}, \ldots, x_1)^T = \alpha_{w-1}\}$. That is, $F$ is the set of possibles authentication keys of the sender $T$ in accordance with the keys of $R_L$. We define a mapping $\phi : F \rightarrow GF(q)^{\ell \times 1}$ by,

$$\phi(U) = U(1, x_w, \ldots, x_1)^T, \quad \forall U \in F.$$

It is straightforward to verify that $\phi$ is one-to-one from $F$ onto $GF(q)^{\ell \times 1}$. This also implies that $R'_w$s key $U(1, x_w, \ldots, x_1)^T = \phi(U)$ is independent of the keys of $R_L$.

In the impersonation attack, $R_L$, generates a codeword $(s, t), s \in S$ and $t \in T = GF(q)^w$, and hopes that it will be accepted by $R_w$ as authentic. It follows,

$$P_I[w, L] = \frac{\max_{(s, t) \in S \times T} \{|U; U \in F \text{ and } sU(1, x_w, \ldots, x_1)^T = t(1, x_w, \ldots, x_1)^T|\}}{|F|} = \frac{q^{\ell-1}}{q^{\ell}} = \frac{1}{q}.$$

In the substitution attack, $R_L$, after seeing a broadcast authenticated codeword $(s, t)$, generates a new codeword $(s', t')$, $s' \neq s$, and hopes that $(s', t')$ will be accepted by $R_w$ as authentic. It follows that,

$$P_S[w, L] = \frac{\max_{s \neq s'} \{|U; U \in F, sU(1, x_w, \ldots, x_1)^T = t(1, x_w, \ldots, x_1)^T, s'U(1, x_w, \ldots, x_1)^T = t'(1, x_w, \ldots, x_1)^T|\}}{|\{U; U \in F, sU(1, x_w, \ldots, x_1)^T = t(1, x_w, \ldots, x_1)^T\}|} = \frac{q^{\ell-2}}{q^{\ell-1}} = \frac{1}{q}.$$

Similarly, we have $P_I[i, L] = P_S[i, L] = 1/q$ for any $w$ subset $\{i\} \cup L$ of $\{1, \ldots, n\}$ with $i \notin L$. Thus we have proved that $P_I = P_S = 1/q$. The proof of the cardinality parameters are obvious.
Corollary 3.18 Let $q \geq n$ be a prime power. There exists a $(w,n)$ multireceiver A-code with the following parameters, 

$$|S| = q + 1, \quad |M| = q^w|S|, \quad |E| = q^{2w}, \quad \text{and} \quad |E_i| = q^2$$

with probability of success in impersonation and substitution attack given by $P_I = \frac{1}{q}$ and $P_S = \frac{1}{q^2}$, respectively.

The corollary follows from the theorem when $\ell = 2$. The resulting MRA-code meets the bounds of Theorem 3.4 and hence is optimal.

It is interesting to note that for $w = n = 1$, the above construction results in a conventional (one-sender to one-receiver) A-code with the following parameters

$$|S| = \frac{q^\ell - 1}{q - 1}, \quad |M| = q|S|, \quad |E| = |E_1| = q^\ell,$$

and the probability of success in impersonation and substitution is given by $P_I = \frac{1}{q}$ and $P_S = \frac{1}{q}$, respectively. Conventional A-codes with these parameters have been constructed from finite geometries. In particular, for $\ell = 2$, the A-code has the same parameters as the A-code of Gilbert, MacWilliams and Sloane [56]. We note that Construction II is more suitable for MRA-codes with large source space. In the DFY construction and Construction I, the order of the field $GF(q)$ determines the lower bound on the success probabilities in impersonation and substitution, and at the same time bounds the size of the source that can be used in the system ($|S| \leq q$). This can result in inefficient constructions for larger sources. For example a source of size $2^{100}$ results in probability of deception lower bounded by $2^{-100}$ which is unnecessarily low. The price paid for this low probability is bigger key sizes which for practical applications is not acceptable. This restriction is removed in Construction II, and by choosing appropriate $\ell$ the size of source can be increased to the required level.
Collaborative and multi-user applications, such as teleconferences and electronic commerce applications, require secure communication among members of a group. Compared to providing confidentiality, ensuring integrity and authenticity of information is much more difficult as in the latter subgroups of participants can participate in a coordinated attack against other group members, while in the former they are passive. It is also worth emphasising that the two goals of confidentiality and authenticity in group communications are independent and authenticated messages might be in plain form, readable by public.

We consider the following scenario. There is a set of users and a Trusted Authority (TA). During initialisation of the system, TA generates keys for all participants and securely gives the keys to them. Later, each user can broadcast a message which is verifiable for its origin and integrity by every other user, individually. We assume that users are not all trusted and may collude to construct a fraudulent message, to be attributed to another user. We assume security is unconditional and does not rely on any computational assumption.

The obvious method of providing protection in the above system is to use a conventional point-to-point authentication system and give a shared key to each pair of users. Now to construct an authenticated message, a user will construct the authenticator for every other user, concatenate it together and append it to the message. This will allow every receiver to individually verify the message, by verifying the authenticator which is constructed using his shared key. Two immediate drawbacks of this system are that it requires a very large key storage, and produces a very long tag for a message resulting in high communication cost. A more serious problem is that the construction does not provide any security. This will become clear in the later discussions.
Multireceiver authentication systems (MRA-codes) [41] can be seen as the first attempt in providing authentication in group communication. However in this model the sender is fixed. In [111] this limitation is removed and the sender can be any group member. The extended model is called \textit{MRA-code with dynamic sender}, or DMRA-codes for short. DMRA-codes capture the essential aspect of authentication problem in groups but the model allows only one sender while in many applications such as dynamic conference key distributions, group members interact with each other and more than one sender exists. Moreover, the only known non-trivial construction [111, 94] is very inflexible and for large size groups, or sources, results in very inefficient constructions with many key bits and long authenticators. In summary, although DMRA-codes do provide a promising starting point for authentication in group communication, for practical applications more general, flexible and efficient models and constructions are required.

Our goal is to have solutions that are efficient both in terms of storage and communication cost. To achieve our goal we propose two new assumptions.

- The largest size of collusion set is $w$.
- There are at most $t$ transmitters (senders).

These are both reasonable assumptions. The first assumption effectively bounds the power of attackers, and the second one is similar to the degree of spoofing in a conventional authentication system but is more complex to protect against as the $t$ messages are from different originators and so a new type of attack, that is changing the originator of a message, is introduced.

This new attack points to the fact that in a DMRA-code with $t$ dynamic senders an authenticated message must carry some information about its origin. The attack works because in general we allow the same message to be sent by more than one sender. This is a realistic requirement in many applications such as a voting system with many participants and only a 'yes' or 'no' answer. In this attack, that we call \textit{directional attack}, an intruder firstly changes the origin information of a message that is already sent by $P_j$, and then resends and attributes it to $P_i$. This could give him a high success chance if $P_i$ and $P_j$ share some key information which is used for generation of authenticators. This observation immediately rules out direct application of schemes that establish a common key among two or more users, including the scheme described above, the construction based on symmetric polynomial [111], or application of KDP and its more general form $(i,j)$-cover-free family [112] for key distribution. It is worth
noting that all such constructions can be immediately used to provide confidentiality in group communication, but exactly because they result in a shared key among two or more participants they cannot be used in group authentication systems.

To include information about the origin in a broadcasted message, a simple approach would be to attach identity information to the message and then authenticate the result. Although this added information could protect against directional attack but will effectively increase the size of the message space, which in the case of a small source and a large number of participants such as the voting system mentioned above, is not acceptable. We will show that this information is theoretically redundant and can be removed in an optimal system. In the rest of this chapter we assume that identity information is appended to the authenticated message.

The contributions of this chapter can be summarised as follows.

1. We formalise the model of DMRA-codes so that it allows more than one sender. This generalises the model of MRA-codes we study in the previous Chapter.

2. We propose a general ‘synthesis’ construction for DMRA-codes by combining a key distribution pattern (KDP) [77] and an A-code, such that the protection of the resulting system can be determined by the protection of the A-code and parameters of the KDP. The construction is especially attractive as it reduces construction of a DMRA-code to the construction of suitable KDPs and A-codes, and so allows direct application of the previously known results in these latter two areas to the construction of DMRA-codes.

3. We consider a DMRA-code with \( t \) senders, \( t \)DMRA-code for short, and give two new constructions for such systems. The first construction is algebraic and uses polynomials in two variable and has a similar flavour to the optimal construction for the single sender case. The second construction is a ‘synthesis’ construction which is combinatorial in nature and combines a perfect hash families and an arbitrary A-code to obtain a \( t \)DMRA-code. The construction provides the generality and flexibility that we noted earlier in the ‘synthesis’ construction of DMRA-codes from KDP and A-codes.

4. To show the applicability of our results, we give an interesting application of DMRA-codes by constructing a secure dynamic conference system. Security means that messages from a conferencee is only readable, and verifiable for its authenticity, by other conference members. We note that a secure conference
key distribution protocol can be immediately used for a conference system that provides confidentiality for conference members, however no assurance about the origin of messages can be given and so the system does not provide any accountability. Our construction is built on an optimal dynamic conference key distribution system proposed in [24], and for large group sizes, as long as the conference size is relatively small, effectively adds authentication without any extra cost (extra key bits).

5. Although our main analysis is for Cartesian A-codes in the context of unconditional security our main constructions are universal. That is they can also be used for A-codes with secrecy and MACs (Message Authentication Codes), resulting in tDMRA-codes in which protection is determined by the security property of the underlying primitive A-system (A-code with or without secrecy, and MAC) and parameters of a combinatorial structure, a KDP or a perfect hash function family. This is a very interesting property that allows a primitive authentication system be used for efficient authentication in large groups.

In section 4.1 we give the model and the definitions. In section 4.2 we extend previous results for single sender case by proving lower bounds on the size of the key and codeword spaces, and give an optimal construction that meets the bounds. We also describe an efficient construction from key distribution pattern. In section 4.3 we consider systems with multiple senders and present two constructions. In section 4.4 we propose a secure dynamic conference system. Computationally secure group authentication systems are discussed in section 4.5 and section 4.6 concludes our results.

4.1 The Model

A \((w, n)\) MRA-codes [41] is an interesting extension of the classical A-systems where a fixed sender can authenticate a single message for a group of \(n\) receivers such that collusions of up to \(w\) receivers cannot construct a fraudulent codeword which is accepted by another receiver. Bounds and construction for MRA-codes are given in 3

An extension of the MRA-code model is when the sender is not fixed and can be any member of the group. We call the system \(\text{MRA-code with dynamic sender}\). There are many applications for such systems. For example, allowing the sender to be dynamic introduces the notion of authenticating with respect to a particular originator. That is, to verify authenticity of a received message a receiver must first assume an originator
4.1. The Model

for the message and then verify the message with respect to that particular originator. Thus a broadcast message in general carries information about its origin, together with its real content. This origin information can be appended to the real content and authenticated, or appended to the authenticated form of the real content. An attacker succeeds by either changing the origin, or the real content and hence the system must provide both origin (entity) authentication and message authentication.

In the model of MRA-code with dynamic senders, there are $n$ users $\mathcal{P} = \{P_1, \ldots, P_n\}$, who want to communicate over a broadcast channel. The channel is subject to spoofing attack: that is a codeword can be inserted into the channel or, a transmitted codeword can be substituted with a fraudulent one. An attack is directed towards a channel, consisting of a pair of users $\{P_i, P_j\}$, $P_i$ as the sender and $P_j$ as the receiver. A spoofer might be an outsider, or a coalition of $w$ insiders. The aim of the spoofer(s) is to construct a codeword that $P_j$ accepts as being sent from $P_i$.

A TA, generates and distributes secret keys for each users. The TA is only active during key distribution phase. The system consists of three phases.

1. **Key Distribution:** The TA randomly chooses a key $e \in \mathcal{E}$ and applies a key distribution algorithm

\[
\tau : \mathcal{E} \rightarrow \mathcal{E}_1 \times \cdots \times \mathcal{E}_n, \quad \tau(e) = (e_1, \ldots, e_n)
\]

to generate a key $e_i$ for user $P_i$, $1 \leq i \leq n$, and secretly sends $e_i$ to $P_i$.

2. **Broadcast:** A user $P_i$ constructs an authenticated message and broadcasts it. For this, $P_i$ uses his/her own authentication algorithm,

\[
\mathcal{U}_i : \mathcal{S} \times \mathcal{E}_i \rightarrow \mathcal{M}_i, \quad \mathcal{U}_i(s, e_i) = m_i,
\]

where $\mathcal{E}_i$ and $\mathcal{M}_i$ are the set of keys and authenticated codeword for $P_i$. The codeword sent by $P_i$ for a source state $s \in \mathcal{S}$ is $(i, \mathcal{U}_i(s, e_i)) = (i, m_i)$.

3. **Verification:** A user $P_j$, $1 \leq j \leq n$, uses his/her verification algorithm $\mathcal{V}_{ji}$ to accept or reject the received codeword. That is, the key $e_j$ determines a set of verification algorithms $\{\mathcal{V}_{ji} ; 1 \leq i \leq n, j \neq i\}$ with

\[
\mathcal{V}_{ji} : \mathcal{M}_i \times \mathcal{E}_j \rightarrow \{0, 1\},
\]

such that $\mathcal{V}_{ji}(m_i, e_j) = 1$ if $P_j$ accepts $m_i$ as an authenticated codeword sent from $P_i$ and $\mathcal{V}_{ji}(m_i, e_j) = 0$ otherwise.
4.1. The Model

We assume that after a key distribution phase, there are at most \( t \) users who broadcast their authenticated messages and the messages all come from a set \( S \) of source states. For the simplicity, we also assume that each sender may only broadcast a single message. We will adopt the Kerckhoff’s principle that details of the system, except the actual keys, are public. We call the system a \((w, n)\) tDMRA-code and represent it by \( C = (S, \mathcal{E}, \{M_i, \mathcal{E}_i\}_{1 \leq i \leq n}) \), or in Cartesian authentication system, by \( C = (S, \mathcal{E}, \{A_i, \mathcal{E}_i\}_{1 \leq i \leq n}) \).

To assess the security, we define the probability of success in various attacks. Let \( B \) and \( A \) be two subsets of \( \{1, \ldots, n\} \) with \( |B| = \beta \leq t \) and \( |A| = \alpha \leq w \). Without loss of generality, let \( B = \{k_1, \ldots, k_\beta\} \) and \( A = \{\ell_1, \ldots, \ell_\alpha\} \), and denote \( P_B = \{P_{k_1}, \ldots, P_{k_\beta}\} \) and \( P_A = \{P_{\ell_1}, \ldots, P_{\ell_\alpha}\} \). Assume that after seeing the authenticated messages \((s_{k_1}, a_{k_1}), \ldots, (s_{k_\beta}, a_{k_\beta})\) broadcast by \( P_{k_1}, \ldots, P_{k_\beta} \), respectively (\( s_{k_1}, \ldots, s_{k_\beta}\) are not necessary distinct), \( P_A \) want to generate a message \((s_i, a_i)\) such that it will be accepted by \( P_j \) as an authenticated message broadcast by \( P_i \). We further assume that \( i, j \notin A \).

Let \( P[A, P_B, P_i, P_j] \) denote the probability of success for malicious users \( P_A \) in constructing a fraudulent message such that \( P_j \) accepts it as authentic and broadcast by \( P_i \), after the broadcast messages from \( P_B \) are seen. We assume the malicious users use their optimal strategy and want to maximise their chance of success. They can choose the message and the channel, that is \( P_i, P_j \), to achieve this goal.

It is easy to see that if \( A \subset A' \), then \( P[A, P_B, P_i, P_j] \leq P[A', P_B, P_i, P_j] \). Thus, without loss of generality, we assume that \( |B| = w \). For each \( 0 \leq k \leq t \), we define

\[
P_{D_k} = \max_{A, B, i, j} P[A, P_B, P_i, P_j]
\]

where the maximum is taken over all possible \( A, B, i, j \) such that \( |A| = w \), \( |B| = k \) and \( i, j \notin A \). We then define the overall probability of deception, denoted by \( P_D \), as

\[
P_D = \max\{P_{D_0}, P_{D_1}, \ldots, P_{D_t}\}.
\]

**Definition 4.1** A MRA-code with dynamic senders \( C = (S, \mathcal{E}, \{A_i, \mathcal{E}_i\}_{1 \leq i \leq n}) \) is called a \((w, n)\) tDMRA-code if \( P_D < 1 \).

\(^1\)There are other possible attacks for the case \( i \in A \) or \( j \in A \). For example, a user claims to have received a message from other user that was never sent, or after broadcasting a message then denies having sent it. To avoid such attack in conventional A-codes, a new participant called an arbiter who in the case of a dispute arbitrates between transmitter and receiver, is introduced. The resulting A-code is called A-code with arbiter, or \( A^2 \)-code. Similar attacks can be considered for tDMRA-codes.
4.2 DMRA-codes with a Single Sender

We start with the simplest \((w, n)\) tDMRA-code in which \(t = 1\) and simply call it \((w, n)\) DMRA-code. This is exactly the same model as MRA-code with dynamic sender introduced in [111]. In section 4.2.1 we give combinatorial lower bounds on the key size for each user, and also the size of the authenticator, and show the tightness of the bounds by giving a construction that meets the bound, and so is optimal. In section 4.2.3 we will give the 'synthesis' construction.

4.2.1 Bounds

Efficiency of a \((w, n)\) DMRA-code \(C = (S, E, \{M_i, E_i\}_{1 \leq i \leq n})\) can be measured in terms of the size of each user’s key space, \(|E_i|\) and the size of the authenticated message space, \(|M_i|\). We do not really need to consider the the size of key space for TA, \(|E|\), as after the key distribution phase TA does not need to remember the key and so can erase his key. The following lower bounds can be used to determine the best performance of a DMRA-code.

**Theorem 4.1** ([94]) In a \((w, n)\) DMRA-code \(C = (S, E, \{M_i, E_i\}_{1 \leq i \leq n})\) with \(P_D \leq 1/q\) and uniform probability distribution on the source \(S\), we have:

(i) \(|E_i| \geq q^{2(w+1)}\), for each \(i \in \{1, 2, \ldots, n\}\),

(ii) \(|M_i| \geq q^{w+1}|S|\), for each \(i \in \{1, 2, \ldots, n\}\).

**Proof.** For each \(i, 1 \leq i \leq n\), \(P_i\) is a possible sender and so the \((w, n)\) MRA-system with dynamic sender induces a \((w, n-1)\) MRA-code, in which the probability of success in impersonation and substitution attacks are both \(1/q\). By applying Theorem 4.1, we obtain the required results. In Section 4.2.2 we will show that the bounds are tight by giving a construction that meets them.

4.2.2 An optimal construction

The following construction is a slightly modified version of the construction given in [111]. We show that the construction has the minimum length of keys for users and the authenticator, and meets the bounds in Theorem 4.1 with equality. We first briefly review Blom key distribution scheme.
4.2. DMRA-codes with a Single Sender

Blom key distribution scheme

Let \( q \geq n \) be a prime power. The TA randomly chooses a symmetric polynomial, \( F(x, y) \), with coefficients in \( GF(q) \) and of degree less than or equal to \( w \). For \( 1 \leq i \leq n \), the TA computes the polynomial \( G_i(x) = F(x, i) \) and gives \( G_i(x) \) to user \( P_i \), i.e., \( G_i(x) \) is the secret information of \( P_i \). The key associated with the pair of users \( P_i \) and \( P_j \) is calculated as, \( k_{ij} = G_i(j) = G_j(i) \). It is proved \([23]\) that the scheme is unconditionally secure against the collusion of \( w \) users in the following sense: the coalition of any \( w \) out of \( n \) users, say \( P_{i_1}, \ldots, P_{i_w} \), has no information about the key \( k_{ij} \) for the pair \( i, j \), where \( i, j \notin \{i_1, \ldots, i_w\} \).

\((w, n)\) MRA-code with dynamic sender based on Blom’s scheme

The \((w, n)\) MRA-code with dynamic sender based on the Blom’s scheme, works as follows. Let \( S \) be the set of source states and \( q \geq \max\{||S||, n\} \) be a prime power.

1. **Key distribution:** The TA chooses \( n \) distinct numbers \( a_i \) in \( GF(q) \) (associate \( a_i \) to user \( P_i \), \( 1 \leq i \leq n \)). These values are public and are used as identity information for users. Then the TA randomly chooses 2 symmetric polynomials of degree less than \( w \) with coefficients in \( GF(q) \),

\[
F_{\ell}(x, y) = (1, x, \ldots, x^w) A_{\ell} \begin{pmatrix} 1 \\ y \\ \vdots \\ y^w \end{pmatrix}, \quad \ell = 0, 1,
\]

where \( A_{\ell} \) is a \((w + 1) \times (w + 1)\) symmetric matrix for \( \ell = 0, 1 \). For \( 1 \leq i \leq n \), the TA computes the polynomials

\[
G_{\ell i}(x) = F_{\ell}(x, a_i) = (1, x, \ldots, x^w) A_{\ell} \begin{pmatrix} 1 \\ a_i \\ \vdots \\ a_i^w \end{pmatrix}, \quad \ell = 0, 1,
\]

and gives the 2-tuple of polynomials, \( (G_{0i}(x), G_{1i}(x)) \), to user \( P_i \). This constitutes the secret information of \( P_i \).

2. **Broadcast:** For \( 1 \leq i \leq n \), assume that the user \( P_i \) wants to generate the authenticated message for a source state \( s \in S \). \( P_i \) computes the polynomial \( M_i(x) = G_{0i}(x) + sG_{1i}(x) \) and broadcasts \( (s, a_i, M_i(x)) \).
3. Verification: The user $P_j$ can verify the authenticity of the message in the following way. $P_j$ accepts $(s, a_i, M_i(x))$ as authentic and being sent from $P_i$ if $M_i(a_j) = G_{0j}(a_t) + sG_{1j}(a_t)$.

**Theorem 4.2** The above scheme is a $(w, n)$ MRA-code with dynamic sender with $P_I = P_S = 1/q$.

**Proof.** Assume that after seeing an authenticated message $(s, a_i, M_i(x))$ broadcasted by the user $P_i$, the users $P_1, \ldots, P_w$ want to generate a new message $(s', a_i, M_i'(x))$, where $s' \neq s$ such that the user $P_j$ will accepts it as authentic, i.e. $M_i'(a_j) = G_{0j}(a_t) + s'G_{1j}(a_t)$. First, we observe that for each $m \in GF(q)$ each user, say $P_t$, can calculate

the polynomial $G_{0t}(x) + mG_{1t}(x) = (1, x, \cdot \cdot \cdot , x^{w-1})(A_0 + mA_1)$.

It follows that for each $m \in GF(q)$, $P_1, \ldots, P_w$ can calculate a $(w + 1) \times w$ matrix $D[m]$ such that the following identity holds,

$$
(A_0 + mA_1) \begin{bmatrix} 1 & \cdots & 1 \\
 a_1 & \cdots & a_w \\
 \vdots & \cdots & \vdots \\
 a_1^w & \cdots & a_w^w 
\end{bmatrix} = D[m]. \tag{4.1}
$$

Since $(s, a_i, M_i(x))$ is broadcasted, it follows that $P_1, \ldots, P_w$ know the following polynomial,

$$
g(x) = (1, x, \cdots , x^w)(A_0 + sA_1) \begin{bmatrix} 1 \\
 a_i \\
 \vdots \\
 a_i^w 
\end{bmatrix}.
$$

By combining equation (4.1) and the polynomial $g(x)$, $P_1, \ldots, P_w$ can also calculate matrices $B$ and $C$ such that the following equations hold.

$$
A_0 + sA_1 = C \tag{4.2}
$$

$$
(A_0 + mA_1) \begin{bmatrix} 1 & \cdots & 1 \\
 a_1 & \cdots & a_w \\
 \vdots & \cdots & \vdots \\
 a_1^w & \cdots & a_w^w 
\end{bmatrix} = D[m] \text{ for all } m \in GF(q) \tag{4.3}
$$
We claim that in the equations (4.2) and (4.3) above, knowing $C$ and $D[m]$ for all $m \in GF(q)$ can not determine the 2-tuple matrices $(A_0, A_1)$. In fact, there exist $q$ distinct 2-tuple matrices $(A_0, A_1)$ satisfying equations (4.2) and (4.3). This is equivalent to the following statement: There exists a 2-tuple of matrices $(A_0, A_1) \neq (0, 0)$ such that the following equations hold

$$A_0 + sA_1 = 0 \quad (4.4)$$

$$\begin{bmatrix} 1 & \cdots & 1 \\ a_1 & \cdots & a_w \\ \vdots & \cdots & \vdots \\ a_1^w & \cdots & a_w^w \end{bmatrix} (A_0 + mA_1) = 0 \quad \text{for all } m \in GF(q) \quad (4.5)$$

Indeed, consider the symmetric polynomial,

$$F(x, y) = (x - a_1) \cdots (x - a_w)(y - a_1) \cdots (y - a_w)$$

$$= (1, x, \cdots, x^w) A \begin{pmatrix} 1 \\ y \\ \vdots \\ y^w \end{pmatrix},$$

where $A$ is a $(w+1) \times (w+1)$ symmetric matrix and $A \neq 0$. We define $A_0 = -sA$ and $A_1 = A$, then it is not difficult to verify that $(A_0, A_1)$ satisfies the desired properties.

We note that since $(-sA, A)$ satisfies equations (4.4) and (4.5), so does $(-rsA, rA)$ for all $r \in GF(q)$. This implies that there are $q$ distinct 2-tuple of symmetric polynomials which are equally likely to be chosen by the TA. For each 2-tuple matrices $(A_0, A_1)$ of the from $(-rsA, rA)$, let

$$(1, a_j, \cdots, a_j^w)(A_0 + s'A_1) = d.$$

Then it is straightforward to verify that $d = 0$ if and only if $r = 0$. This is equivalent to that the $q$ distinct possible 2-tuple polynomials $(F_0(x, y), F_1(x, y))$ chosen by the TA result in $q$ distinct values of the form $F_0(a_i, a_j) + s'F_1(a_i, a_j)$. Therefore the probability of message substitution attack $P_{s\text{message}}$ is $1/q$. Similarly, we can prove $P_{s\text{entity}} = P_I = 1/q$. 
We see that in this construction the size of each user's key is $|E_i| = q^{2(w+1)}$, for all $1 \leq i \leq n$, and the size of codewords is $M_i = q^{w+2} = q^{w+1} |S|$. Thus we have shown that the bounds given in Theorem 4.1 are satisfied with equality.

This construction, although optimal, is very restrictive as $q$ determines not only deception probability but also the size of the key space and the length of the tag. More specifically, for $P_D = 1/q$ key size for each user and authenticator lengths are both of order $O(\log q)$. For large size sources, and/or large groups, the construction requires very high values for $q$ which results in unnecessarily high protection at the expense of very large key spaces and long authenticators.

4.2.3 A General Construction

In the following we show a general construction for $(w,n)$ DMRA-codes by combining a key distribution pattern and an A-code such that the security of the resulting system is determined by the security of the underlying A-code and the parameters of the key distribution pattern. The importance of this construction is that it provides a much higher degree of flexibility in the design of DMRA-codes and results in constructions that are practical. The construction can be seen as an extension of the MRA-code construction in [111] but uses KDP instead of a cover-free family.

Key distribution patterns (KDP) [77] are explicitly or implicitly used by numerous authors to construct key distribution systems [45, 58, 75, 78, 85, 110, 112]).

Let $X = \{x_1, x_2, \ldots, x_v\}$ be a set, and $B = \{B_1, B_2, \ldots, B_n\}$ be a family of subsets of $X$. The set system $(X, B)$ is called a $(n, v, w)$ key distribution pattern (or KDP$(n, v, w)$ for short) if

$$|(B_i \cap B_j) \setminus (\cup_{s=1}^{w} B_s)| \geq 1$$

for any $w + 2$ subset $\{i, j, \ell_1, \ldots, \ell_w\}$ of $\{1, 2, \ldots, n\}$.

Assume there are $n$ users $P_1, \ldots, P_n$. Let $(X, B)$ be a KDP$(n, v, w)$ and $(S, A_0, E_0)$ be a Cartesian authentication code such that the probability of deception (impersonation and substitution attacks) is bounded by $1/q$. Associate $B_i$ to $P_i$, $1 \leq i \leq n$. Both $(X, B)$ and $(S, A_0, E_0)$ are public.

1. Key distribution: For each $1 \leq j \leq v$, TA randomly chooses an authentication key $e_j \in E_0$ and gives $e_j$ to user $P_i$ if $x_i \in B_j$. Thus, user $P_i$ receives a $|B_i|$-tuple, $(e_{i1}, \ldots, e_{i|B_i|}) \in E_0^{[B_i]}$, as his/her secret authentication key.
2. **Broadcast:** When $P_i$ wants to construct an authenticated message for a source state $s \in S$, he computes $|B_i|$ partial authenticators $e_{i_t}(s)$, $1 \leq t \leq |B_i|$, and broadcasts $(s, e_{i_1}(s), \ldots, e_{i_{|B_i|}}(s))$ together with his identity $i$.

3. **Verification:** A user can verify authenticity and origin of the broadcast message in the following way: $P_j$ uses the origin information, $i$, to determine the set $\mathcal{E}_{ij} = \{e_{i_1}\}_{1 \leq k \leq |B_i|} \cap \{e_{j_k}\}_{1 \leq k \leq |B_j|}$ and accepts $(s, e_{i_1}(s), \ldots, e_{i_{|B_i|}}(s))$ as authentic and sent from $P_i$ if for all $e \in \mathcal{E}_{ij}$, $e(s)$ is the same as the corresponding component in $(e_{i_1}(s), \ldots, e_{i_{|B_i|}}(s))$.

**Theorem 4.3** Let the deception probability of the underlying A-code $(S, A_0, E_0)$ be bounded by $1/q$. Then the above construction results in a $(w, n)$ DMRA-code $C = (S, E, \{A_i, E_i\}_{1 \leq i \leq n})$ with $P_D \leq 1/q$. The code has the following parameters

\[
|E| = |E_0|^w, \quad |E_i| = |E_0||B_i|, \quad \text{and} \quad |A_i| = |A_0||B_i|.
\]

**Proof.** Assume there are $w$ colluding users, $P_1, \ldots, P_w$, who see a broadcast message, and want to construct a fraudulent message to be accepted by $P_j$ and attributed to $P_i$. Since $|(B_i \cap B_j) \setminus (\bigcup_{i=1}^w B_i)| \geq 1$, it follows that there exists at least one keys $e_{i,j}$ from the A-code that is known to $P_i$ and $P_j$, but is unknown to $P_1, \ldots, P_w$. Because the success probability of colluders in correctly guessing this key is not better that the outsiders, and because of the properties of the underlying A-code, their success probability is bounded by $1/q$.

The construction also works for general A-codes in which case the broadcast codeword by $P_i$ is $(m_{i_1} \cdots m_{i_{|B_i|}})$ where $m_{i_j} = e_{i_j}(s)$.

The main advantages of the construction is its flexibility in the choice of parameters for different applications. The following example shows the effectiveness of the above construction.

**Example 4.1** Assume a network with 5000 users such that the biggest size of colluding subsets is 5 and messages are strings of size $2^{50}$ bits (i.e. $|S| = 2^{250}$).

1. Using the optimal construction we have:
   - The key storages for the TA and each user are $42 \times 2^{50}$ bits and $12 \times 2^{50}$ bits, respectively;
   - The length of the authenticator is of $6 \times 2^{50}$ bits;
   - $P_D \leq 1/2^{250}$ (unnecessarily low).

2. Using the construction given in this section: Assume $P_D \leq 1/2^{19}$. This is acceptable for all practical purposes. We use an A-code that provides enough protection,
and a KDP with \( n = 5000 \) and \( w = 5 \). We use the A-code which is based on universal hashing families constructed from geometric codes given in [16], with the following parameters: \(|S| = 2^{250}\), authenticator length of 20 bits, and the key length of 125 bits. Next we use the probabilistic method developed in [45], to construct a KDP\((n,v,w)\) with \( n = 5000, v = 3928, w = 5 \) and \( |B_i| = 1123 \) on average. Combining the two, we obtain a \((5, 5000)\)-DMRA code with the following parameters.

- The key storages for the TA and each user are 591 Kbits and 140 Kbits respectively;
- The length of authenticator is 2.5 Kbits;
- \( P_D \leq 1/2^{19} \).

Comparing the optimal construction and the previous scheme shows that the key storage and the communication cost are dramatically reduced in the latter, while the deception probability has only increased from \( 1/2^{250} \) to \( 1/2^{19} \) which for practical purposes is insignificant.

This construction is especially important in practical applications where secure key storage and communication bandwidth are scarce and valuable resources.

To obtain key efficient constructions from Theorem 4.3 we require a KDP\((n,v,w)\) with small \( v \). The trivial key distribution KDP\((n,v,w)\) with \( w \leq n - 2 \) has \( B \) equal to the collection of pairs of \( X \), and so \( n = \frac{v(v-1)}{2} \) and \( |B_i| = v - 1 \) for all \( 1 \leq i \leq n \). A 'good' KDP\((n,v,w)\), is one that for given \( v \) and \( w \), \( n \) and \( |B_i| \) are as small as possible for \( 1 \leq i \leq n \). The constructions in [77] and [58] both require \( v = O(n) \) which is much better than the trivial construction with \( v = O(n^2) \). Dyer, Fenner, Frieze and Thomason [45] gave a probabilistic construction with \( v = O(\log n) \), but an explicit construction having \( v = O(\log n) \) is not known. From [112] we know explicit constructions where \( v \) is a polynomial function of \( \log n \) exist.

It is worth noting that the above construction based on KDPs can only result in 'good' DMRA-code (small \( v \)) if \( w \) is small compared to \( n \). This is to be expected because of the relationship between KDPs and cover-free families and the known bounds on the latter. More specifically, \((X, B)\) where \( X \) is a point set and \( B \) is a family of subsets of \( X \), is called a \( w \)-cover-free family if \( B_{i_0} \not\subset B_{i_1} \cup \cdots \cup B_{i_w} \) holds for all \( B_{i_0}, B_{i_1}, \ldots, B_{i_w} \in B \), where \( B_{i_k} \neq B_{i_j} \) if \( i_k \neq i_j \). A KDP\((n,v,w)\) gives a \( w \)-cover-free family by considering \( B_i \) as the point set and \( \{B_i \cap B_j; j = 1, 2, \ldots, n \) and \( j \neq i\} \) as \( B \). The proof of this claim is straightforward and is omitted. Applying a result of Erdős et al in [47] (Proposition 3.4), it follows that if \( \frac{(w+2)(w+1)}{2} > n - 1 \), then \( |B_i| \geq n - 1 \). This means that the the size of the block in the original KDP must be more than the trivial scheme and so is not possible (the trivial scheme has the biggest block size).
4.3 DMRA-codes with Multiple Senders

In this section we consider tDMRA-codes, with \( t \geq 2 \). In designing a \((w, n)\) tDMRA-code with \( t \geq 2 \), it is important to note that if the protection for a channel between two participants \( P_i \) and \( P_j \) is provided by a symmetric key system, then a message sent by \( P_i \) can be later resent and attributed to \( P_j \). In this case \( P_i \) will accept the message as authentic from \( P_j \) and the success chance of the intruder is 1. To avoid this directional attack, \( P_i \) and \( P_j \) must have different keys.

In the following we first look at two simplistic approaches to the construction of a tDMRA-code and then give two constructions with provable security that are shown to be much more efficient.

**Trivial Construction 1.** An obvious method of constructing a tDMRA-code is to use \( t \) copies of a DMRA-code with \( t \) independent keys. That is, in the key generation phase the TA chooses \( t \) independent keys, \( e^1, e^2, \ldots, e^t \), for a DMRA-code and gives the user \( P_i \), a \( t \) tuple, \((e^1_i, e^2_i, \ldots, e^t_i)\). A user \( P_i \) will use key \( e^\ell_i \) to authenticate (generate or verify) the \( \ell \)th message. In this case the size of the key for each participant is \( t \) times that of a DMRA-code, which for efficient DMRA-codes and small \( t \) could be reasonably low. The length of the tag for each message is the same as the original DMRA-code. However the system is unacceptable as it requires each user to carefully keep track of all the communicated messages and use the correct key for each particular message. If a message is missed, all future communications will be lost.

**Trivial Construction 2.** A second immediate solution will be to use a \((w, n - 1)\) MRA-code. In this case each user receives the key information for sending one message, and the key information for verifying \( n - 1 \) messages. The result is a \((w, t)\) tDMRA-code with \( t = n \). The length of tag in this case is the same as the MRA-code but the key storage is at least a linear function of \( n \). This means that the key storage will be prohibitive for large groups.

4.3.1 A polynomial construction for tDMRA-codes

The first non-trivial construction uses polynomials (non-symmetric) in two variables, and can be considered as an extension of the above optimal DMRA-code.

Let \( S = GF(q) \) be the set of source states. We construct a \((w, n)\) tDMRA-code as follows.
4.3. DMRA-codes with Multiple Senders

1. **Key Distribution:** TA randomly chooses two matrices

\[ A, B \in GF(q)^{(w+1)\times(w+t+1)} \]

, and computes two polynomials

\[ F(x, y) = (1, x, \ldots, x^w)A(1, y, \ldots, y^{w+t})^T \]

and

\[ G(x, y) = (1, x, \ldots, x^w)B(1, y, \ldots, y^{w+t})^T. \]

Then he chooses \( n \) distinct numbers \( a_i \in GF(q) \) and \( n \) distinct numbers \( b_i \in GF(q) \), where \( (a_i, b_i) \) is \( P_i \)'s identity information, and makes them public. For each \( i, 1 \leq i \leq n \), the TA privately sends two pairs of polynomials \( (F(x, a_i), G(x, a_i)), (F(b_i, y), G(b_i, y)) \) to \( P_i \). This constitutes the secret information of \( P_i \).

2. **Broadcast:** If \( P_i \) wants to authenticate a message \( s_i \in GF(q) \), \( P_i \) calculates the polynomial \( U_i(x) = F(x, a_i) + s_iG(x, a_i) \) and broadcasts \( (s_i, U_i(x)) \) and his identity \( a_i \) to other users.

3. **Verification:** \( P_j \) can verify authenticity of \( (s_i, U_i(x)) \) by calculating the polynomial \( V_j(y) = F(b_j, y) + s_jG(b_j, y) \) and accepting \( (s_i, U_i(x)) \) as authentic and being sent from \( P_i \) if \( V_j(a_i) = U_i(b_j) \).

**Theorem 4.4** The above construction results in a \((w, n)\) \( t \)-DMRA-code

\[ C = (S, E, \{A_i, E_i\}_{1 \leq i \leq n}), \]

in which the probability of success for a collusion of up to \( w \) users in performing impersonation or substitution attacks on any other pair of users is at most \( 1/q \). The construction has the following parameters

\[ |E| = q^{2(w+1)(t+w+1)}, |E_i| = q^{4w+2t+4} \text{ and } |A_i| = q^{w+1}, 1 \leq i \leq n. \]

**Proof.** Assume that \( \{P_1, \ldots, P_w\} \) are the colluders. We need to consider the following types of attacks:

1. Without seeing any communication, the colluders construct a message fraudulent message \((s_i, a_i, A_i(x))\) such that \( P_j \) would accept as authentic and sent from \( P_i \).
2. After seeing \( t \) broadcast messages, \((s_{i\ell}, a_{i\ell}, U_{i\ell}(x))\), \( \ell = 1, \ldots, t \) from users \( P_{i1}, \ldots, P_{it} \), the colluders will have one of the following substitution attacks:
   a. For some \( i_\ell \), say \( i_1 \), \( \{P_1, \ldots, P_w\} \) generate a message \((s'_{i_1}, a_{i_1}, U'_{i_1}(x))\), where \( s'_{i_1} \neq s_{i_1} \), such that \( P_j \), for some \( j \in \{1, \ldots, n\}\setminus\{1, \ldots, w\} \), will accepts it as authentic.
and sent from $P_{i'}$. (b.) colluders, $\{P_1, \ldots, P_w\}$, construct a message $(s, a_{i'}, U_{i'}(x))$, where $i' \notin \{i_1, \ldots, i_t\}$ and $s$ may or may not be in $\{s_{i_1}, \ldots, s_{i_t}\}$, such that $P_j$, $j \in \{1, \ldots, n\} \setminus \{1, \ldots, w\}$, will accepts it as authentic and sent from $P_{i'}$. We only give a proof sketch for (2.b) and note that the other two cases can be proved in a similar way. In (2.b), the colluders construct a fraudulent message $(s, a_{i'}, A_{i'}(x))$ aimed at $P_j$. They succeed if they can correctly guess $F(b_j, a_{i'}) + sG(b_j, a_i)$, where $F(x, y)$ and $G(x, y)$ are the two polynomials chosen by the TA. Now for any given $s$, let $s' \in GF(q)$ and $1 + ss' \neq 0$. Consider the polynomials

$$F_r(x, y) = F(x, y) + r(x - b_1) \cdots (x - b_w)(y - a_1) \cdots (y - a_w)(y - a_{i_1}) \cdots (y - a_{i_t})$$

$$G_r(x, y) = G(x, y) + rs'(x - b_1) \cdots (x - b_w)(y - a_1) \cdots (y - a_w)(y - a_{i_1}) \cdots (y - a_{i_t})$$

Then it can be verified that

$$F_r(x, a_i) = F(x, a_i), G_r(x, a_i) = G(x, a_i) \quad \text{and} \quad F_r(b_i, y) = F(b_i, y), G_r(b_i, y) = G(b_i, y)$$

for $1 \leq i \leq w$, and for each $s_{i_t} \in GF(q)$, $F(x, a_{i_t}) + s_{i_t}G(x, a_{i_t}) = F_r(x, a_{i_t}) + s_{i_t}G_r(x, a_{i_t})$. This means that if instead of $F(x, y)$ and $G(x, y)$, the TA had chosen $F_r(x, y), G_r(x, y)$, the colluders had the same information from their secret and the observed messages. Now it can be verified that $F(b_j, a_{i'}) + sG(b_j, a_{i'}) = F_r(b_j, a_{i'}) + sG_r(b_j, a_{i'})$ if and only if $r = 0$. Since $r$ is an arbitrary number in $GF(q)$, and different $r$ give different $F_r(b_j, a_{i'}) + sG_r(b_j, a_{i'})$, this implies that the success probability is $1/q$.

Comparing this construction with trivial construction 2, shows marked improvement of efficiency. In particular, let $n$ independent copies of the optimal MRA-codes based on polynomial scheme [41] be used. It will result in a $(w, n)$ tDMRA-code $C = (S, E, \{M_i, E_i\}_{1 \leq i \leq n})$ with parameters $|E| = q^{\frac{(w+1)n}{2}+n}$ and $|E_i| = q^{2(n-1+w)}$ and $|A_i| = q^{w+1}$ and so the size of the key space for the TA and users are $O(n \log q)$. Thus, when $n$ is much larger than $t$, the key storage for the TA and users can be significantly reduced.

### 4.3.2 A general construction from perfect hash families

In this section, we present a general approach to the construction of DMRA-codes by combining a general A-code and a perfect hash family.

**Perfect Hash Families**

Perfect hash families (PHF) originally arose as part of compiler design—see [76] for a summary of the early results in this area. They have applications to numerous
areas of computer science such as operating systems, language translation systems, and information retrieval systems—see [32] for a survey of recent results. PHF have also been applied to cryptographic applications such as broadcast encryption systems [49], secret sharing schemes [20], and threshold cryptography [19, 37].

A \((n, m, w)\)-perfect hash family is a set of functions \(F\) such that
\[ f : \{1, \ldots, n\} \to \{1, \ldots, m\} \]
for each \( f \in F\), and for any \( X \subseteq \{1, \ldots, n\} \) such that \( |X| = w \), there exists at least a function \( f^X \) in \( F \) such that \( f^X \) is injection on \( X \), i.e. the restriction of \( f^X \) on \( X \) is one-to-one. For a subset \( X \), if the restriction of a function \( f \) on \( X \) is one-to-one, then we call \( f \) perfect on \( X \). We will use the notation \( PHF(N; n, m, w) \) for a \((n, m, w)\) perfect hash family with \(|F| = N|\).

Let \( C_0 = (S, \mathcal{E}_0, A_0) \) be a Cartesian A-code (code without secrecy) with deception \( P_D \leq \epsilon \), and assume \( F = \{f_1, \ldots, f_n\} \) be a \( PHF(N; n, n_0, w + t + 2) \) from \( \{1, \ldots, n\} \) to \( \{1, \ldots, n_0\} \). We construct a \((w, n)\) tDMRA-code \( C = (S, \mathcal{E}, \{A_j, \mathcal{E}_j\}_{1 \leq j \leq n}) \) as follows.

1. **Key Distribution:** The TA randomly chooses \( N \) matrices of size \( n_0 \times n_0 \),
\[ G^1 = (g_{u,v}^1)_{1 \leq u \leq n_0, 1 \leq v \leq n_0}, \ldots, G^N = (g_{u,v}^N)_{1 \leq u \leq n_0, 1 \leq v \leq n_0} \]
with entries \( g_{u,v}^\ell \in E_0 \), for all \( 1 \leq \ell \leq N \) and \( 1 \leq u, v \leq n_0 \). For each \( i, 1 \leq i \leq n \), the TA generates the key \( e_i \) for \( P_i \) as a collection of \( N \) matrices each with a single non-zero row and a single non-zero column given by,
\[ e_i = (e_{i1}, e_{i2}, \ldots, e_{iN}), \]
where
\[ e_{i,\ell} = \begin{bmatrix} g_{f_\ell(i),1}^\ell & \cdots & g_{f_\ell(i),f_\ell(i)-1}^\ell & g_{f_\ell(i),f_\ell(i)}^\ell & \cdots & g_{f_\ell(i),n_0}^\ell \\ \vdots & & \vdots & & & \vdots \\ \vdots & & \vdots & & g_{n_0,f_\ell(i)}^\ell \end{bmatrix} \]
for all \( 1 \leq \ell \leq N \). The TA then securely sends \( e_i \) to \( P_i \). That is, the secret key of \( P_i \) consists of the \( f_\ell(i) \)th column and the \( f_\ell(i) \)th row of matrix \( G^\ell \), for all \( 1 \leq \ell \leq N \).

2. **Broadcast:** If \( P_i \) wants to authenticate a message \( s_i \in S \), \( P_i \) generates his authenticator \( a_i \) for \( s_i \) by
\[ a_i = \begin{bmatrix} g_{1,f_1(i)}^1(s_i) \\ \vdots \\ g_{n_0,f_1(i)}^1(s_i) \\ \vdots \\ g_{1,f_n(i)}^N(s_i) \\ \vdots \\ g_{n_0,f_n(i)}^N(s_i) \end{bmatrix} \]
and broadcasts \((i, s_i, a_i)\) to all other members. That is, \(P_i\) uses his column keys to generate the authenticator.

3. **Verification**: \(P_j\) uses his row key

\[
([g^1_{f(j),1}, \ldots, g^1_{f(j),n_0}], \ldots, [g^N_{f_N(j),1}, \ldots, g^N_{f_N(j),n_0}])
\]

to verify the authenticity of the broadcasted message \((i, s_i, a_i)\) in the following way. For each \(\ell, 1 \leq \ell \leq N\), the \(f_\ell(j)\)th row and the \(f_\ell(i)\)th column of matrix \(G^\ell\) have the common entry \(g^\ell_{f_\ell(j),f_\ell(i)}\). This is used by \(P_j\) to verify if \(g^\ell_{f_\ell(j),f_\ell(i)}(s_i)\) is the correct component of \(a_i\).

**Theorem 4.5** Suppose there exists a Cartesian \(A\)-code \(C_0 = (S, E_0, A_0)\) with deception probability \(P_D \leq \epsilon\), and a PHF\((N; n, n_0, t + w + 2)\). Then the above construction results in a \((w, n)\) tDMRA-code \(C = (S, E, \{A_j E_j\}_{1 \leq j \leq n})\) with deception probability \(P_D^* \leq \epsilon\). The various parameters satisfy

\[
|E| = |E_0|^{n_0^2}, \quad |E_j| = |E_0|^{(2n_0 - 1)N} \quad \text{and} \quad |A_j| = |A_0|^{n_0 N}, \quad 1 \leq j \leq n
\]

**Proof.** Without loss of the generality, we assume that \(w\) malicious users are \(P_1, \ldots, P_w\), after seeing the \(t\) broadcast messages from \(t\) users, say \(P_{w+1}, \ldots, P_{t+w}\), the malicious users want to perform an attack on a pair of users \(P_i\) and \(P_j\), that is, they generate a message \((s_i, a_i)\) and hope that \(P_j\) will accept it as an authenticated message from \(P_i\). There are four cases to be considered: **Case 1**: \(P_i, P_j \not\in \{P_{w+1}, \ldots, P_{t+w}\}\); **Case 2**: \(P_i \in \{P_{w+1}, \ldots, P_{w+t}\}\), but \(P_j \not\in \{P_{w+1}, \ldots, P_{w+t}\}\); **Case 3**: \(P_i \not\in \{P_{w+1}, \ldots, P_{w+t}\}\), but \(P_j \in \{P_{w+1}, \ldots, P_{w+t}\}\); **Case 4**: \(P_i, P_j \in \{P_{w+1}, \ldots, P_{w+t}\}\).

Let \(L = \{1, \ldots, w + t, i, j\} \subseteq \{1, \ldots, n\}\). Then \(|L| \leq w + t + 2\). Since \(F\) is a PHF\((N; n, n_0, w + t + 2)\), it is also a PHF\((N; n, n_0, |L|)\). It follows that there exists a \(f_\ell \in F\) such that \(f_\ell\) is one-to-one on \(L\). We consider the \(\ell\)th component keys for \(P_1, \ldots, P_t, P_{t+1}, \ldots P_{w+t}, P_i, P_j\), their keys corresponding to the matrix \(G^\ell\). Clearly, \(P_1, \ldots, P_t\), pooling their (row and column) keys, has no information about \(g^\ell_{f_\ell(j),f_\ell(i)}\), so the probability that \(P_1, \ldots, P_t\) can correctly generate the partial authenticator \(g^\ell_{f_\ell(j),f_\ell(i)}(s_i)\), denoted by \(P[i,j]\), satisfies

\[
P[i,j] = \begin{cases} 
P_i & \text{(case 1)} \\
P_S & \text{(case 2)} \\
P_i & \text{(case 3)} \\
P_S & \text{(case 4)}
\end{cases}
\]
where $P_I$ and $P_S$ denote the deception probability for impersonation attack and substitution attack of the underlying A-code $C = (\mathcal{S}, \mathcal{E}_0, \mathcal{A}_0)$, respectively. The desired result follows immediately.

**Remarks**

1. We can slightly improve the key sizes for users and the length of authenticators in the above construction. Observe that for each $1 \leq i \leq n$, $P_i$ has the partial keys $g^\ell_{f(i),f(i)}$, which is used to generate ‘partial’ authenticator that can be verified by $P_j$, or to verify authenticity of ‘partial’ authenticator sent from $P_j$, when and only when $f^\ell(i) = f^\ell(j)$. However, given that $f^\ell(i) = f^\ell(j)$, $P_i$ and $P_j$ hold the same $\ell$th component row and column key corresponding to the matrix $G^\ell$, and so $P_i$ can use his $\ell$th component column key to verify the $\ell$th component of the authenticator sent by $P_j$. A similar argument applies to $P_j$ for verifying authenticity of a message received from $P_i$. Thus we can remove the ‘partial’ key $g_{f(i),f(i)}, \ldots, g_{f_N(i),f_N(i)}$ without reducing the security of the scheme, except that the verification of broadcast messages will use some column keys. In other words, the keys in the diagonal of matrices $G^1, \ldots, G^N$ can be removed. In this case, we can save $N \log |\mathcal{E}_0|$ bits for each user’s key and $N \log |\mathcal{A}_0|$ bits for each broadcast message.

2. The construction also results in a $(w', n)$ $t'$DMRA-code if $w' + t' = w + t$. In other words, there is a trade-off between the number of senders and the number of malicious users the system can tolerate.

3. For a given set of parameters, $w, t$ and $n$, and a given A-code the efficiency of the scheme is completely determined by $N$, the size of perfect hash family $F$. Let $N(n, m, w)$ denote the minimum value of $N$ such that a PHF($N; n, m, w$) exists. Thus we will be interested in perfect hash families with small $N(n, m, w)$ for given $n, m$ and $w$. In particular, we are interested in the behaviour of $N(n, m, w)$ as a function of $n$, when $m$ and $w$ are fixed. It is proved in [76] that for fixed $m$ and $w$, $N(n, m, w)$ is $\Theta(\log n)$, however, the proof is non-constructive and PHF that achieve this asymptotic bound are believed to be difficult to construct. ($f(n) = \Theta(\log n)$ means that there exist constants $c_1, c_2$ and $n_0$ such that for $n > n_0$, $c_1 \log n \leq f(n) \leq c_2 \log n$.) In [3, 21] some constructions with reasonable asymptotic performance are given. For example, for fixed $m$ and $w$, $N$ is a polynomial function of $\log n$. Various other bounds on $N(n, m, w)$ can be found in [76, 3, 32, 19].
The basic idea behind the above construction is to use $N$ copies of a $(w, n_0)$ tDMRA-code with small $n_0$, to construct a $(w, n)$ tDMRA-code with large $n$, using a PHF with suitable parameters. In the above construction the $(w, n_0)$ tDMRA-code is obtained through the trivial construction 2, using $n_0$ copies of a MRA-code which itself is obtained from the ‘synthesis’ of an A-code and a trivial cover-free family [111]. Using the method identical to the construction given in this subsection, one can use any other $(w, n_0)$ tDMRA-code in conjunction with the PHF to obtain tDMRA-code for larger number of users. The following theorem can be proved.

**Theorem 4.6** Suppose there exist a Cartesian $(w, n_0)$ tDMRA-code $C = (S, \mathcal{E}, \{A_i, \mathcal{E}_i\}_{1 \leq i \leq n_0}$ with deception probability $P_D \leq \epsilon$, and a PHF$(N; n, n_0, t + w + 2)$. Then there exists a $(w, n)$ tDMRA-code $C^* = (S, \mathcal{E}^*, \{A_j^*, \mathcal{E}_j^*\}_{1 \leq j \leq n})$ with deception probability $P_D \leq \epsilon$. The various parameters satisfy

$$|\mathcal{E}^*| = |\mathcal{E}|^N, \quad |\mathcal{E}_j^*| \leq \max_{1 \leq i \leq n_0} \{|\mathcal{E}_i|^N\} \quad \text{and} \quad |A_j^*| \leq \max_{1 \leq i \leq n_0} \{|A_i|^N\}, \quad 1 \leq j \leq n$$

### 4.4 A Secure Dynamic Conference System

To show the usefulness of group authentication systems we will use DMRA-codes to construct a secure dynamic conference system. By ‘dynamic’ we means the conferencees are not predetermined and a conference can be held among any $c$ members of the group. By ‘secure’ we mean a member of the conference can send a message to all other conference members such that collusion of up to $w$ users, not in the conference, cannot learn anything about the message, and collusion of up to $w$ members of the conference (insiders) cannot substitute a broadcast message with a fraudulent one. We note that constructing a dynamic conference scheme that allows private communication among conferencees is an immediate consequence of a key distribution scheme for a dynamic conference. Such schemes provide a shared key among all conferencees that can be used to encrypt messages. However there is no authentication in the system and encrypted messages can be easily substituted by a malicious group of conferencees, without leaving any trace, simply because the key information is shared among members of the conference. To provide authenticity in the system we can use a DMRA-code.

In the following we give a construction that uses a key distribution system proposed by Blundo, De Santis, Herzberg, Kutten, Vaccaro and Yung [24] (BDHKVY for short) and ensures secrecy and authenticity of the communication in a dynamic conference.

**Key Distribution Systems**
A Key Distribution Systems (KDS) is one of the main primitives for distributing keys in network and group communication [110]. In a KDS, the collection of all subsets of \( n \) users is divided into privileged subsets and forbidden subsets. To each privileged subset, \( G \), of users a secret key, \( k_G \), is attached. \( k_G \) is computable by each member of \( G \) and collusion of members of a forbidden set \( F \), disjoint from \( G \), cannot learn anything about \( k_G \). A TA generates and distributes secret key information to all users.

If privileged sets are \( t \)-subsets of \( \mathcal{P} \), and forbidden sets are all \( w \)-subsets of \( \mathcal{P} \), we use the notation \((t, w)\) KDS. For example, a \((2, w)\)-KDS is a KDS where there is a key associated with each pair of users and no key \( k_{(i,j)} \) can be computed by collusion of any \( w \) users that is disjoint from \( \{i, j\} \). A \((t, w)\) KDS is also called a key distribution for dynamic conferences. A naive approach to constructing a \((t, w)\) KDS will result in prohibitive cost of key generation and distribution. This can be easily seen by noting that a simplistic solution to \((2, w)\)-KDS is an \( n^2 \) problem and for \( t > 2 \) complexity rapidly increases. BDHKVY proposed a \((c, w)\)-KDS in which each user has to store \( (c^{-1})\log q \) bits of key while the TA has to store \( (c^w) \log q \) bits of key which are the minimum possible storage requirements.

**BDHKVY \((c, w)\)-KDS:**

Before describing the schemes, we recall that a polynomial

\[
P(x_1, \ldots, x_c) = \sum_{0 \leq j_1, \ldots, j_c \leq w} a_{j_1, \ldots, j_c} x_1^{j_1} x_2^{j_2} \cdots x_c^{j_c}
\]

of degree at most \( w \), where \( a_{j_1, \ldots, j_c} \in GF(q) \), is said to be symmetric if

\[
P(x_1, \ldots, x_c) = P(x_{\sigma(1)}, \ldots, x_{\sigma(c)})
\]

for any permutation \( \sigma : \{1, 2, \ldots, c\} \rightarrow \{1, 2, \ldots, c\} \). The scheme consists of the following phases.

1. The TA randomly chooses a symmetric polynomial \( P(x_1, \ldots, x_c) \) in \( c \) variables of degree at most \( w \) with coefficients in \( GF(q), q \geq n \).

2. To each user \( P_i \) the TA gives the polynomial \( f_i(x_2, \ldots, x_c) = P(i, x_2, \ldots, x_c) \), that is the polynomial obtained by evaluating \( P(i, x_2, \ldots, x_c) \) at \( x_1 = i \).

3. If the users \( P_{j_1}, \ldots, P_{j_c} \) want to set up a common (conference) key the each user \( P_{j_i} \) evaluates \( f_{j_i}(x_2, \ldots, x_c) \) at \( (x_2, \ldots, x_c) = (j_1, \ldots, j_{i-1}, j_{i+1}, \ldots, j_c) \).

4. The common (conference) key for users \( P_{j_1}, \ldots, P_{j_c} \) is equal to \( k_{j_1, \ldots, j_c} = P(j_1, \ldots, j_c) \).
When \( c = 2 \), BDHKVY scheme coincides with Blom's scheme [23].

Given a \((c, w)\)-KDS, we can easily construct a broadcast encryption system in the following way. Assume that the TA wants to send a message \( s \in GF(q) \) to a group of users \( P_{j_1}, \ldots, P_{j_c} \), or one of the users, \( P_{j_1} \), wants to send \( s \) to other users in \( \{P_{j_2}, \ldots, P_{j_c}\} \). The TA, or \( P_{j_1} \), encrypts \( s \) as \( b = s + k_{j_1, \ldots, j_c} \) and broadcasts \( b \). Then any user in \( \{P_{j_1}, \ldots, P_{j_c}\} \) can decrypt \( b \) to obtain \( s \), by using \( s = b - k_{j_1, \ldots, j_c} \), and any group of at most \( w \) users that are disjoint from \( \{P_{j_1}, \ldots, P_{j_c}\} \) have no information about \( s \).

However the communication is not authenticated. That is, the origin of a message is not known and hence there is no accountability in the system. In the following we show how to add authenticity to this system without having more key bits.

1. **Key Distribution:** Assume that there is a BDHKVY \((c, w)\)-KDS, where the TA has randomly chosen a symmetric polynomial \( P(x_1, \ldots, x_c) \) in \( c \) variables and of degree at most \( w \), and privately transmitted the secret information \( P(i, x_2, \ldots, x_c) \) to each user \( P_i \). The field \( GF(q) \) is chosen such that \( q \geq \max\{ |S|, n+2\binom{n}{c}+c-2 \} \), where \( S \) is the set of source states. To each group of users, \( \{P_{j_1}, \ldots, P_{j_c}\} \), of size \( c \) we associate a number \( N_{j_1, \ldots, j_c} \) such that \( n < N_{j_1, \ldots, j_c} \leq 2\binom{n}{c} \) and such that if \( \{P_{j_1}, \ldots, P_{j_c}\} \neq \{P_{j_1}, \ldots, P_{j'_c}\} \) then \( |N_{j_1, \ldots, j_c} - N_{j'_1, \ldots, j'_c}| \geq 2 \). The numbers \( N_{j_1, \ldots, j_c} \) will serve as identity information for conferences and are made public.

2. **Broadcast:** Assume that \( P_{j_1} \) wants to encrypt a message \( s \in S \) and broadcast it such that each user in \( \{P_{j_2}, \ldots, P_{j_c}\} \) can decrypt the message and individually verify the authenticity and the origin of the message.

   (a) \( P_{j_1} \) constructs two polynomials, of degree at most \( w \),

   \[
   F_{j_1}(x_2) = f_{j_1}(x_2, N_{j_1, \ldots, j_c}, \ldots, N_{j_1, \ldots, j_c} + c - 2) \\
   G_{j_1}(x_2) = f_{j_1}(x_2, N_{j_1, \ldots, j_c} + 1, \ldots, N_{j_1, \ldots, j_c} + c - 1).
   \]

   \( P_{j_1} \) then encrypts \( s \) with the (conference) key \( k_{j_1, \ldots, j_c} \) to obtain \( b = s + k_{j_1, \ldots, j_c} \).

   (b) \( P_{j_1} \) computes the polynomial \( A_{j_1}(x_2) = F_{j_1}(x_2) + bG_{j_1}(x_2) \) of degree at most \( w \), and broadcasts \((b, j_1, A_{j_1}(x_2))\).

3. **Decryption and verification:** Each user \( P_{j_i} \) in \( \{P_{j_2}, \ldots, P_{j_c}\} \) can decrypt and verify the authenticity of the message broadcast by \( P_{j_1} \): in the same manner as (2.1) and (2.2), \( P_{j_i} \) can calculate \( A_{j_i}(x_2) \). Then, \( P_{j_i} \) verifies if \( A_{j_i}(j_1) = A_{j_1}(j_i) \) holds and if true, accepts the broadcast codeword as authentic from \( P_{j_1} \). Finally \( P_{j_i} \) decrypts \( b \) by \( s = b - k_{j_1, \ldots, j_c} \) to get \( s \).
Theorem 4.7 For \( c > 2 \), the above construction provides secrecy and authenticity for \((c, w)\)-KDS for dynamic conferences.

Proof. From [24], we know that the scheme is a \((c, w)\)-KDS. We need to show that it provides authenticity for the broadcast message. It is easy to see that,

\[
P(x_1, x_2, N_{j_1}, \ldots, j_c, \ldots, N_{j_1}, \ldots, j_c + c - 2) \quad \text{and} \quad P(x_1, x_2, N_{j_1}, \ldots, j_c + 1, \ldots, N_{j_1}, \ldots, j_c + c - 1)
\]

are symmetric polynomial in two variables of degree at most \( w \). Because of the properties of BDHKVY scheme [24], any colluding group of up to \( w \) users \( \{P_1, \ldots, P_n\} \), which is disjoint from \( \{P_1, P_2\} \), has no information about,

\[
k_{j_1, j_2, N_{j_1}, \ldots, j_c, N_{j_1}, \ldots, j_c + c - 2}, \quad \text{and} \quad k_{j_1, j_2, N_{j_1}, \ldots, j_c + 1, \ldots, N_{j_1}, \ldots, j_c + c - 1}
\]

This is the key information used for authentication between \( P_{j_1} \) and \( P_{j_2} \). Using the optimal DMRA-code (based on Blom's scheme) in [111], it follows that the messages are authenticated.

A few notes are in order.

1. Compared with the broadcast encryption scheme based on the BDHKVY KDS, the key storage of the above scheme need not to increase, if \(|S| \geq n + 2\binom{n}{c} + c - 2\).

2. In the above construction authenticity was added to the codeword broadcast from one of the users in a privileged group. We can slightly modify the above construction such that the encrypted message can be broadcast by the TA. This can be done by adding a dummy user for the TA, and construct a \((c + 1, w)\)-KDS. Then the whole process in the above construction can be carried out to cater this setting.

3. We have assumed the same level of security for secrecy and authenticity. In general we can have \( w_1 \) as the biggest size of colluding outsiders (not members of the conference) and \( w_2 (\leq w_1) \) as the biggest size of colluding insiders.

4.5 Computationally secure tDMRA-codes

The computational model for studying tDMRA-codes so far has been unconditionally secure model. Although unconditionally secure schemes offer the highest possible security but their key requirements is usually prohibitive and so such systems are usually
4.6 Conclusions

impractical. In practice, data integrity is obtained by using MACs (message authentication codes) and signature schemes. MACs can be seen as the computationally secure version of A-codes. Numerous constructions for MACs exist. MACs can be constructed from block cipher systems (for example DES) in CBC mode, or using cryptographic hash functions like MD5 and SHA-1. MACs with provable security can be obtained through Wegman-Carter construction[121].

A very important aspect of ‘synthesis’ constructions for MRA, DMRA and tDMRA-code is that they work with MACs too. Each ‘synthesis’ construction essentially combines an A-code with a combinatorial structure: a cover-free family, a KDP or a PHF, respectively. By replacing the A-code with a MAC, a system (MRA, DMRA and tDMRA-code) with computational security is obtained such that the security can be directly related to the security of the underlying MAC and parameters of the combinatorial structure.

This universality of ‘synthesis’ constructions is especially important because combinatorial techniques, such as constructing systems for large groups that is obtained through recursive constructions using PHFs, can also be imported to computationally secure model.

4.6 Conclusions

In this chapter, we studied broadcast authentication in group communication. We noted that DMRA-codes could be considered as the basic primitive for this service and gave two constructions, one optimal and flexible and having a ‘synthesis’ nature, for these systems. Although the constructions assume only one codeword sent by a sender, but it is not difficult to extend them to multiple messages from the sender. When multiple messages are from different senders, a new type of attack must be considered. The aim of the attack is to tamper with the origin information in a broadcast message. Protection against this attack implies that the keys used by two users must produce uncorrelated tags and so key distribution systems that establish common key among participant cannot be directly used for key distribution in group authentication systems. We gave two constructions for tDMRA-codes, one algebraic and one by a ‘synthesis’ method. ‘Synthesis’ constructions are especially interesting as they are universally applicable with A-codes, with and without secrecy, and MACs.

A DMRA-code is a powerful tool for securing group communications. We showed a construction for secure dynamic conference systems which provides confidentiality and
authenticity for communicated messages.

The question of optimality of $t$DMRA-code is only answered when $t = 1$. Deriving information theoretic and combinatorial bounds for general DMRA-codes, and constructing optimal $t$DMRA- systems are interesting open problems.
Chapter 5

Shared Generation of Authenticator Systems (SGA-systems)

Share generation of authentication systems (SGA-systems) are a generalisation of Simmons' traditional model of authentication where authenticating a message (also called a source state) requires collaboration of a group of senders. In a traditional A-code a single sender holds the authentication key which is also known to the receiver. In an SGA-system the key information is shared by \( n \) senders. To construct an authenticated message for a source state each sender in an authorised group produces a partial authenticator for the source state and sends it to the combiner who combines them into a full authenticator to be appended to the source state. In a \((t, n)\) threshold SAG-system any group of \( t \) senders is an authorised group and no coalition of \( t - 1 \) (or less) senders can produce a valid authenticator for a source state (impersonation attack) or, after seeing a valid codeword can produce a valid authenticator for a different source state (substitution attack). This means that in a \((t, n)\) threshold SGA-systems receiving an authenticated codeword implies authorisation of at least \( t \) senders. SGA-systems are an important cryptographic primitive in distributed systems where a number of parties located at distant geographic location are to collaboratively authorise an action, sign a document or share responsibility. Unconditional security of these systems result in many key bits to be distributed and stored and hence the systems are not viable for most practical applications. However, they are the only solution for strategic applications when assurance is required over a long period of time. The need for such systems are emphasised by the emergence of new models of computing such as quantum computers which could result in time efficient solutions for known hard problems. Unconditionally secure schemes provide the highest level of security and enjoy a conceptual clarity which is not matched by any other approach.

Threshold SGA-systems are part of a general approach to cryptography, known as threshold cryptography [35] which has received considerable attention in recent years [37]. While computationally secure threshold authentication, e.g. threshold RSA and
DSS signature [15, 31, 39, 55, 86] have been studied extensively, the information-theoretic secure model, i.e. threshold SGA-systems has received much less attention. Unconditionally secure threshold SGA-systems were first introduced by Desmedt, Frankel and Yung [39, 41]. The basic idea behind their schemes is to combine a traditional A-code and a secret sharing scheme to construct a threshold SGA-system. We note that a naive application of a secret sharing scheme to share the secret authentication key among the senders is not acceptable simply because it requires full trust on the combiner and in fact allows him to authenticate any message of his choice and without requiring any collaboration from senders, after receiving partial authenticators from an authorised group. The key assumption in an SGA-system is that the combiner is not trusted and the combining operation does not require any secret information.

In [91], three systems for threshold SGA, including a key efficient system and one based on error correcting codes, are proposed. In [54, 73, 44], the SGA-systems with a general access structure are considered. In this model, the access structure, that is the collection of authorised subsets of senders who can authenticate a message, is not (t, n) threshold. In [54], Gehrmann proved an information-theoretic lower bound on the deception probabilities of impersonation and substitution attacks which is a generalisation of Simmons’ and Brickell’s bounds [100, 27] for traditional A-codes. In [44] an efficient construction (short key length) of SGA-systems using maximum rank distance codes is described. In comparison, the model in [73] is the most general one while the work in [44] and [91], are limited to linear threshold SGA-system and provides more efficiency.

In this chapter, we will concentrate on unconditionally secure threshold SGA-systems. This restriction allows us to derive combinatorial parameters of these systems, extend some of the known constructions and present new efficient ones. The main contributions of this chapter are listed below.

**Combinatorial lower bounds** Security and efficiency of a (t, n) threshold SGA-system can be measured by a number of parameters: the overall deception probability; the key length of each sender and the receiver; and the lengths of the partial authenticators and the full authenticator. We derive combinatorial bounds on the key and authenticator lengths, when deception probability is fixed. We show that the bounds are tight and a polynomial construction given in [41] is optimal and meets the bounds with equality. This remedies an important shortcoming of the previous work and allows us to have clear efficiency comparison of the known constructions.
Key efficient schemes An optimal construction has minimum storage and communication cost for a given source space and deception probability. However, the only known optimal construction has very rigid conditions on parameter values that could result in practically unacceptable systems. In particular in the DFY optimal construction, the keys and authenticators' lengths grow with an increase in the source size while the deception probability decreases. This means that for very large source sizes, very long keys and authenticators must be used and at the same time deception probability is reduced to unnecessary low values. Our motivation for key efficient systems is to increase deception probability, while keeping it in an acceptably low range, where reducing the key and the authenticator lengths hence saving on secure storage and communication bandwidth. We give two constructions that accommodate such a trade-off: the first one is an extension of a key efficient \((t, t)\) threshold SGA-system in [91] to \((t, n)\) threshold, and the second one is a new construction based on error-correcting codes.

Recursive constructions We extend a recursive construction method, implicitly used by Blackburn et al [20] in the context of secret sharing schemes, to SGA-systems. The basic construction uses perfect hash families to construct a \((t, n)\) threshold SGA-system from a \((t, m)\) one, where \(n > m\), and its recursive nature means that it can be applied repeatedly to construct SGA-system for very large groups. This results in a particularly efficient construction when the number of the senders is large and the size of the source space is small.

Robustness In the basic model of SGA-systems, protection is against impersonation and substitution which are the two main authentication attacks. A very important attack on the system is, however, disrupting the working of the system by sending incorrect partial authenticators to the combiner. This will result in calculating an incorrect full authenticator which would be most likely rejected. If the chance of success for such an attack is high then the service becomes unavailable for the receiver. We call a \((t, n)\) threshold SGA-system \((t, n)\) robust if it can correctly produce the full authenticator even in the presence of up to \(t - 1\) arbitrary malicious senders. We discuss system robustness and construct two robust SGA-systems, one with unconditional security and the second one with computational security in order to achieve the robustness.

The chapter is organised as follows. In section 5.1 we describe the model of threshold
SGA-systems and define some important parameters, and then derive information-theoretic and combinatorial lower bounds on \((t, n)\) SGA-systems. In section 5.2 we present two efficient \((t, n)\) SGA-systems. In section 5.3 we show how to build new threshold SGA-systems for large groups from old ones (small groups) by using perfect hash families. In section 5.4 we discuss the robustness in SGA-systems and propose two constructions. In section 5.5 we study multireceiver SGA-systems.

5.1 Model and Bounds for SGA-systems

In an SGA-system there are

1. a trusted authority, \(TA\), who produces key information and securely delivers them to the required parties;
2. a group of \(n\) transmitters, \(\mathcal{T} = \{T_1, T_2, \ldots, T_n\}\) with an access structure;
3. a combiner \(C\) who can only be trusted on the designated combining operation but may collaborate with the intruders by leaking information;
4. a receiver \(R\) who receives codewords and is able to verify their authenticity.

Transmitters want to send a source state \(s \in S\) to the receiver over a public channel. A transmitter has an A-code, called a component A-code, which is used to generate a partial codeword to be sent to the combiner, and the receiver has an A-code, called the channel code which is used for verifying authenticity of a received codeword. The combiner uses a public algorithm to combine the partial codewords and produce a channel codeword which is sent to the receiver. The receiver uses her A-code and a secret key received from the TA to verify the received codeword. All communications, between transmitters and the combiner, and between the combiner and the receiver are over public channels and subject to spoofing.

Let \(G \subseteq 2^\mathcal{T}\) denote the collection of subsets of transmitters that are authorised to authenticate source states. \(G\) is called the access structure of the SGA-system and \(X \in G\) is called an access set. A \((t, n)\) threshold SGA-system is an SGA-system where a \(X \subset \mathcal{T}\) is an access set if and only if \(|X| \geq t\). Transmitters in a non-access set may collude to construct a fraudulent codeword. Although an outsider may attack the system too, but it is sufficient to consider attacks from colluding groups of transmitters as they have access to some key information.

Let \(S\) be the set of source states. An SGA-system consists of three phases.
1. **Key distribution:** The TA generates and distributes keys to the receiver and each transmitter. We denote the sets of keys for the receiver $R$ and transmitter $T_i$, $1 \leq i \leq n$, by $\mathcal{E}$ and $\mathcal{E}_i$, respectively. We also assume that after the key distribution $T_i$ and $R$ hold the key $e_i \in \mathcal{E}_i$ and $e \in \mathcal{E}$, respectively.

2. **Co-authentication:** Assume that a qualified group $B \in \Gamma$ wants to co-authenticate a source state $s \in \mathcal{S}$. Each transmitter in $B$ generates his partial codeword (or authenticator) $f_i(s, e_i) = m_i$ by applying his component A-code $f_i : \mathcal{S} \times \mathcal{E}_i \rightarrow \mathcal{M}_i$, and then sends $m_i$ to the combiner. The combiner then uses a publicly known combination function

$$C_B : \prod_{i \in B} \mathcal{M}_i \rightarrow \mathcal{M}$$

to output the channel codeword (or full authenticator) $C_B(\prod_{i \in B} m_i) = m$ and sends it to the receiver.

3. **Verification:** Assume that the receiver receives a codeword $m'$ from the combiner. She first calculates the codeword $m = f(s, e)$ using the channel A-code $f : \mathcal{S} \times \mathcal{E} \rightarrow \mathcal{M}$ and the key received from the TA in the key distribution phase. If $m' = m$ then she accepts it as authentic, otherwise rejects it.

In the sequel, we will use the following notations: $(\mathcal{S}, \mathcal{M}, \mathcal{E}, f)$ denotes the channel code for the receiver, $(\mathcal{S}, \mathcal{M}, \mathcal{E}_i, f_i)$ denotes the component A-code of the transmitter $T_i$, $1 \leq i \leq n$, and for a subset $L = \{i_1, \ldots, i_L\} \subseteq \{1, \ldots, n\}$, we denote $\mathcal{E}_L = \mathcal{E}_{i_1} \times \cdots \times \mathcal{E}_{i_L}$ and $T_L = \{T_{i_1}, \ldots, T_{i_L}\}$.

We will only consider impersonation and substitution attacks.

*Impersonation attack:* $T_L \notin G$, after receiving their secret keys, generate a message $m \in \mathcal{M}$, and hope that $m$ is accepted by $R$ as authentic. We denote by $P_{I}[L]$ the success probability of $T_L$ in performing an impersonation attack. This can be expressed as

$$P_{I}[L] = \max_{e_L \in \mathcal{E}_L} \max_{m \in \mathcal{M}} P(m \text{ is accepted by } R \mid e_L).$$

*Substitution attack:* $T_L \notin G$, after receiving their secret keys, observe a valid message that is transmitted to the receiver and then construct a codeword $m'$, and hope that $m'$ is accepted by $R$. Let $P_{S}[L]$ denote the success probability of $T_L$ in performing a substitution attack. We have,

$$P_{S}[L] = \max_{e_L \in \mathcal{E}_L} \max_{m \in \mathcal{M}} \max_{m' \neq m \in \mathcal{M}} P(m \text{ is accepted by } R \mid m, e_L).$$
We define enemy's best chance of success in impersonation and substitution attack, denoted by $P_I$ and $P_S$, respectively, as,

$$P_I = \max_L P_I[L] \quad \text{and} \quad P_S = \max_L P_S[L],$$

where the maximum is taken over all possible $L \notin \mathcal{G}$.

The above model and definition is identical to the one used in [73] and for threshold constructions in [91]. Both of these papers have focussed on constructions of provably SGA-systems and have not considered efficiency of such systems.

The first information theoretic bound on the performance of SGA-systems is given in [54] and assumes a slightly different model. The SGA-system, this time acronym for secure group authentication, assumes the receiver to also have the role of the TA and to generate the required keys for transmitters. We note that this difference in the model does not affect analysis of the system as in our model the receiver is also assumed trusted. However, in group authentication systems the receiver constructs shares of her secret key, to be used by transmitters, using a secret sharing scheme and the combining function is assumed to be addition. That is the combiner simply adds the partial authenticators to construct the authenticator for the channel code. This model is a special case of our model which allows a general secret sharing scheme to be used for combining partial authenticators.

For a set $\mathcal{X}$, let $X$ to denote the random variable taking values on the set $\mathcal{X}$ with respect to a probability distribution on $\mathcal{X}$. $H(X)$ denotes the Shannon entropy. The following extends Gehrmann's results [54] to our SGA model and is a generalisation of Simmons' and Brickell's bounds.

**Theorem 5.1** In a $(t, n)$ SGA-system, the following bounds hold.

(i) $P_I[L] \geq 2^{-I(M;E|E_L)}$;

(ii) $P_S[L] \geq 2^{-H(E|M,E_L)}$.

**Proof.** Consider the channel A-code, $f : S \times \mathcal{E} \rightarrow \mathcal{M}$. Simmon's information-theoretic bound gives $\tilde{P}_I \geq 2^{-T(M;E)}$, where $\tilde{P}_I$ denotes the success probability of impersonation attack of the channel A-code. Since $T_L$ has the key information $e_L \in \mathcal{E}_L$ in accordance with the key of channel A-code $e \in \mathcal{E}$. It follows that $P_I[L] \geq 2^{-I(M;E|E_L)}$. Similarly, by applying Brickell's bound for A-code, we have $P_S[L] \geq 2^{-H(E|M,E_L)}$.

Next we prove combinatorial lower bounds on the keys’ and authenticators’ length in a $(t, n)$ SGA-system and for a given deception probability.
Theorem 5.2 Let $P_D = 1/q$ be the overall deception probability of a $(t,n)$ SGA-system. Then the following statements hold

(i) $|\mathcal{E}| \geq q^2$ and $|\mathcal{M}| \geq q|\mathcal{S}|$;

(ii) If $|\mathcal{E}| = q^2$, then $|\mathcal{E}_i| \geq q^2$.

Proof. (i) We denote the overall deception probability of the channel A-code by $\tilde{P}_D$. Clearly, $P_D = 1/q \geq \tilde{P}_D$. By the well-known square root bound, we have $1/q \geq \tilde{P}_D \geq \frac{1}{\sqrt{|\mathcal{E}|}}$. It follows that $|\mathcal{E}| \geq q^2$. On the other hand, it is clear that $P_D \geq \tilde{P}_I \geq |\mathcal{S}|/|\mathcal{M}|$, where $\tilde{P}_I$ denotes the probability of success in impersonation attack of the channel A-code. So $|\mathcal{M}| \geq q|\mathcal{S}|$.

(ii) We first observe that from the assumption that $P_D = 1/q$ and $|\mathcal{E}| = q^2$, we know that the channel A-code is perfect and $H(E) = 2\log q$. For any $t - 1$ subset $L$ of $\{1, \ldots, n\}$, we claim $H(E|E_L) = H(E)$. Let $P_I[L]$ denote the probability of success in impersonation attack by the $t - 1$ transmitters $T_L$. By Theorem 5.1 we know that $P_I[L] \geq 2^{-I(M;E|E_L)}$ and $P_S[L] \geq 2^{-H(E|M,E_L)}$. It follows that

$$
P_I[L]P_S[L] \geq 2^{-I(M;E|E_L)-H(E|M,E_L)} \geq 2^{-H(E|E_L)} \geq 2^{-H(E)} \geq 2^{-H(E)} \geq \frac{1}{q^2} \tag{5.1}$$

On the other hand,

$$\frac{1}{q^2} = P_D^2 \geq P_I[L]P_S[L]$$

and so all inequalities in (5.1) hold with equality and in particular $H(E|E_L) = H(E)$. Next we prove that for any $t$ subset $K$ of $\{1, \ldots, n\}$, $H(E|E_K) = 0$. Since any $t$ subset of transmitters can generate the channel codewords for any source states, it means that any $t$ transmitters, $T_K$, pooling their secret keys can uniquely determine the authentication function of the channel A-code: $f_e : \mathcal{S} \to \mathcal{M}$, which is uniquely determined by receiver’s key and so $H(E|E_K) = 0$. Finally we show that $H(E_i) \geq H(E)$ for all $1 \leq i \leq n$. Since, for any $t - 1$ subset $L$ and $i \notin L$, we have,

$$I(E;E_i|E_L) = H(E|E_L) - H(E|E_i,E_L) = H(E|E_L) - H(E|E_{\{i\}} \cup L) = H(E|E_L) = H(E).$$
On the other hand,

\[ I(E; E_i|E_L) = H(E_i|E_L) - H(E_i|E_L, E) \]

\[ \leq H(E_i|E_L) \]

\[ \leq H(E_i). \]

and so \( H(E_i) \geq H(E) \) for all \( 1 \leq i \leq n \). It follows that \( \log |E_i| \geq H(E_i) \geq H(E) = 2\log q \), and so \( |E_i| \geq q^2 \), proving the desired result.

**Corollary 5.3** Under the assumption of Theorem 5.2, if the bounds in (i), Theorem 5.2, is met with equality, then the length of the full authenticator is \( \log q \).

**Proof.** The result follows immediately by considering the channel A-code and applying the result of A-code.

We will call a \((t, n)\) SGA-system **optimal** if it meets the bounds in Theorem 5.2 with equality.

**An optimal construction** One of the first SGA-systems proposed by Desmedt, Frankel and Yung [41] used Shamir's secret sharing scheme in combination with a well-known A-code. The scheme referred to as DFY polynomial scheme, works as follows. Let \( q \) be a prime power such that \( q \geq \max\{n, |S|\} \) and \( S = GF(q) \). In the key distribution, the TA (or the receiver) chooses two random polynomials, \( P(x) \) and \( Q(x) \), over \( GF(q) \) and with degree at most \( t - 1 \), and privately sends \( P(i) \) and \( Q(i) \) to transmitter \( T_i \) for all \( 1 \leq i \leq n \), and \( P(0) \) and \( Q(0) \) to the receiver \( R \). To authenticate a message \( s \in S \), \( t \) transmitters send their partial authenticators \( A_s(i) = P(i) + sQ(i) \) and their identity \( i \) to the combiner, the combiner, using Lagrange interpolation, creates the authenticator \( A_s(0) \), which is the value of polynomial \( A_s(x) = P(x) + sQ(x) \) evaluated at 0, and send the \( (s, A_s(0)) \) to the receiver. Then the receiver \( R \) accepts \( (s, A_s(0)) \) as authentic if \( A_s(0) = P(0) + sQ(0) \). It is proved [41] that this construction results in a \((t, n)\) SGA-system with \( P_l = P_S = 1/q \). The system has the following parameters \( |S| = q \), \( |E_i| = |E| = q^2 \) and so using Theorem 5.2, is optimal. Moreover the length of the tag is \( \log q \) which is the same as the size of the \( |M| = q|S| \).

5.2 Efficient constructions for SGA-system

In an optimal scheme the size of the authenticator space is inversely proportional to the deception probability. In the case of DFY scheme, \( q \) which determines deception probability is bounded as \( q \geq \max\{n, |S|\} \) and so grows also with an increase in the
number of participants. This results in very inefficient systems when $n$ is large and $S$ is small where tags and keys are very long and $P_D$ is unnecessarily low. In this section we give two constructions that provide higher efficiency in terms of the tag and the key lengths.

A differentiating property of these constructions is the **anonymity property**. In an SGA-system two kinds of anonymity can be considered: first during construction of partial authenticators and next during submission of them. An SGA-system may require a sender to have access to the identities of other members of a collaborating group, before being able to construct his own partial authenticators. In this case collaboration for authenticating a message requires the senders' identity to be revealed to other group members. There are also systems in which this information is not necessary and a sender can maintain his anonymity when constructing his partial authenticator. A second kind of anonymity is with respect to the combiner. An SGA-system may require partial authenticators to carry the identity information of the senders for the combining operation, while in other systems combining operation does not require this information. We refer to these two kinds of anonymity by $A1$ and $A2$. It is not difficult to find scenarios that each of the above variations is necessary. For example a system that provides anonymity of type $A1$ and $A2$ can be used to implement a voting system. In such a system, the source states corresponds to possible voting outcomes and the aim of the vote is to select alternatives that have enough, at least $t$ out of $n$, support.

### 5.2.1 $(t, n)$ SGA-system based on modified den Boer A-code

Following the line of DFY construction, our first SGA-scheme can be regarded as a combination of the Shamir's secret sharing scheme with a key-efficient A-code due to den Boer [33]. An efficient $(t, t)$ SGA-system based on the $(t, t)$ threshold secret sharing scheme of Karnin et al [68] and den Boer A-code was proposed by Safavi-Naini in [91]. We extend this construction to the general $(t, n)$ SGA-system.

**Construction I**

Assume that the set of source states $S$ consists of strings of $u$ bits, where $u$ is a multiple of $h$ such that $u \leq h^2$. For each $s \in S$ we write $s = (s_0, s_1, \ldots, s_\ell), \ell \leq h - 1$, where each $s_i, 0 \leq i \leq \ell$, is a block of $h$ bits. Our scheme works as follows.

1. **Key distribution**: The TA randomly chooses two polynomials $P(x)$ and $Q(x)$ over the finite field $GF(2^h)$ of degree at most $t - 1$, and $n$ distinct non-zero elements $x_1, \ldots, x_n \in GF(2^h)$, which are public identities of the transmitters.
Then the TA privately sends \((P(x_i), Q(x_i))\) to transmitter \(T_i\) for all \(1 \leq i \leq n\), and \((P(0), Q(0))\) to the receiver \(R\).

2. **Co-authentication:** For a source state \(s \in S\), assume that a group \(B\) of \(t\) transmitters, wants to generate the authenticator for \(s\). Each transmitter \(T_i \in B\), calculates his partial authenticator,

\[
a_i = \mu_i + \sum_{j=0}^{\ell} s_j \lambda_i^{2^j},
\]

where

\[
\mu_i = P(x_i) \frac{\prod_{T_j \in B, j \neq i} (0 - x_j)}{\prod_{T_j \in B, j \neq i} (x_i - x_j)}, \quad \text{and} \quad \lambda_i = Q(x_i) \frac{\prod_{T_j \in B, j \neq i} (0 - x_j)}{\prod_{T_j \in B, j \neq i} (x_i - x_j)},
\]

and sends \(a_i\) to the combiner. The combiner computes the sum of all the received partial authenticator from \(B\),

\[
a = \sum_{i \in B} a_i.
\]

and \(m = (s, a)\) to the receiver.

3. The receiver accepts \((s, a)\) as authentic if \(a = P(0) + \sum_{j=0}^{\ell} s_j (Q(0))^{2^j}\).

**Theorem 5.4** Construction I results in a \((t, n)\) SGA-system with the following parameters

\[
|S| = 2^{(\ell+1)}, \quad |M| = 2^h |S|, \quad |\mathcal{E}| = |\mathcal{E}_i| = 2^{2^h}
\]

and the success probability in impersonation and substitution attacks is given by \(P_I = 2^{-h}\) and \(P_S = 2^{\ell-h}\).

**Proof.** Using Lagrange interpolation, we have \(P(0) = \sum_{i=1}^{\ell} \mu_i\) and \(Q(0) = \sum_{i=1}^{\ell} \lambda_i\). It follows that

\[
a = \sum_{i=1}^{\ell} (\mu_i + \sum_{j=0}^{\ell} \lambda_i^{2^j})
= P(0) + \sum_{j=0}^{\ell} s_j (\sum_{i=1}^{\ell} \lambda_i)^{2^j}
= P(0) + \sum_{j=0}^{\ell} s_j (Q(0))^{2^j}.
\]

So \(t\) out of \(n\) transmitters are able to generate an authenticated codeword. To prove that \(P_I = 2^{-h}\) and \(P_S = 2^{\ell-h}\), it is sufficient to show that for any \(L \subseteq \{1, \ldots, n\}\) with \(|L| = t - 1\), we have \(P_I[L] = 2^{-h}\) and \(P_S[L] = 2^{\ell-h}\). Without loss of generality, assume that \(L = \{1, \ldots, t-1\}\), and after key distribution, \(T_L\) hold the keys \((e_1, \ldots, e_{t-1}) \in \mathcal{E}_1 \times \cdots \times \mathcal{E}_{t-1}\). Let \(F = \{e \in \mathcal{E}; \text{ given } (e_1, \ldots, e_{t-1})\}\). That is, \(F\) is the set of all
possible authentication keys of the receiver $R$ in accordance with the given keys of $T_L$. According to Shamir secret sharing scheme we know that $(P(0), Q(0))$ is independent of $((P(x_1), Q(x_1)), \ldots , (P(x_{t-1}), Q(x_{t-1})))$, it follows that $\mathbf{F} = \mathcal{E}$, and so $|\mathbf{F}| = |\mathcal{E}| = 2^{2h}$.

In the impersonation attack, $T_L$ generates a codeword $(s, a), s \in S$ and $a \in GF(2^h)$, and hopes that it will be accepted by $R$ as authentic. Denote $E(s,a) = \{e \in \mathbf{F}; e(s) = a\}$. That is $E(s,a)$ is the set of the solutions of the following equation

$$P(0) + \sum_{i=0}^{t} s_i Q(0) 2^i = a.$$ 

Using an argument identical to [33], we have $|E(s,a)| = 2^h$. It follows that

$$P_I[L] = \max_{s \in S, a \in GF(q)} \frac{|E(s,a)|}{|\mathbf{F}|} = \frac{2^h}{2^{2h}} = 2^{-h}.$$ 

In the substitution attack, $T_L$, after seeing a valid codeword $(s, a)$, generates a new codeword $(s', a'), s' \neq s$, and hope that $(s', a')$ will be accepted by $R$ as authentic. Denote $E(s,a, s'.a') = \{e \in E(s,a); e(s') = a'\}$. We claim that $|E(s,a, s'.a')| = 2^\ell$. Indeed, the numerator gives the equation

$$\sum_{i=0}^{t} (s_i - s'_i)(Q(0)) 2^i = a - a',$$

which has at most $2^\ell$ solutions (values of $Q(0)$), Using $2^h$ for the denominator, we have $|E(s,a, s'.a')| = 2^\ell$. It follows,

$$P_S[L] = \max_{s \neq s' \in S} \frac{|E(s,a, s'.a')|}{|E(s,a)|} = \frac{2^\ell}{2^h} = 2^{\ell-h}.$$ 

Similarly, we have $P_I[L] = 2^{-h}$ and $P_S[L] = 2^{\ell-h}$ for any $t-1$ subset $L$ of $\{1, \ldots, n\}$. Thus we have proved the desired result. The proof of the cardinality parameters are obvious.

We note that in this construction $P_I$ is only determined by the size of the finite field $GF(2^h)$ but $P_S$ rapidly increases with the increase in the number of the source states. For $\ell = 0$, the construction results in the DFY scheme. We also observe that the combining operation for the combiner is the sum of $t$ partial authenticators, and so satisfies A1 but does not satisfy A2.

### 5.2.2 $(t, n)$ SGA-systems from linear error-correcting codes

In this subsection, we will give a construction from linear error-correcting code. The basic idea behind this construction is to use vectors of the dual code for co-authentication
and verification process. Let \( S \subseteq GF(q)^{\ell} \) denote the set of source states obtained by defining the following equivalence relation \( \sim \) over \( GF(q)^{\ell} \): \( s \) is equivalent to \( s' \), \( s \sim s' \), if and only if \( s = rs' \) for some \( 0 \neq r \in GF(q) \). It is easy to see that this relation is in fact an equivalence relation. We define \( S \) as the set of equivalence classes obtained from \( \sim \). It follows that \( |S| = \frac{q^{\ell}-1}{q-1} = q^{\ell-1} + \cdots + q + 1 \).

Construction II

1. **Key distribution:** The TA randomly chooses an \( \ell \times t \) matrix \( U \in GF(q)^{\ell \times t} \). Assume that \( q \geq n + 1 \) (this assumption is not necessary\(^1\)). The TA also chooses \( n \) distinct non-zero elements \( x_1, \ldots, x_n \in GF(q) \) (these elements are public and are used as the identities of the transmitters), then calculates and secretly sends \( U(1, x_i, \ldots, x_i^{t-1})^T = \alpha_i \in GF(q)^{\ell} \) to \( T_i \), for all \( 1 \leq i \leq n \), and \( U(1, 0, \ldots, 0)^T = \alpha \), to the receiver \( R \), which consists of the secret keys of \( T_i, 1 \leq i \leq n \), and \( R \), respectively.

2. **Co-authentication:** Assume that a group of \( t \) transmitters \( B = \{ T_{k_1}, \ldots, T_{k_t} \} \) want to co-authenticate a source state \( s \in S \). Each transmitter \( T_i \) in \( B \) sends their partial authenticator \( a_i = s \alpha_i \in GF(q) \) and their identities \( x_i \) to the combiner, who then computes the authenticator

\[
a = (a_{k_1}, a_{k_2}, \ldots, a_{k_t}) \begin{pmatrix}
1 & 1 & \cdots & 1 \\
x_{k_1} & x_{k_2} & \cdots & x_{k_t} \\
\vdots & \vdots & \ddots & \vdots \\
x_{k_1}^{t-1} & x_{k_2}^{t-1} & \cdots & x_{k_t}^{t-1}
\end{pmatrix}^{-1} \begin{pmatrix}
1 \\
0 \\
\vdots \\
0
\end{pmatrix},
\]

and send \((s, a)\) to the receiver \( R \).

3. **Verification:** The receiver accepts \((s, a)\) as authentic if \( a = s \alpha \).

**Theorem 5.5** Construction II results in a \((t, n)\) SGA-system with \( P_I = P_S = 1/q \). It has the following parameters,

\[|S| = \frac{q^{\ell}-1}{q-1}, \quad |E_i| = |E| = q^\ell, \quad \text{and} \quad |M| = q|S|.|}

**Proof.** It is easy to see that any \( t \) out of \( n \) transmitters are able to generate a valid authenticated codeword. We prove that \( P_I = P_S = 1/q \). Assume that \( t - 1 \) transmitters \( T_L = \{ T_1, \ldots, T_{t-1} \} \) want to perform an impersonation attack,

\(^1\)Instead, it may choose a \( t \times (n + 1) \) matrix \( M = (M_0, M_1, \ldots, M_n) \) over \( GF(q) \) such that any \( t \) columns of \( M \) are linearly independent, the secret key of \( T_i \) is \( UM_i \) and the key of \( R \) is \( UM_0 \).
and after the key distribution, $T_i$ holds the key $\alpha_i, 1 \leq i \leq t - 1$. Let $F = \{U \in GF(q)^{t \times w}; U(1, x_1, \ldots, x_{t-1})^T = \alpha_1, \ldots, U(1, x_{t-1}, \ldots, x_1)^T = \alpha_{t-1}\}$. That is, $F$ is the set of possible matrices that the TA may choose in accordance with the keys of $T_L$. We define a mapping $\phi : F \rightarrow E$ by

$$\phi(U) = U(1, 0, \ldots, 0)^T, \quad \forall U \in F.$$  

It is straightforward to verify that $\phi$ is one-to-one from $F$ onto $E$. It also implies that $R$'s key is independent of keys of $T_L$. In the impersonation attack, $T_L$, generates a codeword $(s, a), s \in S$ and $a \in GF(q)$, and hope that it will be accepted by $R$ as authentic. It follows,

$$P_I[L] = \max_{(s, a) \in S \times GF(q)} \frac{|\{e \in E; e(s) = a\}|}{|E|} = \frac{q^{t-1}}{q} = \frac{1}{q}.$$  

Similarly, in the substitution attack $T_L$, after seeing a valid code $(s, a)$, generate a new codeword $(s', a')$, $s' \neq s$. We have

$$P_S[L] = \max_{(s, a), (s', a') \in S \times GF(q), s \neq s'} \frac{|\{e \in E; e(s) = a, e(s') = a'\}|}{|\{e \in E; e(s) = a\}|} = \frac{q^{t-2}}{q^{t-1}}.$$  

Clearly the cardinality parameters are obvious, which proves the desired result.

Construction II satisfies A2 but does not satisfy A1. Compared to Construction I, the length of authenticator for a codeword of the channel code is very short. However $|E|$ and $|E_i|$ must be at least equal to $|S|$. This means that the key lengths are at least half the key lengths in DFY scheme.

The following corollary is straightforward.

**Corollary 5.6** For $\ell = 2$, Construction II results in an optimal $(t, n)$ SGA-system.

### 5.3 Recursive Constructions

In this section, we introduce a new construction method for threshold SGA-systems. The basic idea is to build a $(t, n)$ threshold SGA-system from a $(t, m)$ SGA-system, where $n > m$, by using a perfect hash family. The construction method can be repeatedly used and so it has the recursive nature.

Recall that an $(n, m, t)$-perfect hash family is a set of functions $F$ such that

$$f : \{1, \ldots, n\} \rightarrow \{1, \ldots, m\}$$

for each $f \in F$, and for any $X \subseteq \{1, \ldots, n\}$ such that $|X| = t$, there exists at least one $f \in F$ such that $f|_X$ is one-to-one. We use the notation $PHF(N; n, m, t)$ for an
(n, m, t)-perfect hash family with |F| = N. When m = t, PHF(N; n, t, t) is called a minimal perfect hashing family.

We construct a (t, n) threshold SGA-system by combining multiple independently generated instances of a (t, m) SGA-system, using a (n, m, t)—perfect hash family. In particular, we choose the underlying threshold SGA-system to be the (t, t) threshold SGA-system proposed in [91], and the perfect hash family is minimal.

Construction III

Assume that the set of source states S consists of strings of u bits, where u is a multiple of h such that u ≤ u^2. For each s ∈ S we write s = (s_0, s_1, ..., s_ℓ), ℓ ≤ h − 1, where each s_i, 0 ≤ i ≤ ℓ, is a block of h bits. Let F = {f_1, f_2, ..., f_N} be a minimal PHF(N; n, t, t) from {1, 2, ..., n} to {1, 2, ..., t}. The three phases of a SGA-system are as follows:

1. Key distribution: The TA randomly chooses two numbers µ, λ in the finite field GF(2^h), where (µ, λ) consists of the key of the receiver R. Then the TA executes a total independent N times (t, t) Karnin-Greene-Hellman secret sharing scheme for the same secret (µ, λ), to produce elements c^1, ..., c^N ∈ (GF(2^h) × GF(2^h))^t. For each j ∈ {1, ..., N}, we write c^j = (c_{1,j}, ..., c_{t,j}), where c_{k,j} = (µ_{k,j}, λ_{k,j}) ∈ GF(2^h) × GF(2^h), 1 ≤ k ≤ t. Thus we have \( \sum_{k=1}^{t} \mu_{k,j} = \mu \) and \( \sum_{k=1}^{t} \lambda_{k,j} = \lambda \) for all 1 ≤ j ≤ N. Then the TA privately sends to T_i the secret key \((d_{i,1}, ..., d_{i,N})\), where

\[
d_{i,j} = c_{f_j(i),j} = (\mu_{f_j(i),j}, \lambda_{f_j(i),j})
\]

for all 1 ≤ i ≤ n and 1 ≤ j ≤ N.

2. Co-authentication: Assume that a group of transmitters B wants to authenticate a source state s = (s_0, s_1, ..., s_ℓ), each transmitter T_i in B computes their partial authenticator \(a_i = (a_{i,1}, ..., a_{i,N})\), where

\[
a_{i,j} = \mu_{f_j(i),j} + \sum_{k=0}^{t} S_k \lambda_{f_j(i),j}^{2^k}
\]

for all 1 ≤ j ≤ N. Then T_i sends \(a_i\) and his identity i to the combiner. After receiving all the partial authenticator and the identities of B, the combiner uses a perfect hashing function \(f_b \in F\) on B (that is \(f_b\) is one-to-one on B) to calculate the authenticator a by

\[
a = \sum_{T_i \in B} a_{i,b}
\]

and sends the authenticated codeword \((s, a)\) to the receiver R.
3. **Verification** The receiver $R$ accepts $(s, a)$ as authentic if

$$a = \mu + \sum_{k=0}^{\ell} s_k \lambda^{2k}.$$  

**Theorem 5.7** Construction III results in a $(t, n)$ SGA-system with $P_I = 2^{-h}$ and $P_S = 2^{t-h}$. It has the following parameters

$$|S| = 2^{h(t+1)}, \quad |M| = 2^h |S|, \quad |S_i| = 2^{2hN}, \text{ and } |E| = 2^{2h}.$$  

**Proof.** In the key distribution, due to the independence of the $N$ times of $(t, t)$ secret sharing scheme and the properties of the perfect hashing family, the secret information of the transmitters consists of a $(t, n)$ perfect secret sharing scheme. It is easy to verify that any $t$ out of $n$ transmitters can generate a valid authenticated codeword. We are left to show that $P_I = 2^{-h}$ and $P_S = 2^{t-h}$. We may regard the system consisting of $N$ independently generated instances of Safavi-Naini’s $(t, t)$ SGA-systems, $\text{SGA}_1, \ldots, \text{SGA}_N$. For any group of transmitters $L$ with $|L| = t - 1$ and each $(t, t)$ SGA-system $\text{SGA}_i$, we use $P^{\text{SGA}_i}_I[L]$ and $P^{\text{SGA}_i}_S[L]$ to denote the success probabilities of impersonation and substitution attacks for $L$ in $\text{SGA}_i$. From [91] we know that $P^{\text{SGA}_i}_I[L] = 2^{-h}$ and $P^{\text{SGA}_i}_S[L] = 2^{\ell-h}$. The channel codeword $(s, a)$ is one chosen from the $N$ channel codewords of the $N$ independent $(t, t)$ SGA-system and so it follows that

$$P_I = \max_L \max_{\text{SGA}_i} \{ P^{\text{SGA}_i}_I[L] \} \quad \text{and} \quad P_S = \max_L \max_{\text{SGA}_i} \{ P^{\text{SGA}_i}_S[L] \}$$

where $L$ runs through all the $t-1$ subset of $\{1, \ldots, n\}$, and $\text{SGA}_i$ runs over the $N$ $(t, t)$ SGA-systems. Thus we immediately have $P_I = 2^{-h}$ and $P_S = 2^{\ell-h}$. The cardinality parameters are obvious, proving the desired results.

It can be seen that Construction III satisfies A2 but not A1. This is in contrast to Construction I which satisfies A1 but not A2.

Now we analyse the efficiency of Construction III. Firstly, the key length for the receiver and the authenticator length are the same with Construction I, while the size of key for each transmitter has increased $N$ times (from $2h$ bits to $2hN$ bits). Thus the key storage for the transmitters of the resulting system depends on the size of the hashing family. It is important to know the minimum value of $N$, denoted by $N(n, m, t)$, for given $n$, $m$ and $t$ and such that a $\text{PHF}(N; n, m, t)$ exists. It is well known that $[3, 76]$ $N(n, m, t)$ is $\Theta(\log n)$ but the existence result is non-constructive and it is believed that a construction with good asymptotic performance is difficult. In [3], the authors give several constructions from known combinatorial objects, in which $N$ is a polynomial function of $\log n$ (for fixed $m$, and $t$).
An important feature of this constructing is that it results in particularly efficient systems when the size of the sender group is large but the size of the source space is small. We noted that in DFY polynomial construction or our proposed error-correcting code construction, the finite field \( GF(q) \) must be chosen such that \( q \geq \max\{n, |S|\} \), and the deception probability, the size of the keys and the length of the authenticator are all determined by \( q \). Although it is acceptable to have the key storage and the length of authenticator as a function of the probability of success, having the number of senders and the size of the source bound by this probability is not reasonable. We give an example which shows how to overcome this shortcoming of the optimal construction by using our recursive construction method.

**Example 5.1** Assume that the size of source is very small, say one bit (that is \( |S| = 2 \)), we want to have a \((t,n)\) threshold SGA-system with \( n \gg t \geq 2 \). Using the known optimal construction, the finite field \( GF(q) \) must be chosen such that \( q \geq n \) and we have \( P_D \leq 1/n, |E_i| = |E| \geq n^2 \) and \( |M| = |S|q \geq 2n \). Now suppose that we are satisfied with \( P_D > 1/2 \), without requiring the lower deception probability \( P_D = 1/q \), we would expect to save some key sizes and authenticator length by increasing the deception probability of the straightforward optimal construction. This can be achieved by applying our construction method. We first choose the first prime power \( p \) such that \( p > t \), then we construct a \((t,p)\) SGA-system \((S,E_i^0,E^0,M^0)\), using DFY optimal construction over \( GF(p) \). It has the following parameters: \( |S| = 2, |E_i^0| = |E^0| = p^2, |M^0| = 2p \) with the deception probability \( P_D^0 = 1/p \). Next, we use the construction method identical to Construction III to obtain a \((t,n)\) with the underlying \((t,p)\) SGA-system and a \((N;n,p,t)\) perfect hash family. The resulting \((t,n)\) threshold SGA-system \((S,E_i,E,M)\) has the following parameters: \( |S| = 2, |E_i| = p^{2N}, |E| = p^2, |M| = 2p \) and the deception probability \( P_D \leq 1/p \). While \( n \) is much larger than \( p \), both the key size of the receiver and the length of the authenticator, even the key size of each transmitter can be significantly reduced.

We note that the method of Construction III can be repeatedly applied, and so it is recursive.

### 5.4 Robust SGA-systems

In this section, we consider a different type of attack in SGA-systems. In authentication attacks the aim of the colluders is to construct a fraudulent codeword acceptable by the
receiver and security of the systems is measured with respect to this attack. However a system can be targeted by the attacker and be made largely 'unavailable' which in the context of authentication means the majority of messages received by the receiver are rejected. Without loss of generality we assume that intruders are a colluding group of senders who use their key information this time to disable correct functioning of the combiner. Their attack consist of injecting partial authenticators or substituting a partial authenticator with a fraudulent one. The attackers are successful if the combiner accepts the set of partial authenticators and constructs the full authenticator based on them which with a very high chance result in the acceptance of the codeword by the receiver.

A $(t, n)$ threshold SGA-system is called $(\ell, n)$ robust if the chance of incorrectly calculating the full authenticator in the presence of up to $(\ell - 1)$ malicious senders is not bigger than $P_D$. We are mainly interested in the case that $\ell = t$ and call such systems simply robust. This is reasonable as $\ell < t$ means that colluding senders have a better chance of success in robustness attack and $\ell > t$ means authentication attack can give them a better success chance and in both cases it is enough to consider only that particular type of attack.

A similar concept is studied in the context of threshold signature schemes the main difference being that in a robust SGA-system we do not assume an authenticated channel between each sender and the combiner while in robust threshold signature schemes this assumption is made. Another related problem is secret reconstruction in a secret sharing schemes in the presence of malicious shareholders. However in this case the only type of attack corresponds to impersonation attack- that is a shareholder submitting a wrong share without seeing any other communication.

To construct a robust SGA-system, the combiner must be able to verify the correctness of the received partial authenticators. We require that this verification not affect the chance of success in authentication attacks. This means that verification ability of the combiner must not rely on the authentication key as otherwise combiner's collusion with the senders must be taken into account. From this point of view robustness is an independent property from authentication and can be achieved using unconditional security approach or using computationally secure systems.

We propose two constructions for robust SGA-systems where in both cases robustness in the SGA-system is achieved by adding robustness to the underlying secret sharing scheme. In the first construction the combiner is given some secret information
in the key distribution phase which is used to verify the integrity of the partial authenticators submitted from senders. We call the system Combiner Verifiable Scheme (CVS). In the second construction a commitment scheme proposed by Pedersen [81] is used in the key distribution phase. This allows public verification of the partial authenticators in the authentication phase. We call this system Public Verification Scheme (PVS). Both schemes provide unconditional security with respect to authentication attacks but different levels of security against robustness attack. In particular, the robustness CVS guarantees robustness in information-theoretic sense while PVS robustness relies on the difficulty of discrete logarithm problem.

### 5.4.1 Combiner Verifiable Scheme (CVS)

In the basic model of SGA-systems, the combiner does not participate in the key distribution phase. That is the combiner has no information secret information. In CVS the combiner also receives some secret information from the TA, but this secret information is independent from authentication key and so is not useful in performing impersonation and substitution attacks. The underlying secret sharing scheme is similar to the unconditionally secure VSSs of BGW protocol [13].

**Construction IV**

Assume that the set of source states is \( S = GF(q) \) and \( q > n \), the construction of a \((t, n)\) SGA-system with cheater detection works as follows.

1. **Key distribution** The TA randomly choose a key \( e = (\mu, \lambda) \in GF(q) \times GF(q) \) for the receiver \( R \) and two \( t \times 2 \) matrices \( M = (a_{i,j})_{1 \leq i \leq t, 1 \leq j \leq 2} \), \( N = (b_{i,j})_{1 \leq i \leq t, 1 \leq j \leq 2} \) over \( GF(q) \) with \( a_{1,1} = \mu \) and \( b_{1,1} = \lambda \). The TA also chooses \( n \) distinct non-zero elements \( x_1, \ldots, x_n \in GF(q) \), these elements are public knowledge and used as the identities of the transmitters. To each \( T_i \), the TA privately sends their secret information \( e_i = (P(x_i, y), Q(x_i, y)), 1 \leq i \leq n \); then the TA randomly chooses an element \( 0 \neq c \in GF(q) \) and gives \( e_C = (P(x, c), Q(x, c), c) \) to the combiner as his secret information, where

\[
P(x, y) = (1, x, \ldots, x^{t-1})M \begin{pmatrix} 1 \\ y \end{pmatrix}, \quad Q(x, y) = (1, x, \ldots, x^{t-1})N \begin{pmatrix} 1 \\ y \end{pmatrix}
\]

2. **Co-authentication** Assume that a group of transmitters \( A \) with \( |A| \geq t \) wants to authenticate a source state \( s \in S \). Each \( T_i \) in \( A \) computes the partial authenticator \( a_i = H_i(y) = P(x_i, y) + sQ(x_i, y) \) and sends it together with his
identity \( x_i \) to the combiner, the combiner verifies the integrity of \( a_i \) as follows: if \( H_i(c) = P(x_i, c) + Q(x_i, c) \) then \( a_i \) is accepted as a correct partial authenticator, otherwise the combiner identifies \( T_i \) as a cheater (assume that the message from \( T_i \) to the combiner has not been tampered). After the verification process, assume that there are \( t \) correct partial authenticators \( a_{i_1} = H_{i_1}(y), \ldots, a_{i_t} = H_{i_t}(y) \) from the transmitters \( B = \{T_{i_1}, \ldots, T_{i_t}\} \subseteq A \), then the combiner, using Lagrange interpolation, computes the authenticator \( m \) by,

\[
a = \sum_{k=1}^{t} c_{ik} H_{ik}(0),
\]

where,

\[
c_{ik} = \frac{\prod_{j \in B, j \neq i_k}(0 - x_j)}{\prod_{j \in B, j \neq i_k}(x_{ik} - x_j)}.
\]

Then the combiner sends \((s, a)\) to the receiver.

3. Verification The receiver \( R \) accepts \((s, a)\) as authentic if \( a = \mu + s\lambda \).

**Theorem 5.8** Construction IV results in a \((t, n)\) SGA-system with cheater detection with \( P_t = P_S = 1/q \). It has the following parameters,

\[
|S| = q, \quad |\mathcal{M}| = q|S|, \quad |\mathcal{E}| = q^2, \quad \text{and} \quad |\mathcal{E}_c| = q^4.
\]

Moreover, the cardinality of the set of keys of the combiner is \(|\mathcal{E}_c| = q^{2t+1}\), and the probability that transmitter \( T_i \) successfully cheats the combiner is \( 1/q \).

**Proof.** The various parameters for the system are obvious. As in DFY \((t, n)\) SGA-system in [41], using Lagrange interpolation, it is easy to see that any \( t \) out of transmitters can generate the valid codeword for a source state. We are left to prove \( P_t = P_S = 1/q \). We only prove \( P_S \), the proof of \( P_t \) can be done in a similar manner. Assume that a group of \( t-1 \) transmitters, say \( L = \{T_1, \ldots, T_{t-1}\} \), after observing a valid channel codeword \((s, a)\), replace it with \((s', a')\) with \( s \neq s' \) and hope that it will be accepted by the receiver. Denote \( E((s,a)) \) the set of the receiver’s keys for which \((s, a)\) is a valid message. For a given \( e_L \in \mathcal{E}_L \), let \( E(e_L) \) be the set of possible key of \( R \) in accordance with \( e_L \), i.e. \( E(e_L) = \{e \in \mathcal{E} \mid \text{there exists a key distribution with } e_L \text{ the keys of } L \text{ and } e \text{ the key of } R\} \). Thus we have,

\[
P_S[L] = \max_{(s, a), (s', a'), e_L} \frac{|E((s,a)) \cap E((s',a')) \cap E(e_L)|}{|E((s,a)) \cap E(e_L)|}.
\]
5.4. Robust SGA-systems

We claim that \( E(e_L) = E \). Assume that the TA has chosen the polynomial \( P(x, y), Q(x, y) \), where the keys for \( R \) and each \( T_i \) are \((P(0, 0), Q(0, 0))\), and \((P(x_i, y), Q(x_i, y))\), respectively. Consider the polynomials,

\[
P'(x, y) = P(x, y) + u(x - x_1) \cdots (x - x_{t-1})(y - \alpha)
\]

where \( u \in GF(q), 0 \neq \alpha \in GF(q) \). Then we have \( P'(x_i, y) = P(x_i, y) \) for all \( 1 \leq i \leq t - 1 \), and when \( u \) run through \( GF(q) \), it will results in \( q \) different \( P'(0, 0) \), this implies that \( u \) is independent of \( P(x_1, y), \ldots, P(x_{t-1}, y) \). Similarly, \( \lambda \) is independent of \( Q(x_1, y), \ldots, Q(x_{t-1}, y) \). A direct calculation follows that,

\[
|E((s, a)) \cap E((s', a'))| = \frac{1}{q}.
\]

Assume that \( T_i \) wants to cheat by submitting wrong partial authenticator such that the combiner accepts it in the generation of authenticator. For a source state \( s \), \( T_i \) knows his partial authenticator \( P(x_i, y) + sQ(x_i, y) = uy + v \). Since \( T_i \) does not know \( c \), it follows that the probability of \( T_i \) successfully cheat the combiner is \( 1/q \).

We have assumed that the integrity verification of the partial authenticators is performed by the combiner. This can be easily generalised to the scenario that each transmitter can verify the integrity of partial authenticators of other transmitters. This is done by replacing \( P(x, y) \) and \( Q(x, y) \) of degree 1 in \( y \) with polynomials of degree \( \ell \) (with \( \ell \geq t - 1 \)) and giving the transmitters two keys, one for generation of partial authenticator and the other for verification of others' authenticators. More precisely, the TA gives \((P(x, y), Q(x, y))\) and \((P(x, c_i), Q(x, c_i), c_i)\) (where \( c_i \) is a randomly element from \( GF(q) \)) to transmitter \( T_i \), where \((P(x, y), Q(x, y))\) is used to generate the partial authenticator of \( T_i \) while \((P(x, c_i), Q(x, c_i), c_i)\) is used to verify the integrity of the partial authenticators of other transmitters. We have shown that in Construction IV a single transmitter can not succeed in cheating the combiner with a probability higher than \( 1/q \). That is, a cheating sender will be identified by the combiner with a probability at least \( 1 - 1/q \). It can be proved that if \( k \) transmitters \((k \leq n)\) collude to deceive the combiner, their success probability is at most \( 1 - (1 - 1/q)^k \). Finally, we point out that Construction I can be modified for the SGA-system with cheater detection, if the underlying Shamir secret sharing scheme is replaced by the secret sharing scheme with cheater detection as being done in Construction IV.
5.4.2 Public Verification Scheme (PVS)

Our PVS uses the commitment scheme of Pedersen’s VSSS and combines it with DFY threshold SGA-system. We briefly recall Pedersen’s commitment scheme. Let \( g \) and \( h \) be elements in \( GF(q) \) such that \( \log_g h \) is intractable and let \( g \) and \( h \) be public. The committer can commits herself to an \( s \in GF(q) \) by choosing \( r \in GF(q) \) at random and computing \( C(s, r) = g^{sr} \). Such a commitment can later be opened by revealing \( s \) and \( r \). It is shown [81] that \( C(s, r) \) reveals no information about \( s \) and that the committer cannot open a commitment to \( s \) as \( s' \), \( s' \neq s \), unless she can find \( \log_g h \).

Construction V

Assume that the set of source states is \( S = GF(q) \) and \( q > n \), our PVS for \((t, n)\) SGA-systems works as follows.

1. **Key distribution:** Assume that \((\mu, \lambda) \in GF(q) \times GF(q)\) is the authentication key of the receiver \( R \), and \( x_1, \ldots, x_n \in GF(q) \) are public identity information of the senders. The TA randomly chooses \( \alpha, \beta \in GF(q) \) and four polynomials \( F(x), G(x), \alpha(x), \beta(x) \) of degree at most \( t - 1 \)

\[
F(x) = \mu_0 + \mu_1 x + \cdots + \mu_{t-1} x^{t-1} \\
G(x) = \lambda_0 + \lambda_1 x + \cdots + \lambda_{t-1} x^{t-1} \\
\alpha(x) = \alpha_0 + \alpha_1 x + \cdots + \alpha_{t-1} x^{t-1} \\
\beta(x) = \beta_0 + \beta_1 x + \cdots + \beta_{t-1} x^{t-1}
\]

such that \( \mu = \mu_0, \lambda = \lambda_0, \alpha = \alpha_0 \) and \( \beta = \beta_0 \). The TA then send \((F(x_i), \alpha(x_i)), (G(x_i), \beta(x_i))\) secretly to \( R_i, i = 1, \ldots, n \), and broadcasts \( C(\mu_i, \alpha_i) \) and \( C(\lambda_i, \beta_i) \) for \( i = 0, 1, \ldots, n \) as described in the Pedersen’s commitment scheme.

2. **Co-authentication:** Assume that a group of transmitters \( A \) with \(|A| \geq t\) want to authenticate a source state \( s \in GF(q) \). Each \( T_i \) in \( A \) computes his partial authenticator \((a_i, r_i)\), where \( a_i = F(x_i) + sG(x_i) \) and \( r_i = \alpha(x_i) + s\beta(x_i) \), then sends it together with his identity \( x_i \) to the combiner. The combiner accepts \((a_i, r_i)\) as a correct partial authenticator if and only if the following holds

\[
g^{a_i r_i} = \prod_{j=0}^{t-1} (C(\mu_i, \alpha_i))^{x_i^j} (C(\lambda_i, \beta_i))^{s x_i^j}.
\]

After the verification process, if there are \( t \) correct partial authenticators \((a_{i_1}, r_{i_1}), \ldots, (a_{i_t}, r_{i_t})\) from the transmitters in \( A \), the combine can, using Lagrange interpolation, computes the authenticator \( a = F(0) + sG(0) \) as DFY scheme and sends \((s, a)\) to the receiver.
3. **Verification**: The receiver $R$ accepts $(s, a)$ as authentic if $a = \mu + s\lambda$.

**Theorem 5.9** Construction $V$ results in a $(t, n)$ threshold SGA-system with $P_I = P_S = 1/q$. It has the following parameters:

$$|S| = q, \quad |\mathcal{E}_i| = q^4, \quad |\mathcal{E}| = q^2, \quad \text{and} \quad |\mathcal{M}| = q|S|.$$ 

Moreover, it is robust for $t < n/2$ and the robustness is guaranteed based on the discrete logarithm problem.

**Proof.** As proved in [81], because of the unconditional secrecy in the Pedersen's commitment scheme, the key distribution establishes a perfect secret sharing scheme with respect to the secret $(\mu, \lambda)$. By DFY scheme, it is unconditional secure with respect to authenticity. The verification for the partial authenticators is straightforward, its security is based on the discrete problem problem as proved in [81]. Various parameters are obvious.

The following table compares the parameters of various constructions presented in this chapter with the DFY scheme.

| Constructions | $P_I$ | $P_S$ | $|S|$ | $|E_i|$ | $|E|$ | $|M|$ |
|---------------|------|------|------|------|------|------|
| DFY           | $1/q$ | $1/q$ | $q$  | $q^2$ | $q^2$ | $q^2$ |
| I             | $2^{-h}$ | $2^{\ell-h}$ | $2^{\ell+1}$ | $2^{2h}$ | $2^{2h}$ | $2^{\ell+h+1}$ |
| II            | $1/q$ | $1/q$ | $(q^\ell - 1)/(q - 1)$ | $q^\ell$ | $q^\ell$ | $q(q^\ell - 1)/(q - 1)$ |
| III           | $2^{-h}$ | $2^{\ell-h}$ | $2^{h(\ell+1)}$ | $2^{2hN}$ | $2^{2h}$ | $2^{h(\ell+2)}$ |
| IV            | $1/q$ | $1/q$ | $q$  | $q^4$ | $q^2$ | $q^2$ |
| V             | $1/q$ | $1/q$ | $q$  | $q^4$ | $q^2$ | $q^2$ |

### 5.5 Multireceivers SGA-system

In this section we introduce a new model, called **multireceivers SGA-system**, which is basically a combination of the SGA-systems and MRA-systems. We present a construction for this model which generalises both DFY schemes for multireceiver and multisender A-codes.

In a **multireceiver SGA-system**, there are four types of participants: a trusted authority TA, a set of transmitters $T = \{T_i\}$ with an access structure, a set of receivers $R = \{R_i\}$ and a combiner $C$. The system consists of three phases:

1. **Key distribution**: The TA privately transmits the key information to each participant in the system.
2. **Co-authentication and Broadcast** Transmitters in an access set generate their partial authenticators for a source and send it to the combiner. The combiner combines these partial authenticators to generate an authenticated codeword which is broadcasted to all the receivers.

3. **Verification** Each receiver can verify the authenticity of the broadcast codeword.

In such a system, attackers could be several collaborating participants of transmitters and receivers, may also include the combiner. We call such attacks **collusion attacks**. We consider a scenario that some of the transmitters and some of the receivers collaborate and share their keys, target one of other receivers for cheating. It could be possible that the collaboration participants determine the targeted receiver's key, or part of it, and then can cheat successfully.

Assume that the opponents have access to the key information of some participants $O$, where $O \subset R \cup T$, they want to cheat one of the receivers $R_i$, where $R_i \notin O$. We define the probabilities of success in the impersonation and substitution attacks as follows.

\[ P_I[i, O] = \max_{e_O, (s, a)} P((s, a) \text{ accepted by } R_i | e_O). \]
\[ P_S[i, O] = \max_{e_O, (s', a'), (s, a), s' \neq s} P((s', a') \text{ accepted by } R_i | (s, a), e_O). \]

where $e_O \in E_O$ is the key of participants in $O$.

**Definition 5.1** A $((t, n); (k, m))$ SGA-system with multireceiver with $n$ transmitters $T = \{T_1, \ldots, T_n\}$ and $m$ receivers $R = \{R_1, \ldots, R_m\}$ is a $(t, n)$ SGA-system with multireceiver satisfying $P_I[i, O] < 1$ and $P_S[i, O] < 1$ for all $O$ with $|O \cap T| < t$ and $|O \cap R| < k$.

From the definition, we know that in a $((t, n); (k, m))$ SGA-system, any $t$ out of $n$ transmitters can generate valid codeword such that every receivers can verify its authenticity, while it provides protect against from collusion attacks of up to $t - 1$ transmitters and $k - 1$ receivers. We define the probabilities of success in the impersonation and substitution attacks in a $((t, n); (k, m))$ SGA-system as follows.

\[ P_I = \max_{O, i} P_I[i, O] \quad \text{and} \quad P_S = \max_{O, i} P_S[i, O] \]

where $O$ runs over all subset of $T \cup R$ with $|O \cap T| < t$ and $|O \cap R| < k$, and $i$ runs over all $\{1, \ldots, m\}$ with $R_i \notin O$.

It is worth mentioning that SGA-system with multireceiver can always be constructed by simply using multiple multireceiver A-systems or multiple SGA-system,
however, this trivial solution suffers from the inefficiency on the key sizes of the participants and the length of authenticator. In this section, we give a construction that is much more efficient than the trivial approach.

**Construction VI**

Assume that the set of source state $S = GF(q)$. The three phases of the system are as follows.

1. **Key distribution:** Assume that $n$ distinct numbers $x_1, \ldots, x_n \in GF(q) \setminus \{0\}$ and $m$ distinct numbers $y_1, \ldots, y_m \in GF(q) \setminus \{0\}$ are public identities of transmitters $T_1, \ldots, T_n$ and receivers $R_1, \ldots, R_m$, respectively. The TA randomly chooses two $t \times k$ matrix $M$ and $N$ over $GF(q)$ and constructs two polynomials,

$$F(x, y) = (1, x, \ldots, x^{t-1})M \left( \begin{array}{c} 1 \\ y \\ \vdots \\ y^{k-1} \end{array} \right)$$

and,

$$G(x, y) = (1, x, \ldots, x^{t-1})N \left( \begin{array}{c} 1 \\ y \\ \vdots \\ y^{k-1} \end{array} \right)$$

(a) For each transmitter $T_i (1 \leq i \leq n)$, the TA privately transmits $f_i(y) = F(x_i, y)$ and $g_i(y) = G(x_i, y)$ to $T_i$.

(b) For each receiver $R_j (1 \leq j \leq m)$, the TA privately transmits two numbers $\mu_j = F(0, y_j)$ and $\lambda_j = G(0, y_j)$ to $R_j$.

2. **Co-authentication and broadcast:** Assume that $t$ transmitters $B$ want to authenticate a source state $s$, each transmitter $T_i$ in $B$ calculates its partial authenticator $f_i(y) + sg_i(y)$ and sends it, together with its identity $x_i$, to the combiner. The combiner calculates the polynomial $H(x, y) = F(x, y) + sG(x, y)$ and evaluate at $x = 0$, then broadcasts $(s, H(0, y))$ to all the receivers.

3. **Verification:** Every receiver $R_j$ can verify the authenticity of the broadcast message in the following way. $R_j$ accepts $(s, H(0, y))$ as authentic if

$$H(0, y_j) = \mu_j + s\lambda_j.$$
Theorem 5.10 Construction VI results in a \(((t, n); (k, m))\) multireceiver SGA-system with \(P_I = P_S = 1/q\). It has the following parameters

\[ |S| = q, \quad |\mathcal{E}_T| = q^{2k}, \quad |\mathcal{E}_R| = q^2 \quad \text{and} \quad |\mathcal{M}| = q^k|S|. \]

Proof. (sketch) Deriving various parameters is straightforward. It is also easy to show that any \(t\) out of \(n\) transmitters can construct an authenticated codeword for source state. We are left to prove that \(P_I = P_S = 1/q\). It is sufficient to show that for any \(O\) with \(|O \cap T| = t - 1\) and \(|O \cap R| = k - 1\), and \(P_I[i, O] = P_S[i, O] = 1/q\) for any \(i\) with \(R_i \notin O \cap R\). Without loss of generality, we assume that \(O = \{T_1, \ldots, T_{t-1}, R_1, \ldots, R_{k-1}\}\) and \(O\) holds the keys \(\{e_{T_1}, \ldots, e_{T_{k-1}}, e_{R_1}, \ldots, e_{R_{k-1}}\}\). Let \(F = \{e_{R_i} \in \mathcal{E}_R | e_O = \{e_{T_1}, \ldots, e_{T_{k-1}}, e_{R_1}, \ldots, e_{R_{k-1}}\}\}\). That is, \(F\) is the set of possible keys of receiver \(R_i\) in accordance with the given keys of \(O\). We show that \(e_{R_i}\) is independent of \(e_O\). Indeed, consider the polynomial

\[ F(x, y) = (1, x, \ldots, x^{t-1}) \mathcal{M}(1, y, \ldots, y^{k-1})^T, \]

which is chosen by the TA. Let,

\[ Q(x, y) = F(x, y) + r \prod_{\ell=1}^{t-1} (x - x_{\ell}) \prod_{j=1}^{k-1} (y - y_j), \]

where \(r \in GF(q)\). Then we have \(Q(x_\ell, y_j) = F(x_\ell, y_j)\) for all \(\ell, 1 \leq \ell \leq t - 1\) and \(Q(0, y_j) = F(0, y_j)\) for all \(j, 1 \leq j \leq k - 1\). That means that both \(Q(x, y)\) and \(F(x, y)\) will result in the same keys for \(O\). Since \(\prod_{\ell=1}^{t-1} (0 - x_{\ell}) \prod_{j=1}^{k-1} (y_i - y_j) \neq 0\), we know that \(Q(0, y_i) = F(0, y_i)\) if and only if \(r = 0\). But \(r\) can be any element of \(GF(q)\), which results in \(q\) possible values of \(\mu_i\). Similarly, it can be shown that \(\lambda_j\) can take \(q\) possible values consistent with \(e_j\). Thus we have shown that the key \(e_{R_i} = (\mu_i, \lambda_i)\) of \(R_i\) is independent of keys \(e_O\) of \(O\). That is, \(F = E_{R_i} = GF(q)^2\).

In the impersonation attack, \(O\) generate a codeword \((s, A(y))\) and hope that it will be accepted by \(R_i\) as authentic. Let,

\[ E_{R_i}(s, A(y)) = \{e_{R_i} = (\mu_i, \lambda_i) \in F | A(y_i) = \mu_i + s\lambda_i\}. \]

It is straightforward to show that for any \(s \in GF(q)\) and \(A(y) \in GF(q)[y]\) of degree at most \(k - 1\) we have \(|E_{R_i}(s, A(y))| = q\), and so

\[ P_I[O] = \max_{s \in GF(q), A(y)} \frac{|E_{R_i}(s, A(y))|}{|F|} = q/q^2 = 1/q. \]
In the substitution attack, $O$, after seeing a valid codeword $(s, A(y))$, generate a new codeword $(s', A'(y))$ with $s' \neq s$ and hope that $(s', A'(y))$ will be accepted by $R_i$ as authentic. Let,

$$E(s.A(y), s'.A'(y)) = \{e_{R_i} = (\mu_i, \lambda_i) \in F|\mu_i + s\lambda_i = A(y_i), \mu_i + s'\lambda_i = A'(y_i)\}.$$ 

Then we have for any $s' \in GF(q)$ and $A'(y)$, $|E(s.A(y), s'.A'(y))| = 1$, and so

$$P_S[O] = \max_{s,s' \in GF(q), A(y), A'(y)} \frac{|E(s.A(y), s'.A'(y))|}{|E_{R_i}(s, A(y))|} = 1/q.$$ 

Thus we have proved the desired result.

## 5.6 Conclusions

In this chapter we proved combinatorial lower bounds and gave two key efficient constructions for SGA-systems, one based on den Boer A-codes and the second one based on error-correcting codes. We introduced a recursive construction based on perfect hash family to construct SGA-systems for large groups. We also studied the robustness of SGA-systems and gave two construction for robust SGA-systems. Finally, we proposed schemes for multireceiver SGA-systems.
Chapter 6

Multiplicative Secret Sharing

Secret sharing schemes play an important role in the constructions of shared generation of authenticators and multireceiver authentication systems. We propose a new method of construction for the multiplicative secret sharing schemes, which are interests in their own right.

6.1 Secret Sharing Schemes

Secret sharing schemes, introduced independently by Shamir [96] and Blakely [22], are one of the main building blocks of secure distributed computation. A secret sharing scheme gives a method of breaking a secret into shares that are distributed among members of a group $V$, called participants, such that only authorised subgroups of $V$ can recover the secret. To generate the shares of a secret, a dealer uses a random input to select a distribution rule that determines the share given to each participant. In a perfect secret sharing scheme unauthorised subgroups do not learn anything about the secret. In a $(t,n)$ threshold scheme, any group of at least $t$ out of $n$ users can recover the secret.

An important efficiency measure of a secret sharing scheme is the size of share. In a perfect scheme the size of a share is at least equal to the size of the secret. Systems that achieve this lower bound are called ideal. A second efficiency measure is the randomness required by the dealer, measured by the randomness coefficient [25], which determines the number of random bits for each bit of secret. Bounds on randomness coefficient for special access structures are known but little is known in the case of general access structures. As noted in [25] because generation of true randomness is expensive the amount of randomness required by a system to achieve certain level of performance is an important efficiency consideration.

Most constructions of secret sharing schemes, including Shamir's and Blakely's original constructions, require the secret and the shares to belong to a finite field. This is a
very restrictive condition and greatly limits applications of secret sharing schemes. For example threshold generation of signature in RSA, or shared proof of knowledge for graph isomorphism require secret sets and share sets to be ring and group respectively. Constructing secret sharing schemes that allow the secret and share set to be less structured has been extensively studied. Two important approaches have been homomorphic schemes of Benaloh [12] and multiplicative schemes of Desmedt and Frankel [40]. In a homomorphic scheme secret set and share set both have group structure and a homomorphic mapping between $n$-tuples over the share sets and elements of the secret set allows recovery of the secret. Multiplicative schemes require even less structure on the underlying sets: they only assume a group structure on the secret set. Each participant in an authorised group of participants can apply a function on its secret share. The combination of those values using the group operation results in the recovery of the secret. A special case of a multiplicative secret sharing scheme is a linear one. The aforementioned functions on the shares are then homomorphisms from the share space to the secret key space. Linear secret sharing schemes were introduced by Karnin-Greene-Hellman (for a finite field case) [68].

Shamir’s threshold scheme is a linear scheme. Desmedt and Frankel [40] constructed a $(t,n)$ multiplicative scheme where secret belongs to $Z_{\phi(N)}$ and $N$ is an RSA modulus, by generalising Shamir’s construction to polynomial over $Z_{\phi(N)}$, with the motivation of constructing a threshold RSA signature scheme. The main drawback of the scheme was that each participant’s share was $n$ times the size of the secret. That is the construction had a share expansion, defined as the ratio of the share size to the secret size, between $n$ and $2n$, and so was impractical for many situations. The multiplicative property of secret sharing scheme were extensively used in applications such as threshold cryptography (also called function sharing [34]). Homomorphic secret sharing schemes play an important role in voting schemes and in proactive secret sharing [79, 60]. Linear secret sharing schemes are important to achieve proactive threshold cryptography (see e.g., [51, 50]).

### 6.1.1 Constructing new threshold schemes from old ones

One of the important approaches to constructing multiplicative (homomorphic, linear) schemes has been to use existing multiplicative (homomorphic, linear) schemes to construct new ones. In [38] a recursive method for constructing new secret sharing schemes from old ones was given. One attraction of this construction was that it allowed new multiplicative schemes with desirable parameters be constructed from some existing
ones when the secret belongs to an arbitrary group. This construction in general is impractical as it requires many secret sharing schemes with lower threshold and smaller group size to exist, and results in schemes with a very high share expansion in the general case. However for \( t = 2 \) and \( t = n - 1 \) the share expansion is equal to \( \log n \) which is much better than the scheme in [40].

A major improvement on the above construction is a recursive construction due to Blackburn, Burmester, Desmedt and Wild [20]( referred to as BBDW for short), who used multiple instances of a \((t, m)\) secret sharing scheme to construct shares for the same secret for a \((t, n)\) schemes where \( n > m \). Later it was noted [19] that the share generation for the new scheme is in fact through application of a prefect hash family. Two important aspects of this construction are relatively small share expansion, \( O(\log n) \), and independence from the algebraic structure of the secret and share sets. Compared to the recursive construction in [38], BBDW construction guarantees logarithmic share expansion for any size threshold. The construction can be used to obtain efficient multiplicative and homomorphic schemes for large groups. In particular, since by using Karnin-Green-Hellman [68], one can always construct a \((t, t)\) scheme, constructing multiplicative \((t, n)\) schemes effectively reduces to the construction of a perfect hash family with the right parameters, as share expansions of order \( O(\log n) \) can only be obtained if a perfect hash family with the right set of parameters exists.

A disadvantage of this recursive method is that many instances of the original scheme, \( O(\log n) \) to be precise, is required. Since to produce each instance the dealer has to use a new randomly selected share distribution rule, the construction requires many random bits to be used and so randomness coefficient of the new scheme is \( O(\log n) \) times the randomness coefficient of the old scheme. Another drawback of BBDW approach is that in some cryptographic settings, such as in mutually trusted authority free (MTA-free) environments, it might be difficult or even impossible to execute multiple instances of the secret sharing scheme with a common secret.

### 6.1.2 Our results

We introduce a new approach to constructing new threshold schemes from old ones that has all the advantages of BBDW recursive construction and results in a new scheme whose randomness coefficient is the same as the old scheme. The basic idea is to start with the set of shares generated for an instance of the original scheme and then form the shares of the new scheme as subsets of the shares of the old one. To form the subsets, we introduce a new combinatorial structure, called a strong cover free family.
Cover-free families are extensively studied by various authors [47],[111] and have found numerous applications in cryptography. Strong cover-free families are a special case of cover-free families that can be constructed in a number of ways and in particular using universal classes of hash functions and error correcting codes. Distributing shares to the new participants according to the blocks of a strong cover-free family results in a threshold scheme which inherits security and structural properties of the original scheme, that is perfectness, homomorphic and multiplicative property, with the same randomness of the original scheme.

Our approach, although produces similar results to BBDW, is quite different in nature. In BBDW scheme the aim is to construct a \((t,n)\) scheme by using many instances of a \((t,m)\) scheme, \(m \ll n\). The shares of the new scheme is a vector of subshares and for a \(t\)-subset the collection of subshares in a particular component of the vector allows recovery of the secret. In our scheme again the shares in the new scheme is a vector of subshares but construction of the secret for a particular \(t\)-subset requires all of their subshares. In both cases the secret remains the same. In BBDW the randomness coefficient of the new scheme is a multiple of that of the old one, but in our approach it stays the same. However since the old scheme in the two methods are chosen with different parameters, and because little is known about the randomness coefficients of multiplicative secret sharing schemes over finite groups, it is not straightforward to have a general quantitative comparison of randomness coefficients between the two approaches.

We then introduce a new class of authentication codes, called linear A-code and show how to construct a SGA-system based on a multiplicative secret sharing scheme and a linear A-code.

This chapter is organised as follows. In Section 6.2 we give preliminary definitions and then recall the BBDW scheme. In Section 6.3 we introduce our construction which relies on a new type of combinatorial structure, SCFF, for which we give bounds on parameters and constructions in Section 6.4 and 6.5. In Section 6.6 we evaluate the new approach and propose directions for future research.

## 6.2 Preliminaries

Let \(\mathcal{P} = \{P_1, \ldots, P_n\}\) be a group of \(n\) participants and let \(\mathcal{K}\) denote the set of secrets. We assume \(P_i\)'s share is selected from the set \(S_i\). A \((t,n)\)-threshold scheme is a pair of algorithms: the dealer algorithm and the combiner algorithm. For a secret from \(\mathcal{K}\) and
6.2. Preliminaries

a randomly chosen element of $\mathcal{R}$, the dealer algorithm applies the mapping

$$\mathcal{D} : \mathcal{K} \times \mathcal{R} \to S_1 \times \ldots \times S_n$$

to assign shares to participants in $\mathcal{P}$. The combiner algorithm takes the shares of a subset $B \subseteq \mathcal{P}$ of participants and returns the secret, if the set $B \subseteq \mathcal{P}$ and $|B| \geq t$, or it fails.

$$C : \bigcup_{P_i \in A} \{S_i\} \to \mathcal{K} \cup \{\text{fail}\}.$$  

The classic construction of a $(t, n)$ secret sharing scheme is Shamir's scheme [96], which uses Lagrange interpolation on polynomials over finite fields. It has $\mathcal{S} = S_i = GF(q)$ and works as follows. To construct a $(t, n)$ threshold scheme $(n < q)$ protecting $k \in \mathcal{K}$, let $\mathcal{R} = GF(q)^{t-1}$ and randomly choose $n$ distinct non-zero values $x_1, \ldots, x_n \in GF(q)$. $\mathcal{D}$ is defined by

$$\mathcal{D}(k, r_1, \ldots, r_{t-1}) = (f(x_1), \ldots, f(x_n)),$$

where $f(x_i) = k + r_1 x_i + r_2 x_i^2 + \cdots + r_{t-1} x_i^{t-1}$. The values $x_i$ are made public ($x_i$ is associated with $P_i$). The function $C$ takes as input at least $t$ valid shares and uses Lagrange interpolation formula to compute $f(x)$ as

$$f(x) = \sum_{i \in B} f(x_i) \prod_{j \in B, j \neq i} \frac{(x - x_j)}{(x_i - x_j)}$$

where $B \subseteq \{1, \ldots, n\}$ and $|B| = t$, hence reconstructing the secret $k = f(0)$.

**Definition 6.1** Let $(\mathcal{D}, C)$ be a $(t, n)$ threshold scheme for which the key space $\mathcal{K}$ is a finite group with respect to the operation "\(*". The scheme $(\mathcal{D}, C)$ is multiplicative over $(\mathcal{K}, \ast)$ if for all sets $B = \{i_1, i_2, \ldots, i_t\}$ of $t$ distinct participants there exists a family $f_{i_1,B}, f_{i_2,B}, \ldots, f_{i_t,B}$ of functions from $S_{i_1}, S_{i_2}, \ldots, S_{i_t}$ to $\mathcal{K}$ and a public ordering $i_1, \ldots, i_t$ of the elements of $B$ with the following property. For all key $k \in \mathcal{K}$ and shares $s_{i_1}, \ldots, s_{i_t}$ that have been distributed to $B$ by algorithm $\mathcal{D}$ on input $k$, we may express $k$ in the form:

$$k = f_{i_1,B}(s_{i_1}) \ast f_{i_2,B}(s_{i_2}) \ast \cdots \ast f_{i_t,B}(s_{i_t}).$$

It should be noted that multiplicative schemes only require a group structure on the set of keys $\mathcal{K}$, which is different from homomorphic schemes [12]. However, when the functions $f_{i,B}$ are homomorphic, then the secret sharing scheme is called linear. Shamir's scheme over the finite field $Z_q$ is a linear scheme, where $f_{i,B}(s_{i})$ is defined by

$$f_{i,B}(s_{i}) = f(x_i) \prod_{j \in B, j \neq i} \frac{-x_j}{(x_i - x_j)}.$$
One measure for efficiency of the secret sharing scheme can be through the notion of share expansion.

**Definition 6.2** Under the above notations, we define the share expansion of a secret share scheme as

\[
\rho = \max_{1 \leq i \leq n} \frac{\log |S_i|}{|K|}.
\]

Throughout this chapter, all logarithms are to the base 2, unless otherwise indicated.

### 6.2.1 Blackburn-Burmester-Desmedt-Wild’s scheme

The most efficient construction of multiplicative secret sharing schemes has been through an elegant recursive construction due to Blackburn, Burmester, Desmedt and Wild [20]. It uses a perfect hash family and multiple instances of a \((t, m)\) scheme to build a new \((t, n)\) scheme, where \(n > m\). The original description of the scheme was without any reference to perfect hash families. Later Blackburn [19] gave a general description of the scheme using perfect hash families. In the remainder of this section we briefly review BBDW scheme as it will serve as a benchmark to measure performance of our scheme.

A \((n, m, w)\)-perfect hash family is a set of functions \(\mathbf{F}\) such that \(f : \{1, \ldots, n\} \rightarrow \{1, \ldots, m\}\) for each \(f \in \mathbf{F}\), and for any \(X \subseteq \{1, \ldots, n\}\) such that \(|X| = w\), there exists at least a function \(f^X\) in \(\mathbf{F}\) such that \(f^X\) is injection on \(X\), i.e. the restriction of \(f^X\) on \(X\) is one-to-one. We will use the notation \(\text{PHF}(N; n, m, w)\) for a \((n, m, w)\) perfect hash family with \(|\mathbf{F}| = N\). Let \(N(n, m, w)\) denote the minimum value \(N\) such that a \(\text{PHF}(N; n, m, w)\) exists. From [76], we know that for fixed \(m\) and \(w\), \(N(n, m, w)\) is \(\Theta(\log n)\).

BBDW construction uses a perfect hash family to combine multiple instances of an existing threshold scheme to obtain a new threshold scheme with larger number of participants while retaining the same threshold, and with relatively small share expansion. Let \((D_0, C_0)\) be a \((t, m)\) secret sharing scheme and and let \(\mathbf{F} = \{f_1, \ldots, f_N\}\) be a \(\text{PHF}(N; n, m, t)\). BBDW scheme works as follows.

1. To share a secret \(k \in \mathcal{K}\), executes the dealer algorithm \(D_0\) independently \(N\) times to produce \(c^1, \ldots, c^N \in \mathcal{S}_1 \times \cdots \times \mathcal{S}_m\), where \(\mathcal{S}_i\) is the share set of \(i\)th participant in the \((t, m)\) scheme. For each \(j \in \{1, \ldots, N\}\), write \(c^j = (c^j_1, \ldots, c^j_m)\), where \(c^j_k \in \mathcal{S}_k\) for all \(1 \leq k \leq m\).
6.3. Our approach

2. Define a new \((t,n)\) threshold scheme with \(n\) participants \(\mathcal{P} = \{P_1, \ldots, P_n\}\) by constructing the share \(d_i = (d_{i,1}, \ldots, d_{i,N})\), where \(d_{i,j} = c_{f_i(j),j}\) for all \(1 \leq i \leq n\) and \(1 \leq j \leq N\), and assigning \(d_i\) to \(P_i\).

We denote the new \((t,n)\) threshold scheme as \((\mathcal{D}, \mathcal{C})\).

**Theorem 6.1 ([20, 19])** If \((\mathcal{D}_0, \mathcal{C}_0)\) is a perfect \((t,m)\) threshold scheme with share expansion \(\rho_0\), then \((\mathcal{D}, \mathcal{C})\), constructed as above, is a perfect \((t,n)\) threshold scheme with share expansion of \(\rho_0 N\). Moreover, \((\mathcal{D}, \mathcal{C})\) is multiplicative provided \((\mathcal{D}_0, \mathcal{C}_0)\) is multiplicative.

It is known that \(PHF(N; n, m, w)\) with \(N = O(\log n)\) exists, which means that the share expansion in BBDW approach can reach \(O(\log n)\). In practice, since the existence result on PHF that satisfies the bound \(N(n, m, w)\) with equality is non-constructive, explicit constructions of \(PHF(N; n, m, w)\) with reasonable size \(N\) is of interest.

### 6.3 Our approach

We first give a formal definition of a strong cover-free family.

**Definition 6.3** A strong cover-free family is a set system \((X, \mathcal{B})\) such that the following properties are satisfied:

1. \(X = \{x_1, \ldots, x_v\}\) called points;
2. \(\mathcal{B} = \{B_1, \ldots, B_n\}\) is a set of \(n\) subsets of \(X\), called blocks (\(B_i \subseteq X\));
3. For any \(\Delta\) and any \(\Lambda \subseteq \{1, \ldots, n\}\) with \(|\Delta| = t\) and \(|\Lambda| = t - 1\), \(|\bigcup_{i \in \Delta} B_i| > |\bigcup_{j \in \Lambda} B_j|\).

We will call \((X, \mathcal{B})\) a \((v, n, t)\)-strong cover-free family (or \((v, n, t)\) - SCFF for short).

The idea behind our new construction is to combine an existing threshold scheme and a SCFF to construct a new threshold scheme for large group of participants. The construction works as follows.

1. Assume \((X, \mathcal{B})\) is a \((v, n, t)\)-SCFF. Let \(\ell\) be a integer such that \(\min_{\Delta} |\bigcup_{i \in \Delta} A_i| \geq \ell > \max_{\Lambda} |\bigcup_{i \in \Lambda} A_i|\) where \(\Delta\) runs through all \(t\)-subsets of \(\{1, \ldots, n\}\) and \(\Lambda\) runs through all \((t - 1)\)-subsets of \(\{1, \ldots, n\}\). Since \((X, \mathcal{B})\) is a SCFF, such \(\ell\) exists.
2. Assume there is a \((\ell, v)\) threshold scheme \((D_0, C_0)\). For a secret \(k \in K\), the \(v\) shares of \((D_0, C_0)\) are \(a_1, \ldots, a_v\).

3. Define a \((t, n)\) threshold scheme for \(n\) participants \(P_1, \ldots, P_n\) by constructing \(n\) share \(s_1, \ldots, s_n\) as \(s_i = \{a_j \mid \text{if and only if } x_j \in B_i\}\) and assigning \(s_i\) to the participants \(P_i\) for all \(1 \leq i \leq n\).

We denote the resulting \((t, n)\) scheme as \((D, C)\) and prove the following result.

**Theorem 6.2** If \((D_0, C_0)\) is perfect, then \((D, C)\) is perfect. Moreover, if \((D_0, C_0)\) is multiplicative, then so is \((D, C)\).

**Proof.** (Sketch) Clearly, any \(t\) participants, \(P_{i_1}, \ldots, P_{i_t}\), say, have \(|s_{i_1} \cup \cdots \cup s_{i_t}|\) shares from the \(v\) share of the \((\ell, v)\) threshold scheme \((D_0, C_0)\). From the choice of \(\ell\), we know that \(t\) participants can reconstruct the secret by applying the combiner algorithm \(C_0\). Next, any \(t - 1\) participants have no extra information about \(k\) provided \((D_0, C_0)\) is perfect. Indeed, without loss of generality, assume that \(P_1, \ldots, P_{t-1}\) want to recover the secret by using their shares \(s_1 \cup \cdots \cup s_{t-1} \subseteq \{a_1, \ldots, a_v\}\). Since \(|s_1 \cup \cdots \cup s_{t-1}| \leq \ell\) and the underlying \((t, v)\) scheme \((D_0, C_0)\) is perfect, the claim follows. Thirdly, the verification for the multiplicative property is straightforward.

Note that the share expansion \(\rho\) of \((D, C)\) is determined by the share expansion \(\rho_0\) of \((D_0, C_0)\) and the parameters of the \((v, n, t)\)-SCFF \((X, B)\). We have \(\rho \leq \max_{1 \leq i \leq n} |B_i| \rho_0\).

In particular, if \((D_0, C_0)\) is *ideal*, i.e. \(\rho_0 = 1\) and \(|B_i| = r\) for all \(1 \leq i \leq n\), then \(\rho = r\).

### 6.3.1 An example

To illustrate the efficiency of our new approach, we compare our construction and the BBDW construction through an example. As noted before BBDW scheme can reach logarithmic share expansion. As we will show in section 6.5.2 our construction can achieve the same result too. One of the advantages of our method is that unlike BBDW construction that requires multiple instances of the old scheme for the *same* secret, which is difficult or impossible in some cryptographic settings like MTA-free environment [61], it only uses a single instance of the old secret sharing scheme. Another advantage of our approach is that the randomness coefficient of the new system is the same as the old system while in BBDW scheme it is \(N\) times that of the old one. We note that this does not imply that the randomness coefficient in the latter is \(N\) times the former simply because the old scheme in the two cases are different and in particular in BBDW the old scheme has much smaller parameters. The following example, though contrived to some extent, explains these ideas.
Example 6.1 Suppose we want to construct a multiplicative \((2,70)\) secret sharing over \(GF(2^4)\) (as an Abelian group). Using BBDW construction, we use a \((N;70,2,2)\) perfect hash family and \(N\) instances of the multiplicative \((2,2)\) ideal threshold schemes with the same secret. It is well known that \((N;70,2,2)\) perfect hash family with optimal \(N = \lceil \log 70 \rceil = 7\) does exist. So the share expansion of the resulting \((2,70)\) scheme is \[\max_i \{ \log |S_i| / \log |\mathcal{K}| \} = 7,\] and the dealers randomness is \(N \times \log |\mathcal{K}| = 7 \times 4 = 28\) bits.

Now we use our SCFF approach. Let \(X = \{ x_1, x_2, \ldots, x_8 \}\) and \(\mathcal{B} = \{ B : |B| = 4, B \subseteq X \}\). Then \((X, \mathcal{B})\) is a \((8,70,2)\)-SCFF. Let the underlying secret sharing scheme be a \((5,8)\) multiplicative ideal secret sharing scheme (for example using Shamir’s construction). Then it yields a \((2,70)\) multiplicative threshold scheme, in which the share expansion is 4 and the randomness requested is \(4 \times 4 = 16\) bits.

6.4 Bounds

As noted earlier the efficiency of our new construction relies on the parameters of strong cover free family used in the construction. In this section we will derive some bounds on various parameters of SCFFs.

A trivial construction of SCFF is by letting \(\mathcal{B}\) to be the collection of singleton sets of \(X\), in this case, \(n \leq v\). However, for our applications we will be only interested in SCFF with \(n > v\). The following theorem completely characterises the SCFF when \(t = 2\).

Theorem 6.3 There exists a \((v,n,2)\)-SCFF if and only if \(n \leq \binom{v}{\lfloor \frac{v}{2} \rfloor}\).

Proof. Assume that \((X, \mathcal{B})\) is a \((v,n,2)\)-SCFF, then it is easy to see that there do not exist two distinct blocks \(B_i, B_j\) such that \(B_i \subseteq B_j\), i.e., \((X, \mathcal{B})\) is a Sperner Family. It is well-known that there exists a Sperner family consisting of \(n\) subsets of a \(v\)-set only if \(n \leq \binom{v}{\lfloor \frac{v}{2} \rfloor}\) (see [28]). Conversely, we can take all \(\lfloor \frac{v}{2} \rfloor\)-subsets of a \(v\)-set, it is easy to see that it results in a \((v,n,2)\)-SCFF with \(n = \binom{v}{\lfloor \frac{v}{2} \rfloor}\), proving the desired result.

Next, we derive a lower bound on \(v\) for given \(n\) and \(t\).

Theorem 6.4 In any \((v,n,t)\)-SCFF, we have \(v \geq (t - 1) \log \frac{n}{t-1}\).

Proof. Assume that \((X, \mathcal{B})\) is a \((v,n,t)\)-SCFF. Let \(\mathcal{F} = \{ \cup_{i \in \Delta} B_i : \Delta \subseteq \{1, \ldots, n\} \} \) \(\)with \(|\Delta| = t - 1, B_i \in \mathcal{B}\). We observe that for any \(\Delta \neq \Delta' \subset \{1, \ldots, n\}\) with \(|\Delta| = |\Delta'| = t - 1\), we have \(\cup_{i \in \Delta} B_i \neq \cup_{j \in \Delta'} B_j\). Indeed, otherwise assume that there
are $\Delta$ and $\Delta'$ with $|\Delta| = |\Delta'| = t - 1$ such that $\cup_{i \in \Delta} B_i = \cup_{j \in \Delta} B_j$. Since $\Delta \neq \Delta'$, we may assume that there is an element $\ell \in \Delta'$, but $\ell \notin \Delta$. It follows $\cup_{i \in \Delta \cup \{\ell\}} B_i = \cup_{i \in \Delta} B_i$ and hence $|\cup_{i \in \Delta \cup \{\ell\}} B_i| = |\cup_{i \in \Delta} B_i|$, which contradicts the assumption that $(X, B)$ is a $(v, n, t) - SCFF$. So we have $|F| = \binom{n}{t-1}$. Similarly, it is easy to see that $F$ is a Sperner family, that is, for any $F \neq F' \in F$, we always have $F \nsubseteq F'$. It follows that $|F| \leq \binom{\frac{n}{2}}{\frac{t}{2}}$. Since $\binom{n}{t-1} > \binom{\frac{n}{2}}{\frac{t}{2}}$ and $\binom{\frac{n}{2}}{\frac{t}{2}} \leq 2^v$, we obtain $\binom{n}{t-1} \leq 2^v$, and so $v \geq (t-1) \log \binom{n}{t-1}$.

The above theorem can be restated as $n \leq (t-1)2^\frac{v}{t-1}$, which gives an upper bound on $n$ for given $v$ and $t$. The following theorem improves the upper bound on $n$ when the SCFF has constant size of blocks.

**Theorem 6.5** Let $(X, B)$ be a $(v, n, t) - SCFF$ such that we have, for all $B_i \in B$ $|B_i| = r$. Then $n \leq \binom{r}{\ell}/\binom{r-1}{\ell-1}$, where $\ell = \lfloor r/t - 1 \rfloor$.

**Proof.** Assume that $(X, B)$ is a $(v, n, t)$-SCFF. Let $\Delta$ be a subset of $\{1, \ldots, n\}$ such that $|\Delta| = t - 1$, and let $i \notin \Delta$. Then we have $|\Delta \cup \{i\}| = t$, and so $|\cup_{j \in \Delta \cup \{i\}} B_j| > |\cup_{j \in \Delta} B_j|$. It follows that $B_i \nsubseteq \cup_{j \in \Delta} B_j$, that is, the union of any $t - 1$ blocks in $B$ can not cover any remaining one in $B$ (such a set system is called $(v, n, t - 1)$-cover free family [47]). By Proposition 2.1 of [47], the result follows immediately.

We can show that in any non-trivial SCFF $(X, B)$, that is, $|X| < |B|$, the parameter $t$ can not be too large relative to $n$. Indeed, assume that $(X, B)$ is a $(v, n, t) - SCFF$. From the proof of Theorem 6.5 we know that $(X, B)$ is a $(v, n, t - 1)$-cover-free family. By Proposition 3.4 of [47], we have $n \geq \binom{t+1}{2} > t^2/2$, and the desired result follows.

**Theorem 6.6** In a $(v, n, t) - SCFF$, where $v < n$, we have $t < \sqrt{2n}$.

### 6.5 Constructions

The following lemma is essential for our later constructions.

**Lemma 6.7** Let $(X, B)$ be a set system such that

1. $|B_i| = r$, for $i \in \{1, \ldots, n\}$;

2. $|B_i \cap B_j| \leq \mu$, for $i \neq j \in \{1, \ldots, n\}$.

Then $(X, B)$ is a $(v, n, t) SCFF$ provided $\binom{t}{2} < r/\mu$. 
6.5. Constructions

Proof. Let \( \Delta \) and \( \Lambda \) be two subsets of \( \{1, \ldots, n\} \) such that \( |\Delta| = t \) and \( |\Lambda| = t - 1 \). We have 
\[
|\bigcup_{i \in \Delta} B_i| \geq \sum_{i \in \Delta} |B_i| - \sum_{i,j \in \Delta} |B_i \cap B_j| \geq tr - \binom{t}{2} \mu = (t - 1)r + (r - \binom{t}{2} \mu) > (t - 1)r - \sum_{j \in \Lambda} |B_j| \geq |\bigcup_{j \in \Lambda} B_j|.
\]

6.5.1 Constructions from combinatorial designs

In this subsection, we will give some constructions of SCFF from certain combinatorial designs, including \( \mu \)-designs, packing designs and orthogonal array. Similar constructions for tractability scheme and frameproof codes can be found in [111].

As before, we will use \( (X, B) \) to denote a set system in which \( X \) is a finite set and \( B \) is a family of subsets of \( X \). The elements of \( X \) and \( B \) are called points and blocks, respectively. A \( \mu - (v, r, \lambda) \) design is a set system \( (X, B) \), where \( |X| = v, |B| = r \) for every \( B \in B \), and for every \( \mu \)-subset of \( X \) occurs in exactly \( \lambda \) blocks in \( B \). We will be only interested in \( \mu - (v, r, 1) \) design. It is well known that in a \( \mu - (v, r, 1) \) design the number of blocks \( n \) is exactly \( \binom{v}{\mu}/\binom{r}{\mu} \). Assume there exists a \( \mu - (v, r, 1) \) design \( (X, B) \). Then for each pair \( B_i, B_j \in B \), we have \( |B_i \cap B_j| \leq \mu - 1 \). From Lemma 6.7 the following theorem is immediate.

**Theorem 6.8** If there exists a \( \mu - (v, r, 1) \) design, then there exists a \( (v, \binom{v}{\mu}/\binom{r}{\mu}, r) \)-SCFF for any \( r \) satisfying \( \frac{r}{\mu - 1} \).

There are many results on existence and constructions of \( \mu - (v, r, 1) \) design for \( r = 2, 3 \). On the other hand, no \( \mu - (v, r, 1) \) design with \( v > r > \mu \) is known to exist for \( \mu \geq 6 \). Furthermore, it is known that for \( 3 \leq r \leq 5 \), a \( 2 - (v, r, 1) \) design exists if and only if \( v \equiv 1 \), or \( r \mod (r^2 - r) \). To apply Theorem 6.8, it is required \( r \geq 4 \) and so that \( \binom{3}{2} \leq \frac{r}{r/(\mu - 1)} \), where \( \mu = 2 \). Since \( 2 - (v, 4, 1) \) design exists for any \( v \equiv 1, 4 \mod 12 \), Theorem 6.8 yields the following result.

**Theorem 6.9** There exists \( (v, \frac{v(v-1)}{12}, 3) \)-SCFF for all \( v \equiv 1, 4 \mod 12 \).

A \( \mu - (v, r, \lambda) \) packing design is a set system \( (X, B) \), where \( |X| = v, |B| = r \) for every \( B \in B \), and every \( \mu \)-subset of \( X \) occurs in at most \( \lambda \) blocks in \( B \). Similar to Theorem 6.8, we have the following theorem.

**Theorem 6.10** If there exists a \( \mu - (v, r, 1) \) packing design having \( n \) blocks, then there exists a \( (v, n, t) \)-SCFF if \( \binom{t}{2} < \frac{r}{\mu - 1} \).
As we noted previously, no \( \mu - (v, r, 1) \) designs are known to exist if \( v > r > \mu \geq 6 \). However, for any \( \mu \), there are infinite classes of packing designs with a “large” number of blocks (i.e. close to \( \binom{v}{\mu}/\binom{r}{\mu} \)). Such packing designs can also be constructed from orthogonal array. Recall that an orthogonal array \( OA(\mu, r, s) \) is a \( r \times s^\mu \) array, with entries from a set of \( s \geq 2 \) symbols, such that in any \( \mu \) rows, every \( \mu \times 1 \) column vector appears exactly once. In [111], Stinson and Wei showed that if there is an \( OA(\mu, r, s) \), then there is a \( \mu - (rs, r, 1) \) packing design that contains \( s^\mu \) blocks. It is well known that for any prime power \( q \) with \( \mu < q \), there exists an \( OA(\mu, q + 1, q) \). It follows that there exists a \( \mu - (q^2 + q, q + 1, 1) \) packing design \( (X, B) \). From Theorem 6.10, we have the following theorem.

**Theorem 6.11** For any prime power \( q \) and any integer \( \mu < q \), there exist \( (q^2 + q, q^\mu, t) - SCFF \) for any \( t \) satisfying \( \binom{t}{\mu} < \frac{q-1}{\mu-1} \).

### 6.5.2 Constructions from universal hashing families

The concept of universal hashing family was invented by Carter and Wegman [29] and has found numerous applications in computer science [108].

Let \( \epsilon > 0 \). A multiset \( H \) of \( N \) function from a \( n \)-set \( X \) to a \( m \)-set \( Y \) is called \( \epsilon \)-almost universal (\( \epsilon - AU \) for short) if for any two distinct elements \( x_1, x_2 \in X \), there exists at most \( \epsilon N \) functions \( h \in H \) such that \( h(x_1) = h(x_2) \). Without loss of generality we will assume that \( n \geq m \). We call \( H \) an \( \epsilon - AU(N; n, m) \) hashing family. The following shows that SCFF can be constructed from \( AU \) hashing families.

**Theorem 6.12** If there exists an \( \epsilon - AU(N; n, m) \) hashing family, then there exists a \((Nm, n, t)-SCFF \) provided \( \binom{t}{\mu} < 1/\epsilon \).

**Proof.** Assume that \( H \) is an \( \epsilon - AU(N; n, m) \) hashing family from \( S \) to \( T \). We construct a set system \( (X, B) \) as follows. Set \( X = H \times T = \{(h, t) : h \in H, t \in T \} \) and \( B = \{B_s : s \in S \} \), where for each \( s \in S \) we define \( B_s = \{(h, h(s)) : h \in H \} \). Then it is easy to see that \( |X| = Nm, |B| = n \) and \( |B_s| = N \) for each \( s \in S \). For each pair \( B_s, B_{s'} \in B \), we have,

\[
|B_s \cap B_{s'}| = |\{(h, h(s)) : h \in H\} \cap \{(h, h(s')) : h \in H\}| \\
= |\{h : h(s) = h(s'), h \in H\}| \\
\leq \epsilon N
\]

From Lemma 6.7, we know that \((X, B)\) is an \((Nm, n, t)\) SCFF if \( \binom{t}{\mu} < \frac{N}{\epsilon N} = \frac{1}{\epsilon} \), and the desired result follows.
\( \epsilon - AU \) are strictly related to error-correcting codes (see [17]). Let \( Y \) be an alphabet of \( q \) symbols. An \((N, M, D, q)\) code is a set \( \mathcal{C} \) of \( M \) vectors in \( Y^N \) such that the Hamming distance between any two distinct vectors in \( \mathcal{C} \) is at least \( D \). The code is \textit{linear} if \( q \) is a prime power, \( Y = GF(q) \), and \( \mathcal{C} \) is a subspace of \( GF(q)^N \). Then we will denote it by an \([N, m, D, q]\) code, where \( m = \log_q M \) is the \textit{dimension} of the code.

Let \( \mathcal{C} \) be a \((N, M, D, q)\) code, we can define a family of functions \( H = \{h_1, \ldots, h_N\} \) from \( M \) to \( Y \),

\[
(h_1(v), h_2(v), \ldots, h_N(v)) = (u_1, \ldots, u_N)
\]

for any \( v = (v_1, \ldots, v_N) \in M \).

The following equivalence is due to Bierbrauer, Johansson, Kabatianskii and Smeets [17], we took it from Stinson [108]

\textbf{Theorem 6.13 ([108, 17])} \textit{If there exists an \((N, M, D, q)\) code, then there exists a \((1 - \frac{D}{N}) - AU(N; M, q)\) hash family. Conversely, if there exists an \( \epsilon - AU(N; n, m)\) hash family, then there exists an \((N, n, N(1 - \epsilon), m)\) code.}

If we apply the above theorem to Justesen codes (Theorem 9.2.4 [118]), we obtain a \((v, n, t)\)-SCFF \((X, B)\) with \(|B_i| = O(\log n)\), for all \( B_i \in B \), and the share expansion of the new scheme in our construction is \(O(\log n)\). Blackburn et al [20] showed a similar result by using perfect hash families.

Another application of Theorem 6.13 is to use Reed-Solomon codes. An \textit{extended Reed-Solomon} code is a linear code having parameters \([q, t, q - t + 1, q]\), where \( t \leq q \) and \( q \) is a prime power. Applying Theorem 6.12 and 6.13 we have the following theorem.

\textbf{Theorem 6.14} \textit{Let \( q \) be a prime power and \( 1 \leq \ell \leq q \). There exists a \((q^2, q^\ell, t)\) SCFF, where \( t \leq \sqrt{\frac{2q}{\ell - 1}} + 1 \).}

\textbf{Proof.} Applying the extended Reed-Solomon codes in Theorem 6.13, we know that there is a \( \frac{\ell - 1}{q} - AU(q, q^\ell, q)\) hashing family. The result follows immediately from Theorem 6.12.

Using the recursive construction, Stinson (Theorem 6.1 [105]) proved that there exists an \( i/q - AU(q^i; q^{2^i}, q)\) hashing family. This in conjunction with with Theorem 6.12 gives us an infinite class of SCFFs.

\textbf{Theorem 6.15} \textit{Let \( q \) be a prime power and let \( i \geq 1 \) be an integer. Let \( t \leq \sqrt{\frac{2q}{i^2}} + 1 \). Then there exists a \((q^{i+1}, q^{2^i}, t)\) SCFF.}
6.5.3 Construction based on exponential sums

In [59] Helleseth and Johansson used exponential sums over finite fields to construct strongly universal hashing families and authentication codes. Motivated by the universal construction in the previous subsection we show that exponential sums can be used to construct SCFF with good parameters.

Let $GF(q)$ be a finite field with characteristic $p$, and let $Tr_{q^m/q}(\alpha)$ be the trace function from $GF(q^m)$ to $GF(q)$ defined by

$$Tr_{q^m/q}(\alpha) = \alpha + \alpha^q + \cdots + \alpha^{q^{m-1}}.$$ 

Lemma 6.16 ([59]) Let $f(x) = \sum_{i=1}^{D} a_i x^i \in GF(q^m)[x]$ be a polynomial of degree $D$ that is not expressible in the form $f(x) = g(x)^p - g(x) + \theta$ for any $g(x) \in GF(q^m)[x], \theta \in GF(q^m)$. Let

$$N_\alpha(f) = \{|x \in GF(q^m) : Tr_{q^m/q}(f(x)) = \alpha|\}.$$ 

Then $N_\alpha(f) \leq q^{m-1} + (D - 1)\sqrt{q^m}$ for any $\alpha \in GF(q)$.

Let $D \leq \sqrt{q^m}$, we define a set $F_D$ of polynomials with degree less than or equal to $D$ by,

$$F_D = \{f(x) : f(x) = a_1 x + a_2 x^2 + \cdots + a_D x^D \in GF(q^m)[x], a_i = 0, \text{ whenever } p|i\}.$$ 

Then, $F_D$ is clearly a $(D - [D/p])$-dimensional vector space over $GF(q^m)$, and so we have $|F_D| = q^{m(D-[D/p])}$. Moreover, it is easy to see that for each $f(x) \in F_D$, $f(x)$ can not be expressed in the form $f(x) = g(x)^p - g(x) + \theta$, and Lemma 6.16 can be applied. That is, for each $f(x) \in F_D$ and $\alpha \in GF(q)$, we have $N_\alpha(f) \leq q^{m-1} + (D - 1)\sqrt{q^m}$.

For each $\beta \in GF(q^m)$, we associate a function $g_\beta$ from $F_D$ to $GF(q)$ defined by $g_\beta(f) = Tr_{q^m/q}(f(\beta))$.

Let $G = \{g_\beta : \beta \in GF(q^m)\}$, $X = G \times GF(q)$ and $B = \{B_f, f \in F_D\}$, where $B_f = \{(g_\beta, g_\beta(f)) : \beta \in GF(q^m)\}$. We will show that such set systems $(X, B)$ are SCFF with appropriate parameters.

Lemma 6.17 $|B_f \cap B_{f'}| \leq q^{m-1} + (D - 1)\sqrt{q^m}$, for any $f \neq f' \in F_D$.

Proof.

$$|B_f \cap B_{f'}| = |\{g_\beta : g_\beta(f) = g_\beta(f')\}|$$
$$= |\{\beta : Tr_{q^m/q}(f(\beta)) = Tr_{q^m/q}(f'(\beta))\}|$$
$$= |\{\beta : Tr_{q^m/q}(f - f')(\beta) = 0, \beta \in GF(q^m)\}|$$
$$\leq N_0(f - f')$$
$$\leq q^{m-1} + (D - 1)\sqrt{q^m}$$
Combining Lemma 6.17 and Lemma 6.7, we have the following result.

**Theorem 6.18** \((X, B)\) is a \((q^{m+1}, q^{m(D-[D/p])}, t) - SCFF\) provided

\[
\left( \frac{t}{2} \right) \leq q^m/(q^{m-1} + (D - 1)\sqrt{q^m}).
\]

The above theorem results in an infinite class of SCFFs with good parameters. For example, taking \(D = q^{m/2-1} + 1\) for any even \(m\), then for any \(t\) satisfying \(\left( \frac{t}{2} \right) \leq q/2\), applying the above theorem gives a

\[
(q^m, q^{m(q^{m/2-1}-[q^{m/2-1}/p])}, \lfloor \sqrt{2q^{m/4}} \rfloor) - SCFF
\]

for all even \(m\). A simple approximation yields that there is an infinite class of \((v, n, t) - SCFFs\) in which the parameters satisfy \(\log n = C\sqrt{v}\log v\), where \(C\) is a fixed constant. We have \((\log n)^2 = C^2 v(\log v)^2 \geq c^2 v\), and so there exists an infinite class of \((v, n, t) - SCFF\) in which \(v = O((\log n)^2)\).

### 6.6 Evaluation

In the following we translate some of the results in the previous section into constructions of new threshold schemes from old ones. Unlike BBDW scheme in which threshold value in the old and the new system are the same, in our approach the threshold value may change. Given a \((v, n, t) - SCFF\) it is straightforward to see that the new scheme is a \((t, n)\) threshold scheme. However the old scheme is a \((\ell, v)\) threshold scheme where \(\ell = \min_{\Delta} |\cup_{i \in \Delta} A_i|\) where \(\Delta\) runs through all \(t\)-subsets of \(\{1, \ldots, n\}\). The value of \(\ell\) must be calculated for each case separately and so in the following we only give the parameters of the new scheme.

1. For any integer \(v \equiv 1, 4 \mod 12\), a \((\ell, v)\) scheme results in a \((3, \frac{v(v-1)}{12})\) scheme (Theorem 6.9).

2. For any prime power \(q\) and any integer \(\mu < q\), a \((\ell, q^2 + q)\) scheme results in \((t, q^\mu)\) scheme provided \(\left( \frac{t}{2} \right) < \frac{q-1}{\mu-1}\) (Theorem 6.11).

3. For any prime power \(q\) and integer \(i \geq 1\), a \((\ell, v)\) scheme results in a \((t, q^{2^i})\) scheme provided \(t \leq \sqrt{\frac{2v}{t}} + 1\) (Theorem 6.15).

4. For any prime power \(q\) and even \(m\), a \((b, q^m)\) scheme results in a \((t, q^{cmq^{m/2}})\) scheme provided \(q > 2\left( \frac{t}{2} \right)^2\), where \(c\) is some fixed constant (Theorem 6.18).
For example using $q = 81$, $\mu = 3$ and the result in 2 we can construct a $(t, 81^3)$ scheme for $t \leq 9$. Using the same value of $q$ and $i = 2$ together with the result in 3 we obtain a $(t, 81^4)$ scheme for $t \leq 10$. In the former case the old scheme is a $(\ell, 81^2 + 81)$ threshold scheme and in the latter case it is a $(\ell, 81^3)$ scheme. However finding $\ell$ requires careful investigation.

It is tempting to look for a SCFF with $\ell = v$. This will imply that we can construct a $(t, n)$ scheme from a $(v, v)$ scheme and as the latter kind of scheme can always be efficiently constructed in multiplicative case, construction of multiplicative $(t, n)$ will reduce to the construction of appropriate SCFF. However the following theorem shows that this is not possible.

Suppose we have a $(v, n, t)$-SCFF with $|X| = v$ and $|B| = n$. It is easy to see that the addressed problem of constructing a $(t, n)$ scheme from a $(v, v)$ scheme is equivalent to the following condition

$$B_{i_1} \cup \ldots \cup B_{i_t} = X \text{ and } B_{j_1} \cup \ldots \cup B_{j_{t-1}} \neq X.$$ (6.1)

**Theorem 6.19** Let $(X, B)$ be a $(v, n, t)$-SCFF. If the condition (6.1) holds, then $v \leq n$.

**Proof.** Since $(X, B)$ is a $(v, n, t)$-SCFF, from Theorem 6.6, we have $t < \sqrt{2n}$. Now if the condition (6.1) holds, then for each $j \in \Delta$, $B_j \supseteq X \setminus \bigcup_{i \in \Delta} B_i \neq \emptyset$. It follows that $\bigcap_{j \in \Gamma} B_j \supseteq X \setminus \bigcup_{i \in \Delta} B_i$, where $\Gamma = \{1, \ldots, n\} \setminus \Delta$. In other words, $\bigcap_{j \in \Gamma} B_j \subseteq \bigcup_{i \in \Delta} B_i$.

Now we consider the set systems $(X, B')$, where $B' = \{B' \colon B' = X \setminus B, B \in B\}$. Then we have $\bigcap_{i \in \Delta} B'_i \subseteq \bigcup_{j \in \Gamma} B'_j$. It follows that for any $\Gamma \subseteq \{1, \ldots, n\}$ with $|\Gamma| = n - t + 1$ and $i \not\in \Gamma$ we have $B'_i \not\subseteq \bigcup_{j \in \Gamma} B'_j$. That is, $(X, B')$ is a $(n - t + 1)$-cover-free family. From Proposition 3.4 in [47], we have \(\left(\frac{n-t+1}{2}\right) < n\), this contradicts with $t < \sqrt{2n}$ as we observed already.

This shows that to construct a multiplicative $(t, n)$ scheme using SCFF, we need a $(\ell, v)$ multiplicative with $\ell \neq v$. It is interesting to note that in BBDW construction, by using a minimal perfect hash family $PHF(N, n, t, t)$ one can construct a $(t, n)$ scheme from a $(t, t)$ scheme.

### 6.7 Concluding Remarks

We proposed a new approach to constructing a new $(t, n)$ threshold scheme from an old $(\ell, v)$ scheme. The approach can achieve similar efficiency as BBDW approach in terms of share expansion, but in practice is not as easy to use. The very attractive feature of
BBDW is that the old scheme can be a \((t, t)\) scheme for which an efficient construction for all values of \(t\) exists. Our approach starting with the same old scheme will result in \(v > n\). However, our approach is the only possible approach in environments such as MTA-free environment, in which multiple instances of the old scheme with the same secret cannot be guaranteed. With respect to the second efficiency parameter of a threshold scheme, that is randomness coefficient, our approach is likely to be more efficient than BBDW scheme. Although making a definite statement requires further research and is an open problem. Overall the two approaches are complementary and in conjunction with each other can produce threshold schemes with a wide range of parameters.

SCFFs, although strictly defined and used in the context of the construction of new threshold schemes from old ones, are of interest in their own rights. Although a \((v, n, t)\)-SCFF in general cannot be shown to be a \((v, n, t - i)\)-SCFF, but all the constructions given in the present chapter have this property. Construction of SCFF families for which \(\ell = \min_{\Delta} |\bigcup_{i \in \Delta} A_i|\), where \(\Delta\) runs through all \((t - 1)\)-subsets of \(\{1, \ldots, n\}\), can be easily calculated is an open problem of high interest in our approach.


[80] D. Pei, Information theoretic bounds for authentication codes and PBIB, Presented at the Rump session of Asiacrypt'91


