# Secure Convertible Codes

**Justin Zhang**

CMU-CS-24-160

May 2025

Computer Science Department
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

**Thesis Committee:**
Rashmi Vinayak, Chair
Aayush Jain

*Submitted in partial fulfillment of the requirements*
*for the degree of Master of Science in Computer Science.*

**Keywords:** Erasure Codes, Information-theoretic Security, Fault Tolerance

*For my grandmother.*

# Abstract

Large-scale distributed storage systems rely on erasure codes to ensure fault tolerance against node failures. Due to the observed changing failure rates within these systems, code redundancy tuning, or *code conversion* has been shown to reduce storage cost. Previous work has developed theoretical bounds and constructions for *convertible codes,* a specialized class of erasure codes optimizing either access or bandwidth costs during conversion.

In this thesis, we address the challenge of securing convertible codes in the presence of an eavesdropper. We introduce an eavesdropper-secrecy model for convertible codes wherein an eavesdropper gaining read access to a subset of the codeword symbols learns nothing (information-theoretically) about the underlying message. We then focus on access-cost optimal convertible codes, and we then derive the information-theoretic upper bound on the number of message symbols that can be stored securely. We provide an explicit construction that simultaneously reaches this secrecy bound while admitting access-cost optimal conversion using concatenation of nested codes with traditional convertible codes. Since our construction works with all traditional access-optimal convertible codes, we show that access-optimal secure convertible codes exist for all message and codeword length parameters. Lastly, we discuss a relaxed secrecy model where the number of eavesdropped symbols on each initial and final codeword is known.

## Acknowledgments

Thank you to my advisor, Rashmi Vinayak, for her incredible patience and guidance. Under her tutelage, I learned how to ask the right questions, argue in the face of science's unforgiving rigor, and communicate my ideas effectively.

More broadly, I would like to thank the computer science department for the wide assortment of professors and classes which fostered my love for theory. And of course, I thank family and friends, whose love and support kept me grounded in the busiest of times. I hope I have made you all proud.

# Contents

# List of Figures

# Chapter 1

# Introduction

Erasure codes provide a low-storage overhead solution to ensure fault tolerance against node failures in large-scale distributed storage systems [5, 7]. In this approach, the data is divided into $k$ message symbols, which are then encoded using an $[n, k]$ erasure code into $n$ coded symbols, forming a *codeword* using an $[n, k]$ erasure code. These codewords are distributed across $n$ different nodes in the storage system. To achieve optimal storage efficiency and fault tolerance, Maximum Distance Separable (MDS) codes are typically employed. Informally, the MDS property ensures data integrity by allowing recovery of the original data even if up to $(n - k)$ nodes fail. In other words, any $k$ out of the $n$ codeword symbols are sufficient to decode the original data.

The parameters $n$ and $k$ are selected based on the observed node failure rates, which, as shown by Kadekodi et al., can vary over time [8]. During periods of high failure rates, $n$ and $k$ are configured to achieve a high redundancy ratio $\frac{n}{k}$, ensuring greater fault tolerance at the expense of increased storage overhead. Conversely, during periods of low failure rates, a lower redundancy ratio is preferred, reducing storage overhead. However, changing the parameters $n$ and $k$ on already encoded data using the conventional approach - decoing the data from the initial code and re-encoding it with a new code - incurs significant costs in terms of I/O and network bandwidth [9].

This problem has been formalized under the theoretical framework of *code conversion* [9], which defines the conversion of data from an initial code $\mathcal{C}^I$ with parameters $[n^I, k^I]$ to a final code $\mathcal{C}^F$ with parameters $[n^F, k^F]$. *Convertible codes* [9] are a class of codes that by design minimize the costs of code conversion, while maintaining certain decodability guarantees (such as the MDS property) in both the initial and final codes. Convertible codes have been studied primarily in terms of minimizing conversion costs, with two key cost metrics: access cost [9, 12], which measures the number of symbols accessed during conversion, and bandwidth cost [10, 11], which measures the amount of information downloaded. Access-optimal convertible codes are known for all parameter settings, while bandwidth-optimal convertible codes have been developed for certain parameter regimes.

In this thesis, we consider the problem of information-theoretic security of convertible codes. Specifically, we investigate security against passive *eavesdroppers* who gain read access to some of codeword symbols stored in the system and try to learn information about the message symbols. This problem setting has been inspired by several prior works on information-theoretic

security in distributed storage codes under various models, such as secure regenerating codes [2, 4, 15, 17]. We first introduce a secrecy model for convertible codes, incorporating requirements for data decoding, code conversion, and eavesdropper secrecy. For a specified security parameter $\ell$, the objective is to ensure that an eavesdropper who reads any $\ell$ code symbols of a convertible code learns no information about the message symbols.

We then focus on access-optimal convertible codes and establish an upper bound on the number of message symbols that can be securely stored using convertible codes using an information-theoretic approach. We then present an explicit construction of an *access-optimal* secure convertible code that achieves this upper bound for all parameter settings. The proposed construction uses code concatenation of nested codes [19] with traditional convertible codes [9, 12]. Lastly, we consider a relaxation of the secrecy model, where we assume additional knowledge about the symbols accessed by eavesdroppers on each codeword. We call this modified model the *fine-grained secrecy model*. We derive the secrecy capacity under this model and discuss the plausible scenarios that can be modeled and accounted for using the fine-grained model.

The outline of the thesis is as follows. We review the relevant background and notation contextualizing secure convertible codes in Chapter 2. In Chapter 3, we presents the secrecy model for convertible codes. In Chapter 4, we prove the secrecy capacity of any secure convertible code and presents a construction of access-optimal secure convertible codes that reach secrecy capacity for all parameters. We present the fine-grained secrecy model for convertible codes in Chapter 5, deriving the upper bound on secrecy capacity and discussing the scenarios that can be captured using the fine-grained secrecy model. Lastly, we conclude the thesis with a discussion and future work description in Chapter 6.

# Chapter 2

# Background and Relevant Work

In this chapter, we review relevant background and prior work. We will first define notation that we will use throughout the thesis.

## 2.1 Notation

Caligraphic, uppercase letters $\mathbb{T}$ denote sets. Bold lowercase letters will denote vectors, e.g. a $n$-length vector, $\boldsymbol{x} \in \mathbb{F}^n$ , where $\mathbb{F}$ is a finite field. When relevant, we denote $\mathbb{F}_q$ as the finite field of size $q$. The $i$'th symbol of a vector $\boldsymbol{x}$ is written (non-bold) as $x_i$. A vector subscripted with a set, e.g. $\boldsymbol{x}_S$, denote the projection of the vector to each coordinate in the set $\boldsymbol{x}$ e.g. $\boldsymbol{x}_{\mathbb{S}} = [x_i : i \in \mathbb{S}]$. Uppercase letters denote matrices, e.g. a matrix of size $k \times n$, $G \in \mathbb{F}^{k \times n}$, while Calligraphic letters $\mathcal{C}$ will denote codes. For any vector $\boldsymbol{x}$, its corresponding random variable is denoted as $\boldsymbol{\mathcal{X}}$ (uppercase, calligraphic, and bold). Let $[i] = \{1, 2, \ldots, i\}$. Let $\Pi(i)$ be the set of all partitions of $[i]$. Lastly, let $H$ be the entropy function (in base $|\mathbb{F}|$).

## 2.2 Erasure Codes

An $(n, k)$ erasure code $\mathcal{C}$ is a mapping from *messages* $\boldsymbol{m} \in \mathbb{F}^k$ to *codewords* $\boldsymbol{c} \in \mathbb{F}^n$. We say $\mathcal{C}$ is linear if it is a linear mapping and can be represented with *generator matrix* $G \in \mathbb{F}^{k \times n}$ (we denote this by using square brackets e.g $\mathcal{C}$ is a $[n, k]$ code). We also say a $[n, k]$ erasure code $\mathcal{C}$ is systematic if $G = \Big[ I_k \mid P \Big]$, where $I_k$ is the identity matrix of size $k$ and we say $P$ is the *parity matrix*.

Further, a $[n, k]$ code $\mathcal{C}$ is *Maximum Distance Separable (MDS)* if any subset of $k$ columns are linearly independent (and thus form a non-singular matrix). In other words, any $k$ symbols of a codeword is sufficient to reconstructing the entire codeword and its underlying message for systematic codes. In the next sections we will often refer to erasure codes simply as *codes*.

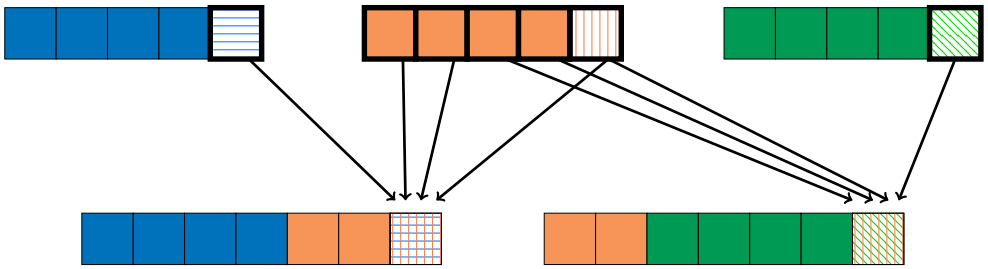

Figure 2.1: A systematic access-optimal $[5, 4; 7, 6]$ convertible code, where only the non-systematic symbols are changed. The bold-edge symbols were accessed by the conversion procedure. The arrows represent which accessed symbol was used in the computation of each new non-systematic symbol in the final codewords.

## 2.3 Convertible Codes

The traditional convertible codes framework captures the conversion between an initial and a final *configuration* of stored data [9]. In the initial configuration, data is encoded using an $(n^I, k^I)$ code $\mathcal{C}^I$, while in the final configuration, the same data is encoded using an $(n^F, k^F)$ code $\mathcal{C}^F$. Non-trivial conversion occurs when $k^I \neq k^F$, allowing multiple codewords in both configurations. Let $\boldsymbol{m} \in \mathbb{F}^k$ be the message symbols to be stored, where $k = \text{lcm}(k^I, k^F)$. The initial configuration contains $\lambda^I = k/k^I$ codewords, and the final configuration contains $\lambda^F = k/k^F$ codewords. The message symbols in each codeword is determined by the initial and final partitions $\mathbb{P}^I$ and $\mathbb{P}^F$ of $[k]$. A conversion procedure is then defined to transform the initial configuration into the final one. The access cost of conversion is measured by the number of codeword symbols used by the conversion procedure.

More formally,

**Definition 1** (Convertible Code [9])**.** A $[n^I, k^I; n^F, k^F]$ convertible code is defined by:

1. A pair of codes $(\mathcal{C}^I, \mathcal{C}^F)$ where $\mathcal{C}^I$ is a $(n^I, k^I)$ code over $\mathbb{F}$ and $\mathcal{C}^F$ is a $(n^F, k^F)$ code.
2. A pair of partitions $\mathcal{P}^I, \mathcal{P}^F \in \Pi(k)$, such that each subset $P_i^I \in \mathcal{P}^I$ has size $k^I$, and each subset $P_j^F \in \mathcal{P}^F$ has size $k^F$.
3. A conversion procedure that takes initial codewords $\{\mathcal{C}^I(\boldsymbol{m}_{\mathbb{P}_i^I}) : \mathbb{P}_i^I \in \mathbb{P}^I\}$ to final codewords $\{\mathcal{C}^F(\boldsymbol{m}_{\mathbb{P}_j^F}) : \mathbb{P}_j^F \in \mathbb{P}^F\}$.

We say that a convertible code is MDS if the initial and final code are both MDS. Similarly, a convertible code is linear if the initial and final code are both linear. Convertible codes were first studied in the context of their access cost. Access cost is the measure of the total number of nodes accessed in the process of conversion. Optimal bounds and constructions for the access cost of convertible codes is known for all parameters $n^I, k^I, n^F, k^F \in \mathbb{N}$ such that $n^I > k^I, n^F > k^F$ [9, 12].

In the interest of 'securing' of the message, these codes are insufficient since they reveal information about the message. We will devise codes that transform existing MDS convertible codes into ones that obfuscate the message.

**Example: Access Optimal** $[4, 2; 6, 4]$ **Convertible Code** Let $\theta \in \mathbb{F}$ be a primitive element. A MDS, systematic, access-optimal $[4, 2; 6, 4]$ convertible code $(\mathcal{C}^I, \mathcal{C}^F)$ has the following corresponding generator matrices $G^I$ and $G^F$:

$$G^I = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & \theta \end{bmatrix}, G^F = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & \theta \\ 0 & 0 & 1 & 0 & 1 & \theta^2 \\ 0 & 0 & 0 & 1 & 1 & \theta^3 \end{bmatrix}$$

Here, $\lambda^I = 2$ and $\lambda^F = 1$ i.e., there are 2 initial codewords and 1 final codewords. Let the message vectors underlying each initial codeword $\boldsymbol{m}_1^I, \boldsymbol{m}_2^I \in \mathbb{F}^2$ be defined as

$$\boldsymbol{m}_1^I = \begin{bmatrix} m_1 & m_2 \end{bmatrix},$$
$$\boldsymbol{m}_2^I = \begin{bmatrix} m_3 & m_4 \end{bmatrix}.$$

Define the message vector underlying the final codeword $\boldsymbol{m}_1^F \in \mathbb{F}^4$ as

$$\boldsymbol{m}_1^F = \begin{bmatrix} m_1 & m_2 & m_3 & m_4 \end{bmatrix}.$$

Then, the initial codewords are

$$\mathcal{C}^I(\boldsymbol{m}_1^I) = \begin{bmatrix} m_1 & m_2 & p_{1,2} & p'_{1,2} \end{bmatrix}$$
$$\mathcal{C}^I(\boldsymbol{m}_2^I) = \begin{bmatrix} m_3 & m_4 & p_{3,4} & p'_{3,4} \end{bmatrix},$$

where $p_{i,j} = m_i + m_j$ and $p'_{i,j} = m_i + m_j\theta$, for $i, j \in [4]$. In the access-optimal conversion procedure, only the 4 symbols of the initial codewords $p_{1,2}, p'_{1,2}, p_{3,4}$, and $p'_{3,4}$ are accessed to compute the final codeword

$$\mathcal{C}^F(\boldsymbol{m}_1^F) = \begin{bmatrix} m_1 & m_2 & m_3 & m_4 & p_{1,2} + p_{3,4} & p'_{1,2} + p'_{3,4}\theta^2 \end{bmatrix}.$$

Although not the focus of this thesis, we note that convertible codes have also been studied in the context of their bandwidth cost. The bandwidth cost of conversion is measured by the amount of data downloaded (download bandwidth) within a network of nodes performing conversion. In this setting, codeword symbols are stored at a finer granularity, where during conversion, only a fraction of the codeword symbol may be downloaded. Formally, the initial and final codes of the convertible codes are vector codes, which are codes over finite extension of finite fields $\mathbb{F}_{q^\alpha}$ for some $\alpha \in \mathbb{N}$. Within the *merge* regime ($k^F = \lambda^I k^I, \lambda^I \geq 2$), there are optimal constructions employing piggyback codes [11, 16]. In the *split* regime ($k^I = \lambda^F k^F, \lambda^I \geq 2$), constructions employ a similar piggyback code technique, where these codes are optimal assuming a plausible conjecture on the bandwidth lower bound of convertible codes [10].

## 2.4 Wiretap Channels

The *wiretap channel* was first introduced by Wyner in [20], where a transmitter Alice sends messages to receiver Bob across a discrete, memory-less channel (DMC) in the presence of an

eavesdropper Eve, who observes a partial view output via another DMC. Wyner derives a tradeoff between the maximum rate of information that Alice can convey and the allowed amount of eavesdropping such that Alice's communication remains perfectly secret to Eve.

An extension of the wire-tap channel, known as the *wire-tap channel II*, was studied by Ozarow and Wyner in [14], where there is additional restrictions fixes Alice's message and encoding lengths. That is, Alice must convey a $k$ symbol message to Bob over the channel with an $n$ symbol transmission, where Eve can observe $\ell < n$ symbols. Ozarow and Wyner derive a similar upper bound on the information conveyed as in the previous wire-tap channel setting, but they also show an explicit construction matching the maximum information rate while remaining perfectly secret to any eavesdropper. Their construction is based on transmitting the cosets of a chosen *group code*, which will appear uniformly random to an eavesdropper who can only see a partial number of symbols. Since we adapt this technique for our setting of convertible codes, we delve into the technicalities in the following subsection 2.4.1.

Further, Subramanian and McLaughlin in [19] study the *Erasure-Erasure wire-tap channel*, a further extension of the wire-tap channel II with additional erasures $\mu$ in receiver Bob's view of Alice's sent transmission. Note that an Erasure-Erasure channel is a wire-tap II channel when $\mu = 0$. They construct a *nested code*, which is based on the coset encoding of Ozarow and Wyner, with an additional concatenation of an erasure code. Likewise, nested coding will be useful in constructing secure convertible codes, so we provide their background theory in the following subsection 2.4.1.

### 2.4.1 Coset Binning and Nested Codes

Coset and nested coding used in the wiretap II channel[14] and the Erasure-Erasure channel[19] setting will be integral to our construction of secure convertible codes. We compile their relevant techniques in detail.

First, we look at coset codes in the wiretap II setting. Suppose Alice has a $k$ symbol message $\boldsymbol{m} \in \mathbb{F}^k$ she wants to transmit to Bob via an $n$ symbol coded message $\boldsymbol{x} \in \mathbb{F}^n$ through a perfect channel, where Eve may observe any $\ell < k$ symbols of her coded message. Alice's objective is to choose a $n$ symbol encoding scheme that achieves perfect secrecy; that is, Eve does not gain any information Alice's underlying message. We can state the requirements of the wiretap II setting in information theoretic terms.

**Definition 2** (Wiretap II Secure [14])**.** A code $\mathcal{C} \subseteq \mathbb{F}^n$ is $(k, n, \ell)$-Wiretap II Secure if for any uniformly random chosen $\boldsymbol{m} \in \mathbb{F}^k$ and $\boldsymbol{x} = \mathcal{C}(\boldsymbol{m})$,

$$\begin{aligned} H(\boldsymbol{\mathcal{S}} \mid \boldsymbol{\mathcal{X}}_{\mathbb{E}}) &= H(\boldsymbol{\mathcal{S}}), && (\forall \mathbb{E} \subset [n], |\mathbb{E}| \leq \ell) \\ H(\boldsymbol{\mathcal{S}} \mid \boldsymbol{\mathcal{X}}) &= 0. \end{aligned}$$

The first equation ensures that any $\ell$ symbols do not reveal anything about a message $\boldsymbol{m}$ chosen uniformly at random, and the second ensures that the entire message is recoverable when reading the entire coded message $\boldsymbol{x}$. This is possible by employing the use of cosets as encodings of messages. Informally, a partial view of a coset vector admits candidate matches across different cosets, where each candidate coset contains an equal number of candidate vectors. Further,

the number of candidate cosets will be equal to $2^k$, the number of possible messages. Hence, an eavesdropper will have no information, while the receiver will be able to decode the entire message. Formally,

**Lemma 3** (Coset Codes [14, 19])**.** For $k, n, \ell$ positive integers such that $k < n$ and $\ell < n$, there exists a code $\mathcal{C}_*$ that is $(k, n, \ell)$-Wiretap II secure.

*Proof.* Choose $\mathcal{C}_*$ to be an MDS $[n, n-k]$ code. Since $|\mathcal{C}_*| = q^{n-k}$, there are $q^k$ cosets of $\mathcal{C}_*$ and so there is a one-to-one mapping from cosets to messages. Suppose message $\boldsymbol{m}$ maps to coset $\boldsymbol{a} + \mathcal{C}_*$, for some $\boldsymbol{a} \in \mathbb{F}^n$. Then, $\boldsymbol{x}$ is chosen to be a uniformly at random chosen element of $\boldsymbol{a} + \mathcal{C}_*$. By the bijection from cosets to messages, $\boldsymbol{x}$ completely determines $\boldsymbol{m}$ implying $H(\boldsymbol{\mathcal{S}}|\boldsymbol{\mathcal{X}}) = 0$.

What is left to show is that for any $\mathbb{E} \subset [n]$ with$|\mathbb{E}| \leq \ell$, this encoding implies $H(\boldsymbol{\mathcal{S}}|\boldsymbol{\mathcal{X}}_{\mathbb{E}}) = H(\boldsymbol{\mathcal{S}})$. Let $\boldsymbol{a} \in \mathbb{F}^n$ be a vector which *matches* $\boldsymbol{x}$ at the indices that Eve sees. More formally, for all $i \in \mathbb{E}$, $\boldsymbol{a}_i = \boldsymbol{x}_i$.[1] Then, we can define the set of all matching vectors lying within the coset of $\boldsymbol{a}$ as $\boldsymbol{a} + \mathcal{C}_{[n]\setminus\mathbb{E}}$, where

$$\mathcal{C}_{[n]\setminus\mathbb{E}} = \{c \in \mathcal{C}_* : c_{\mathbb{E}} = 0\}.$$

Thus, number of matches in each coset is $\left|\mathcal{C}_{[n]\setminus\mathbb{E}}\right|$, where there are $\frac{q^{n-|\mathbb{E}|}}{\left|C_{[n]\setminus\mathbb{E}}\right|}$ such cosets. Thus,

$$\begin{aligned} H(\boldsymbol{\mathcal{S}}|\boldsymbol{\mathcal{X}}_{\mathbb{E}}) = \log_q \left( \frac{q^{n-|\mathbb{E}|}}{\left|\mathcal{C}_{[n]\setminus\mathbb{E}}\right|} \right) &= (n - |\mathbb{E}|) - \dim \mathcal{C}_{[n]\setminus\mathbb{E}} \\ &= n - |\mathbb{E}| - (n - k - |\mathbb{E}|) = H(\boldsymbol{\mathcal{S}}). \end{aligned}$$

□

We move onto the Erasure-Erasure channel, where recall that this channel adds erasures within Bob's view of the encoded message $\boldsymbol{x}$. This addition is interesting due to push-and-pull between Bob's and Eve's goals. Now that Bob can only see a subset of the encoded message symbols, enforcing his ability to decode the original message $\boldsymbol{m}$ while enforcing information theoretic security in Eve's view lowers the amount of information Alice can securely convey to Bob.

We define (information theoretically) secure codes in the Erasure-Erasure channel with the addition of erasures in Bob's view.

**Definition 4** ((MDS) Erasure-Erasure Secure [19])**.** A code $\mathcal{C} \subseteq \mathbb{F}^n$ is $(k, n, \ell, \nu)$- Erasure-Erasure secure if for any uniformly random chosen $\boldsymbol{s} \in \mathbb{F}^{k_S}$ for some $k_S \leq k$, there exists $\boldsymbol{x} \in \mathcal{C}$ such that

$$\begin{aligned} H(\boldsymbol{\mathcal{S}} \mid \boldsymbol{\mathcal{X}}_{\mathbb{E}}) &= H(\boldsymbol{\mathcal{S}}), && (\forall \mathbb{E} \subset [n], |\mathbb{E}| \leq \ell) \\ H(\boldsymbol{\mathcal{S}} \mid \boldsymbol{\mathcal{X}}_{\mathbb{B}}) &= 0 && (\forall \mathbb{B} \subset [n], |\mathbb{B}| \geq \nu) \end{aligned}$$

If $\nu = k$, we say $\mathcal{C}$ is an MDS $(k, n, \ell)$-Erasure-Erasure secure code.

[1]For example, $\boldsymbol{x} = [0, 1, 2, 3, 4]$ and $\mathbb{E} = \{1, 3\}$. Then, $\boldsymbol{a} = [b_0, 1, b_2, 3, b_4]$ matches $\boldsymbol{x}$ for any $b_0, b_2, b_4 \in \mathbb{F}_5$.

The modification of the second equation enforces that in the case of any $n - \nu$ erasures, the original message is still decodable. As mentioned prior, the goal of recovering the message with only $\nu$ symbols and ensuring uniform randomness with any $\ell$ symbol will lower the number of message symbols that are secure in such a system. In other words, our messages of length $k$ will have reduced entropy. Using information theoretic arguments, we can get an upperbound on the secrecy capacity.

**Theorem 5** (Upperbound on Secrecy Capacity for Erasure-Erasure Channels [19])**.** For $k, n, \ell, \nu$ positive integers such that $\ell < \nu < k < n$, and code $\mathcal{C} : \mathbb{F}^k \to \mathbb{F}^n$, if $\mathcal{C}$ is a $(k, n, \ell, \nu)$-Erasure-Erasure secure code, then for any uniformly random $\boldsymbol{s} \in \mathbb{F}^{k_s}$ sent over the channel, $H(\boldsymbol{\mathcal{S}}) \leq \nu - \ell$. If $\mathcal{C}$ is a MDS $(k, n, \ell, \nu)$-Erasure-Erasure secure code, $H(\boldsymbol{\mathcal{S}}) \leq k - \ell$.

*Proof.* Suppose $\mathbb{E} \subset \mathbb{B} \subset [n]$ such that $|\mathbb{E}| = \ell$ and $|\mathbb{B}| = \nu$. Then,

$$\begin{aligned} H(\boldsymbol{\mathcal{S}}) &= H(\boldsymbol{\mathcal{S}}|\boldsymbol{\mathcal{X}}_{\mathbb{E}}) - H(\boldsymbol{\mathcal{S}}|\boldsymbol{\mathcal{X}}_{\mathbb{B}}) \\ &= H(\boldsymbol{\mathcal{S}}|\boldsymbol{\mathcal{X}}_{\mathbb{E}}) - H(\boldsymbol{\mathcal{S}}\boldsymbol{x}_{\mathbb{E}}, \boldsymbol{x}_{\mathbb{B}\setminus\mathbb{E}}) \\ &= I(\boldsymbol{\mathcal{S}}; \boldsymbol{\mathcal{X}}_{\mathbb{B}\setminus\mathbb{E}}|\boldsymbol{\mathcal{X}}_{\mathbb{E}}) \\ &\leq H(\boldsymbol{\mathcal{X}}_{\mathbb{B}\setminus\mathbb{E}}|\boldsymbol{\mathcal{X}}_{\mathbb{E}}) \\ &\leq H(\boldsymbol{\mathcal{X}}_{\mathbb{B}\setminus\mathbb{E}}) \\ &\leq \nu - \ell \end{aligned}$$

□

The intuition of the above bound is as follows: if Bob chooses some set of $\nu$ symbols to recover the message $S$, the worst case is when Eve chooses all her $\ell$ symbols from Bob's recovery set. In this case, the information conveyed by the $\nu$ symbols must be reduced by at least $\ell$.

By the previous theorem, if $\mathcal{C}$ is an MDS $(k, n, \ell, \nu)$-Erasure-Erasure secure code, then Bob can store at most $k_s \in \mathbb{N}$ symbols securely, for $k_s \leq k - \ell$. There are Erasure-Erasure Secure codes that reach the upper bound on the secrecy capacity derived in theorem 5. While coset codes are not Erasure-Erasure secure because cosets do not have any erasure correction guarantees in general (intuitively, we could not even recover from one erasure of a coset code since the encoding looks uniformly random until all $n$ symbols are read), they are an important building block for a suitable construction known as the *nested code*.

**Definition 6** (Nested Code [19])**.** An MDS $[n, k]$ code $\mathcal{C}$ is a $\ell$-nested code if it has a generator matrix $G = \begin{bmatrix} G_{\boldsymbol{s}} \\ G_{\boldsymbol{\kappa}} \end{bmatrix} \in \mathbb{F}^{k \times n}$, where $G_{\boldsymbol{\kappa}} \in \mathbb{F}^{\ell \times n}$ is a generator matrix of a MDS code.

First, a new message vector of length $k$ is constructed comprising of the message symbols to be encoded, $\boldsymbol{s} \in \mathbb{F}^{k-\ell}$, and some *masking* symbols, $\boldsymbol{\kappa} \in \mathbb{F}^{\ell}$, where each masking symbol is chosen uniformly at random over $\mathbb{F}$. Let $\boldsymbol{m} = \begin{bmatrix} \boldsymbol{s} & \boldsymbol{\kappa} \end{bmatrix}$ be this message vector. Then, its encoding is $\boldsymbol{m}G = \boldsymbol{s}G_{\boldsymbol{s}} + \boldsymbol{\kappa}G_{\boldsymbol{\kappa}}$. One can verify that no information about the secure message symbols $\boldsymbol{s}$ is leaked from any $j < \ell$ codeword symbols due to the addition of the encoding of the masked symbols $\boldsymbol{\kappa}G_{\boldsymbol{\kappa}}$.

For intuition, one can view nested codes as a careful modification of coset codes that enforces the MDS property. Again, let $\mathcal{C}_\kappa$ be a $[n, n-k]$ code. Each coset codeword $\boldsymbol{a} + \boldsymbol{x}$, where $\boldsymbol{a} \in \mathbb{F}^n$

and $\boldsymbol{x} \in \mathcal{C}_\kappa$, is a vector within the coset corresponding to $\boldsymbol{a}$. Bob can then decode the coset codeword into the vector $\boldsymbol{a}$ since its coset is unique, and Bob can see all the coset codeword symbols. In the Erasure-Erasure setting, Bob only has a partial view, which we can treat as erasures in the coset codeword. Thus, to ensure Bob can decode the message, we form our cosets over codewords of an erasure code $\mathcal{C}_S$ i.e., $\boldsymbol{a} = \mathcal{C}_S(\boldsymbol{s})$. Changing the message vector from an arbitrary vector $\boldsymbol{a} \in \mathbb{F}^n$ to a codeword $\boldsymbol{a} \in \mathcal{C}_S$ lowers the size of the message space Alice may convey to Bob securely, and the secrecy capacity upper bound in Theorem 5 states at least how much information is lost. We formally prove this intuition and show that nested codes match the secrecy capacity upper bound below using a similar argument in Lemma 3.

**Lemma 7** (Nested Codes [19])**.** For any $k, n, \ell$ positive integers such that $\ell < k < n$, there exists an MDS $(k, n, \ell)$-Erasure-Erasure secure code reaching secrecy capacity.

*Proof.* Let $\mathcal{D} = \mathcal{C}_S + \mathcal{C}_*$ be an MDS $[n, k]$ code, where $\mathcal{C}_S$ is a $[n, k-\ell]$ code and $\mathcal{C}_*$ be a MDS $[n, \ell]$ code (i.e. $\mathcal{D}$ is a $\ell$-nested $[n, k]$ code). Further, suppose $\mathcal{C}_S \cap \mathcal{C}_* = \{0\}$. Let $G_S, G_*$, and $G$ be the generator matrices of $\mathcal{C}_S$, $\mathcal{C}_*$, and $\mathcal{D}$ respectively. Let the message be $\boldsymbol{m} = \begin{bmatrix} \boldsymbol{s} & \boldsymbol{\kappa} \end{bmatrix} \in \mathbb{F}^k$, where $\boldsymbol{s} \in \mathbb{F}^{k-\ell}$ and $\boldsymbol{\kappa} \in \mathbb{F}^\ell$. Encode the message under $\mathcal{D}$,

$$\mathcal{D}\left(\begin{bmatrix} \boldsymbol{s} & \boldsymbol{\kappa} \end{bmatrix}\right) = \begin{bmatrix} \boldsymbol{s} & \boldsymbol{\kappa} \end{bmatrix} G = \boldsymbol{s} G_S + \boldsymbol{\kappa} G_*.$$

and let $\boldsymbol{x} = D\left(\begin{bmatrix} \boldsymbol{s} & \boldsymbol{\kappa} \end{bmatrix}\right)$. Since $\boldsymbol{x}$ is an element of some coset of $\mathcal{C}_*$, we can use the same analysis in lemma 3 to show that for any $\mathbb{J} \subset [n]$ of revealed indices, we have

$$H(\boldsymbol{\mathcal{S}} \mid \boldsymbol{\mathcal{X}}_\mathbb{J}) = \dim \mathcal{D}_{[n]\setminus\mathbb{J}} - \dim \mathcal{C}_{*[n]\setminus\mathbb{J}}{}^2,$$

where for any $\mathbb{B} \subset [n]$ of size $|B| = k$,

$$H(\boldsymbol{\mathcal{S}} \mid \boldsymbol{\mathcal{X}}_\mathbb{B}) = \dim D_{[n]\setminus\mathbb{B}} - \dim \mathcal{C}_{*[n]\setminus\mathbb{B}} = (k-k) - 0 = 0.$$

Note that we use the fact that for any MDS $[n, k]$ code $\mathcal{C}$, $\dim \mathcal{C}_{[n]\setminus\mathbb{B}} = \max\{0, k - |J|\}$. Lastly, for any $\mathbb{E} \subset [n]$ of size $|\mathbb{E}| = \ell$,

$$\dim \mathcal{D}_{[n]\setminus\mathbb{E}} - \dim \mathcal{C}_{*[n]\setminus\mathbb{E}} = (k-\ell) - (\ell-\ell) = k - \ell = H(\boldsymbol{\mathcal{S}}). \qquad \square$$

Nested codes can be constructed from Reed Solomon (RS) codes. That is, for a $[n, k]$ $\ell$-nested code, we can take overall code $\mathcal{D}$ to be a $[n, k]$ RS code, where $\mathcal{C}_{\boldsymbol{\kappa}}$ is the $[n, \ell]$ RS code. The following illustrates an example construction for the special case of an $\ell$-nested code when $n = k$, which we will use for our secure convertible code construction:

**Example: A $[4, 4]$ 2-Nested Code** Consider a nested coding for $n = k = 4, \ell = 2$. Suppose $\mathcal{D}$ is the MDS 2-nested code over $\mathbb{F}_5$ with generator matrix $G$ defined as

$$G = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 1 & 2 & 0 & 1 \end{bmatrix}.$$

[2]There are a total of $\left|D_{[n]\setminus J}\right|$ matched cosets for $\boldsymbol{x}_\mathbb{J}$. Each matched coset will have $\left|\mathcal{C}_{*[n]\setminus J}\right|$ elements, where the number of cosets to consider are $\frac{\left|D_{[n]\setminus J}\right|}{\left|\mathcal{C}_{*[n]\setminus J}\right|}$.

The secrecy capacity is $4-2=2$. The secure message symbols are $s_1, s_2 \in \mathbb{F}_4$ and the (uniformly random) masking symbols are $\kappa_1, \kappa_2 \in \mathbb{F}_5$. Let the message be $\boldsymbol{m} = \begin{bmatrix} s_1 & s_2 & \kappa_1 & \kappa_2 \end{bmatrix}$.

Then $\mathcal{D}(\boldsymbol{m}) = \begin{bmatrix} s_1 + \kappa_{1,1} & s_2 + \kappa_{1,2} & \kappa_1 & \kappa_2 \end{bmatrix}$, where $\kappa_{i,j} = i\kappa_1 + j\kappa_2$. Any eavesdropper reading any $2$ symbols learns nothing about the secure message symbols $s_1$ and $s_2$.

# Chapter 3

# Modeling Eavesdropper-Secure Convertible Codes

In this chapter, we define secure convertible codes by enhancing the existing convertible code framework with protection against eavesdroppers. A message $\boldsymbol{m} \in \mathbb{F}^k$ is stored on the convertible code, resulting in an initial/final configuration comprising of initial/final codewords. Now, suppose that an eavesdropper gains read-access to any $\ell < \min\{k^I, k^F\}$ codeword symbols, spanning across initial and final configuration, as illustrated in Figure 3.1.

Note that the eavesdropper may choose to read some out of the $\ell$ compromised symbols from the initial configuration and/or wait for the conversion to occur to choose the remaining compromised symbols from the final configuration. Also, note that this secrecy model captures the special case when the codeword symbols downloaded during the conversion process are compromised, in the access-cost setting. This scenario is identical to the case of eavesdropper reading the corresponding codeword symbols in the initial configuration.

We introduce additional notation to formally define the desired properties of secure convertible codes under passive eavesdroppers. Let $\boldsymbol{s} \in \mathbb{F}^{k_S}$ be the message symbols to be securely stored for some $k_S \in \mathbb{N}$. Let $\boldsymbol{\mathcal{S}}$ be the corresponding random variable, which is assumed to be uniformly distributed over $\mathbb{F}^{k_S}$ representing (incompressible) data. Hence $H(\boldsymbol{\mathcal{S}}) = k_S$.

Next, we introduce notation to specify the partition of the secure message symbols into the

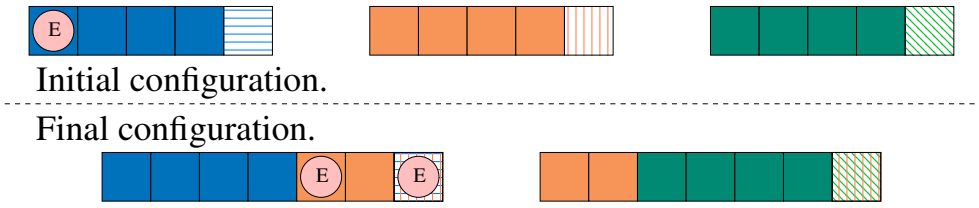


Figure 3.1: A $[5, 4; 7, 6]$ convertible code with 3 symbols read by the eavesdropper (pink circles). Initial codewords are on the top of the diagram while the final codewords are on the bottom of the diagram. The initial/final codewords make up the initial/final configuration. Note that an eavesdropper can choose to read symbols across different initial and final codewords, or they can read all the symbols from a single codeword.

initial and final codewords. Let $\mathbb{S}^I, \mathbb{S}^F \in \Pi(k_S)$, where $|\mathbb{S}^I| = \lambda^I$ and $|\mathbb{S}^F| = \lambda^F$ denote *secure symbol partitions* that specify the mapping of secure message symbols into the initial and final codewords. Likewise, for $i \in [\lambda^I]$ let $\boldsymbol{\mathcal{S}}_i^I$ be the random variable corresponding to the secure message symbols of $i$'th initial codeword, and for $j \in [\lambda^F]$, let $\boldsymbol{\mathcal{S}}_j^F$ be the random variables corresponding to the secure message symbols of the $j$'th final codeword.

In traditional convertible codes, initial and final configurations are implicitly defined as the collection of their corresponding codewords. However, to analyze an eavesdropper who may read symbols across multiple codewords during the conversion (see figure 3.1), it is more convenient to define these configurations as vectors. Let $\boldsymbol{x}^I \in \mathbb{F}^{\lambda^I n^I}$ represent the vector consisting of all the codewords in the initial configuration (in the implicit ordering specified by the Convertible code), and $\boldsymbol{x}^F \in \mathbb{F}^{\lambda^F n^F}$ represent the same for the final configuration. Then, for each $i \in [\lambda^I]$, the vector $\boldsymbol{x}_i^I \in \mathbb{F}^{n^I}$ is the $i$'th initial codeword and for each $j \in [\lambda^F]$ the vector $\boldsymbol{x}_j^F \in \mathbb{F}^{n^F}$ is the $j$'th final codeword.

**Definition 8.** A $(\ell, k_S)-$Secure $[n^I, k^I; n^F, k^F]$ convertible code is a $[n^I, k^I; n^F, k^F]$ convertible code that can store a message $\boldsymbol{s} \in \mathbb{F}^{k_S}$ satisfying following decoding and secrecy properties:

1. **Decoding (MDS property).** For each $i \in [\lambda^I]$ and any subset $\mathbb{B} \subset [n^I]$ of size $k^I$,

$$H(\boldsymbol{\mathcal{S}}_i^I | \boldsymbol{\mathcal{X}}_{i,\mathbb{B}}^I) = 0,$$

   and for each $j \in [\lambda^F]$ and any subset $\mathbb{B} \subset [n^F]$ of size $k^F$,

$$H(\boldsymbol{\mathcal{S}}_j^F | \boldsymbol{\mathcal{X}}_{j,\mathbb{B}}^F) = 0.$$

2. $\ell$**-Secrecy.** For any $\mathbb{E}^I \subset [\lambda^I n^I]$, $\mathbb{E}^F \subset [\lambda^F n^F]$ of combined size $\left|\mathbb{E}^I\right| + \left|\mathbb{E}^F\right| \leq \ell$,

$$H(\boldsymbol{\mathcal{S}} \mid \boldsymbol{\mathcal{X}}_{\mathbb{E}^I}^I, \boldsymbol{\mathcal{X}}_{\mathbb{E}^F}^F) = H(\boldsymbol{\mathcal{S}}).$$

As in traditional convertible codes, the access cost is measured by the number of initial symbols accessed in the conversion procedure. We are interested in secure convertible codes that maximize $k_S$ and minimize access cost simultaneously. In the following section, we prove the information-theoretic upper bound on the number of secure message symbols that be stored using a convertible code. For $(\ell, k_S)$-secure convertible codes that reach the upper bound on secrecy capacity, we drop the $k_S$ from the notation, simply denoting them as optimal $\ell$*-secure convertible codes.*

# Chapter 4

# Secrecy Upper Bound and Optimal Construction for Secure Convertible Codes

## 4.1 Upper Bound on the Secrecy Capacity of Convertible Codes

In order to derive an upper bound on the secrecy capacity, we first address a necessary nuance of $\ell$-secure convertible codes. In this model, an eavesdropper is given the highest level of flexibility, where she can choose any symbol within the initial or final configuration to access. In particular, she may choose to read only the symbols of an individual codeword. Thus, in order for $\ell$-secrecy to hold for the overall convertible code, *each codeword* must be secure to $\ell$ eavesdroppers. This intuition is captured in the following lemma.

**Lemma 9.** For any $(\ell, k_S)$-secure $[n^I, k^I; n^F, k^F]$ convertible code with secure message symbols $\boldsymbol{s} \in \mathbb{F}^{k_S}$, the following hold:

1. **Initial codeword Secrecy**: For any $i \in [\lambda^I]$ and subset $\mathbb{E}_i^I \subset [n^I]$ of size $\ell$, we have

$$H(\boldsymbol{\mathcal{S}}_i^I | \boldsymbol{\mathcal{X}}_{i,\mathbb{E}_i^I}^I) = H(\boldsymbol{\mathcal{S}}_i^I).$$

2. **Final codeword Secrecy**: For any $j \in [\lambda^F]$ and subset $\mathbb{E}_j^F \subset [n^F]$ of size $\ell$,

$$H(\boldsymbol{\mathcal{S}}_j^F | \boldsymbol{\mathcal{X}}_{j,\mathbb{E}_j^F}^F) = H(\boldsymbol{\mathcal{S}}_j^F).$$

*Proof.* This follows from Definition 8. □

Lemma 9 is used to derive the upper bound on the number of secure message symbols $k_S$ for a $\ell$-secure convertible codes.

**Theorem 10.** For positive integers $k^I, n^I, k^F, n^F, \ell, k_S$ such that $k^I \leq n^I, k^F \leq n^F, \ell < \min\{k^I, k^F\}$, any $(\ell, k_S)$-secure $[n^I, k^I; n^F, k^F]$ convertible code storing $\boldsymbol{s} \in \mathbb{F}^{k_S}$ satisfies

$$H(\boldsymbol{\mathcal{S}}) \leq \min\{\lambda^I(k^I - \ell), \lambda^F(k^F - \ell)\}.$$

*Proof.* Suppose $k^I \leq k^F$. Fix $i \in [\lambda^I]$ and suppose $\mathbb{E} \subset \mathbb{B} \subset [n^I]$ such that $|\mathbb{E}| = \ell$ and $|\mathbb{B}| = k^I$. Then,

$$H(\boldsymbol{\mathcal{S}}) = \sum_{i=1}^{\lambda^I} H(\boldsymbol{\mathcal{S}}_i^I) \leq \lambda^I(k^I - \ell).$$

where the last inequality follows from

$$\begin{aligned} H(\boldsymbol{\mathcal{S}}_i^I) &= H(\boldsymbol{\mathcal{S}}_i^I|\boldsymbol{\mathcal{X}}_{i,\mathbb{E}}) - H(\boldsymbol{\mathcal{S}}_i^I|\boldsymbol{\mathcal{X}}_{i,\mathbb{B}}) && \text{(Lemma 9)} \\ &= H(\boldsymbol{\mathcal{S}}_i^I|\boldsymbol{\mathcal{X}}_{i,\mathbb{E}}) - H(\boldsymbol{\mathcal{S}}_i^I|\boldsymbol{\mathcal{X}}_{i,\mathbb{E}}, \boldsymbol{\mathcal{X}}_{i,\mathbb{B}\setminus\mathbb{E}}\} \\ &= I(\boldsymbol{\mathcal{S}}_i^I; \boldsymbol{\mathcal{X}}_{i,\mathbb{B}\setminus\mathbb{E}}|\boldsymbol{\mathcal{X}}_{i,\mathbb{E}}) \\ &\leq H(\boldsymbol{\mathcal{X}}_{i,\mathbb{B}\setminus\mathbb{E}}|\boldsymbol{\mathcal{X}}_{i,\mathbb{E}}) \leq H(\boldsymbol{\mathcal{X}}_{i,\mathbb{B}\setminus\mathbb{E}}) \leq (k^I - \ell), \end{aligned}$$

Suppose $k^F < k^I$. Fix $j \in [\lambda^F]$ and suppose $\mathbb{E} \subset \mathbb{B} \subset [n^F]$ such that $|\mathbb{E}| = \ell$ and $|\mathbb{B}| = k^F$. Then, symmetric to the previous case, we have $H(\boldsymbol{\mathcal{S}}_j^F) \leq (k^F - \ell)$ and $H(\boldsymbol{\mathcal{S}}) \leq \lambda^F(k^F - \ell)$. Putting the cases together, we have our desired bound. □

Note that $\lambda^I(k^I - \ell) \leq \lambda^F(k^F - \ell)$ if and only if $\lambda^I \geq \lambda^F$ i.e., the upperbound on the secrecy capacity is determined by whether there are more initial codewords ($\lambda^I \geq \lambda^F$) or there are more final codewords ($\lambda^I < \lambda^F$).

The intuition for the secrecy capacity upperbound is the tension between the information needed for decoding and the information hidden by $\ell$-secrecy. First, the MDS property of the initial code implies that any $k^I$ initial codeword symbols of an initial codeword is sufficient for decoding the underlying $k^I$ message symbols. An eavesdropper reading $\ell$ codeword symbols can get at most $\ell$ symbols worth of information that, in the worst case, directly overlaps with the $k^I$ message symbols, so at most $k^I - \ell$ of these symbols may be meaningful. Since the same holds for the final codewords, we have our secrecy capacity upperbound.

For intuition, we interpret the secrecy capacity upper bound in the context of the Erasure-Erasure channel. In the secure convertible code model, $\lambda^I$ codewords from the $[n^I, k^I]$ initial code $\mathcal{C}^I$ and $\lambda^F$ codewords from the $[n^F, k^F]$ final code $\mathcal{C}^F$ are sent through an Erasure-Erasure channel. Since these encodings are a concatenation of MDS code codewords, the decoder must chooses $\lambda^I$ (resp. $\lambda^F$) subsets of each initial (resp. final) codeword. The best an eavesdropper can do is to choose to eavesdrop all their $\ell$ symbols in a particular codeword's subset. Thus, a secure convertible code is only possible if it handles $\ell$-eavesdropped symbols on each codeword.

Note that an interesting extension of the above intuition is to consider if knowing the number of symbols that can be compromised per codeword would improve the secrecy capacity. We denote the maximum information stored in this setting as *fine-grained secrecy capacity*. Likewise, codes which satisfy this setting are denoted as *fine-grained secure convertible codes*. Fine-grained secure conversion is explored further in chapter 5.

In the next section, we show that it is possible to construct secure convertible codes for all valid parameters by bootstrapping existing access-optimal convertible codes with nested codes.

## 4.2 A Construction for General Parameters

In the context of securing convertible codes, we apply a concatenated code, where the outer code

is a nested code, and the inner code is the initial code of the convertible code. Intuitively, the nested code applies secrecy onto the message before it is stored on a convertible code. After conversion, the applied secrecy from the nested code will be present in the final codewords.

**Example 1: 1-secure $[5, 4; 7, 6]$ Convertible Code** Consider a 1-secure $[5, 4; 7, 6]$ convertible code over $\mathbb{F}_7$. Here, $\lambda^I = 3$ and $\lambda^F = 2$, so the upper bound on the secrecy capacity is $\min\{3(4-1), 2(6-1)\} = 9$. Let $\boldsymbol{s} = s_1 \dots s_9 \in \mathbb{F}_7^9$ be the secure message symbols and let $\kappa \in \mathbb{F}_7$ be a masking symbol. Consider the MDS 1-nested $[4, 4]$ code $\mathcal{D}^I$ with generator matrix

$$G = \begin{bmatrix} G_s \\ G_\kappa \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}.$$

Then, set the message vectors in the initial configuration $\boldsymbol{m}_1^I, \boldsymbol{m}_2^I, \boldsymbol{m}_3^I \in \mathbb{F}_7^4$ as

$$\begin{aligned} \boldsymbol{m}_1^I &= \mathcal{D}^I\left(\begin{bmatrix} s_1 & s_2 & s_3 & \kappa \end{bmatrix}\right) = \begin{bmatrix} \hat{s_1} & \hat{s_2} & \hat{s_3} & \kappa \end{bmatrix}, \\ \boldsymbol{m}_2^I &= \mathcal{D}^I\left(\begin{bmatrix} s_4 & s_5 & s_6 & \kappa \end{bmatrix}\right) = \begin{bmatrix} \hat{s_4} & \hat{s_5} & \hat{s_6} & \kappa \end{bmatrix}, \\ \boldsymbol{m}_3^I &= \mathcal{D}^I\left(\begin{bmatrix} s_7 & s_8 & s_9 & \kappa \end{bmatrix}\right) = \begin{bmatrix} \hat{s_7} & \hat{s_8} & \hat{s_9} & \kappa \end{bmatrix}, \end{aligned}$$

where $\hat{s_i} = s_i + \kappa$. An eavesdropper reading any 1 symbol learns nothing about the secure message symbols; either they read masking symbol $\kappa$ or an obfuscated secure symbol $s_i + \kappa$. Note that the secure message symbols in any initial configuration message vector $\boldsymbol{m}_i^I$ can be decoded by reading all 4 of its symbols, following from the MDS property of $\mathcal{D}^I$.

Let $(\mathcal{C}^I, \mathcal{C}^F)$ be an access-optimal $[5, 4; 7, 6]$ convertible code. The initial configuration codeword is set to

$$\boldsymbol{x}^I = \left(\mathcal{C}^I\left(\boldsymbol{m}_1^I\right), \mathcal{C}^I\left(\boldsymbol{m}_2^I\right), \mathcal{C}^I\left(\boldsymbol{m}_3^I\right)\right),$$

Using the conversion procedure of the convertible code $\left(\mathcal{C}^I, \mathcal{C}^F\right)$ on the initial codewords results in final codewords $\mathcal{C}^F\left(\boldsymbol{m}_1^F\right), \mathcal{C}^F\left(\boldsymbol{m}_2^F\right)$, where the message vectors in the final configuration $\boldsymbol{m}_1^F, \boldsymbol{m}_2^F$ are defined as

$$\begin{aligned} \boldsymbol{m}_1^F &= \begin{bmatrix} \hat{s_1} & \hat{s_2} & \hat{s_3} & \hat{s_4} & \hat{s_5} & \kappa \end{bmatrix}, \\ \boldsymbol{m}_2^F &= \begin{bmatrix} \hat{s_6} & \hat{s_7} & \hat{s_8} & \hat{s_9} & \kappa & \kappa \end{bmatrix}. \end{aligned}$$

Again, in the final configuration, any 1 symbol that an eavesdropper reads is either a masking symbol or a masked secure symbol. Lastly, the secure message symbols can be decoded from any final codeword $j = 1, 2$.

**Example 2: 2-secure $[7, 6; 5, 4]$-Convertible Code** Consider a 2-secure $[7, 6; 5, 4]$ convertible code over $\mathbb{F}_7$. Here, $\lambda^I = 2$ and $\lambda^F = 3$, so the upper bound secrecy capacity is $\min\{2(6 -$

$2), 3(4-2)\} = 6$. Let $\boldsymbol{s} = s_1 \dots s_6 \in \mathbb{F}_7^6$ be the secure message symbols and let $\kappa_1, \kappa_2 \in \mathbb{F}_7^2$ be the masking symbols. Consider the MDS 2-nested $[4, 4]$ code $\mathcal{D}^F$ with generator matrix

$$G = \begin{bmatrix} G_s \\ G_\kappa \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 2 & 0 & 1 \end{bmatrix}.$$

Then, set the message vectors in the final configuration $\boldsymbol{m}_1^F, \boldsymbol{m}_2^F, \boldsymbol{m}_3^F \in \mathbb{F}_7^4$ as

$$\begin{aligned} \boldsymbol{m}_1^F &= \mathcal{D}^F\left(\begin{bmatrix} s_1 & s_2 & \kappa_1 & \kappa_2 \end{bmatrix}\right) = \begin{bmatrix} s_1 + \kappa_{1,1} & s_2 + \kappa_{1,2} & \kappa_1 & \kappa_2 \end{bmatrix}, \\ \boldsymbol{m}_2^F &= \mathcal{D}^F\left(\begin{bmatrix} s_3 & s_4 & \kappa_1 & \kappa_2 \end{bmatrix}\right) = \begin{bmatrix} s_3 + \kappa_{1,1} & s_4 + \kappa_{1,2} & \kappa_1 & \kappa_2 \end{bmatrix}, \\ \boldsymbol{m}_3^F &= \mathcal{D}^F\left(\begin{bmatrix} s_5 & s_6 & \kappa_1 & \kappa_2 \end{bmatrix}\right) = \begin{bmatrix} s_5 + \kappa_{1,1} & s_6 + \kappa_{1,2} & \kappa_1 & \kappa_2 \end{bmatrix}, \end{aligned}$$

where $\kappa_{i,j} = i\kappa_1 + j\kappa_2$. Observe that an eavesdropper reading any 2 symbols learn nothing about the secure message symbols $\boldsymbol{s}$. Note that the secure message symbols in any final configuration message vector $\boldsymbol{m}_j^F$ can be decoded by reading all 4 of its symbols, following from the MDS property of $\mathcal{D}^F$.

Let $(\mathcal{C}^I, \mathcal{C}^F)$ be an access-optimal $[7, 6; 5, 4]$ convertible code. The final configuration codeword is set to

$$\boldsymbol{x}^F = \left(\mathcal{C}^F\left(\boldsymbol{m}_1^F\right), \mathcal{C}^F\left(\boldsymbol{m}_2^F\right), \mathcal{C}^F\left(\boldsymbol{m}_3^F\right)\right).$$

Define $\boldsymbol{m}_i^I$ for each $i \in [\lambda^I]$ such that running the conversion procedure on the initial configuration codeword $\boldsymbol{x}^I = \left(\mathcal{C}^I(\boldsymbol{m}_i^I)\right)_{i \in [\lambda^I]}$ results in $\boldsymbol{x}^F$. For instance, we can define message vectors in the initial configuration $\boldsymbol{m}_1^I, \boldsymbol{m}_2^I \in \mathbb{F}_7^6$ as

$$\begin{aligned} \boldsymbol{m}_1^I &= \begin{bmatrix} s_1 + \kappa_{1,1} & s_2 + \kappa_{1,2} & s_3 + \kappa_{1,1} & s_4 + \kappa_{1,2} & \kappa_1 & \kappa_2 \end{bmatrix}, \\ \boldsymbol{m}_2^I &= \begin{bmatrix} s_5 + \kappa_{1,1} & s_6 + \kappa_{1,2} & \kappa_1 & \kappa_2 & \kappa_1 & \kappa_2 \end{bmatrix}, \end{aligned}$$

based on an access-optimal $[7, 6; 5, 4]$ convertible code with corresponding message partitions.

The following construction shows how to use the approach in the previous example for general parameters.

### 4.2.1 General Construction

By Theorem 10, the secrecy capacity is $k_S \leq \min\{\lambda^I(k^I - \ell), \lambda^F(k^F - \ell)\}$. Without loss of generality, suppose that $\lambda^I(k^I - \ell) \leq \lambda^F(k^F - \ell)$, which is equivalent to $\lambda^I \geq \lambda^F$.

**Construction 11.**

**Preliminaries**. Suppose the secure message symbols are $s_1, \dots, s_{\lambda^I(k^I - \ell)} \in \mathbb{F}$ and the redundant symbols are $\kappa_1, \dots, \kappa_\ell \in \mathbb{F}$. Further, let $\boldsymbol{s}_i^I = [s_{i(k-\ell)+1} \dots s_{(i+1)(k-\ell)}]$ for $i \in [\lambda^I]$ and $\boldsymbol{\kappa} = [\kappa_1 \dots \kappa_\ell]$. By [12], there exists an access optimal $[n^I, k^I; n^F, k^F]$ convertible code $(\mathcal{C}^I, \mathcal{C}^F)$, which will be used in the construction. Next, let $\mathcal{D}^I$ be an MDS $\ell$-nested $[k^I, k^I]$ code with generator $G^I = \begin{bmatrix} G_s^I \\ G_\kappa^I \end{bmatrix}$ where $G_\kappa^I \in \mathbb{F}^{\ell \times k^I}$ is the generator of an MDS $[k^I, \ell]$ code and $G_s^I$

is defined as $G_s^I = \left[\begin{array}{c|c} \boldsymbol{I}_{k^I-\ell} & \mathbf{0} \end{array}\right]$, where $\boldsymbol{I}_{k^I-\ell}$ is the identity matrix of size $k^I - \ell$ and $\mathbf{0} \in \mathbb{F}^{k^I \times \ell}$ is the all-zeros matrix. It is not hard to confirm that $\mathcal{D}^I$ is MDS i.e., $G^I$ is invertible.

**Encoding in the initial configuration**. Form the $i$'th message vector in the initial configuration $\boldsymbol{m}_i^I$ for each $i \in [\lambda^I]$ as $\boldsymbol{m}_i^I = \mathcal{D}^I\left(\begin{bmatrix}\boldsymbol{s}_i^I & \boldsymbol{\kappa}\end{bmatrix}\right) = \boldsymbol{s}_i^I G_s^I + \boldsymbol{\kappa} G_\kappa^I$. Next, form the initial configuration codeword as $\boldsymbol{x}^I = (\mathcal{C}^I(\boldsymbol{m}_i^I))_{i\in[\lambda^I]}$.

**Decoding in the initial configuration**. To decode any initial codeword $\mathcal{C}^I(\boldsymbol{m}_i^I)$, use the decoding algorithm for $\mathcal{C}^I$, then apply the decoding algorithm for $\mathcal{D}^I$ (apply the inverse of its generator matrix $G^{-1}$).

**Code conversion** The final configuration (and final codewords) is constructed by running the conversion procedure of the underlying convertible code as-is to obtain $\boldsymbol{x}^F = (\mathcal{C}^F(\boldsymbol{m}_j^F))_{j\in[\lambda^F]}$, where $\boldsymbol{m}_j^F$ are the message vectors in the final configuration.

**Decoding in the final configuration**. To decode all secure message symbols of a given final codeword $j \in [\lambda^F]$: 1) apply the decoder of $\mathcal{C}^F$ to recover $\boldsymbol{m}_j^F$, 2) recover $\boldsymbol{\kappa}$ from $\boldsymbol{m}_j^F$, 3) Each message symbol in the final configuration has at most one secure symbol $s_{iq}$ to decode, where $i \in [\lambda^I]$, and $q \in [k^I - \ell]$. For each message symbol $(\boldsymbol{m}_j^F)_p$, where $p \in [k^F]$, corresponding to a unique secure symbol $s_{iq}$ (as will be shown below), output $(m_j^F)_p - (\boldsymbol{\kappa} G_\kappa)_q$. In Theorem 12, we prove that each assertion is valid and this procedure always correctly decodes $s_{iq}$.

When $\lambda^I < \lambda^F$, the construction follows along similar lines. In this case, we start the construction by defining the final configuration and then work backwards. Form $\boldsymbol{s}_j^F$ for $j \in [\lambda^F]$, message vectors in the final configuration codeword $\boldsymbol{m}_j^F$, and final configuration codeword $\boldsymbol{x}^F = (\mathcal{C}^F(\boldsymbol{m}_j^F))_{j\in[\lambda^F]}$ similarly. Then, define $\boldsymbol{m}_i^I$ for each $i \in [\lambda^I]$ such that running the conversion procedure on the initial configuration codeword $\boldsymbol{x}^I = (\mathcal{C}^I(\boldsymbol{m}_i^I))_{j\in[\lambda^F]}$ results in $\boldsymbol{x}^F$.

We prove that our construction is an optimal secure convertible code with optimal access cost.

**Theorem 12.** For any integers $n^I, n^F, k^I, k^F$ such that $0 \le k^I \le n^I, 0 \le k^F \le n^F$, and $\ell < \min\{k^I, k^F\}$, construction 11 is an optimal $\ell$-secure $[n^I, k^I; n^F, k^F]$ convertible code with optimal access cost.

*Proof.* Without loss of generality, consider the construction when $\lambda^I \ge \lambda^F$. First, the initial codewords are decodable and $\ell$-secure by construction of the initial configuration. Decodability follows from the MDS property of codes $\mathcal{C}^I$ and $\mathcal{D}^I$. Each initial codeword $\mathcal{C}^I(\boldsymbol{m}_i^I)$ is $\ell$-secure since each message vector in the initial configuration $\boldsymbol{m}_i^I$ are codewords of code $\mathcal{D}^I$, which is an $\ell$-secure code. Next, the final codewords $\mathcal{C}^F(\boldsymbol{m}_j^F)$ retain $\ell$-secrecy. If not, this implies that the message vectors in the final configuration $\boldsymbol{m}_j^F$ do not have $\ell$-secrecy because the contrapositive, that messages with $\ell$-secrecy imply their encodings have $\ell$-secrecy, is true. Then, there is some subset of message symbols in the final configuration of size less than $\ell$ that reveal nonzero information about the secure message symbols. However, these message symbols in the final configuration were originally message symbols in the initial configuration and since initial messages are codewords of $\mathcal{D}^I$, there exists a subset of less than $\ell$ codeword symbols that reveal information about the secure message symbols, contradicting the $\ell$-secrecy of code $\mathcal{D}^I$.

It is left to show that each final codeword $\mathcal{C}^F(\boldsymbol{m}_j^F)$ is decoded correctly by our specified algorithm. In step 1, $\boldsymbol{m}_j^F$ is recovered by the decoder of $\mathcal{C}^F$. Next, step 2 is always possible i.e.,

every message vector in the final configuration contains a copy of $\boldsymbol{\kappa}$. Each final codeword will contain all symbols of some message vector in the initial configuration $\boldsymbol{m}_i^I$ due to properties of the access-optimal convertible codes constructed by Maturana et al. [12]. In their construction, for all $\mathbb{P}_j^F \in \mathbb{P}^F$, there is a $\mathbb{P}_i^I \in \mathbb{P}^I$ such that $\mathbb{P}_i^I \subset \mathbb{P}_j^F$. Thus, each final codeword has all symbols of some message vector in the initial configuration $\boldsymbol{m}_i^I = \mathcal{D}^I\left(\begin{bmatrix} \boldsymbol{s}_i^I & \boldsymbol{\kappa} \end{bmatrix}\right)$. Thus, since $\mathcal{D}^I$ is MDS, we can recover $\boldsymbol{\kappa}$.

For step 3, we first show that each message symbol in the final configuration symbol $(m_j^F)_p$ corresponds to at most one secure symbol $s_{iq}$. We show this for *initial* message symbols, since message symbols in the final configuration are just a repartitioning of the symbols of message vectors in the initial configuration. This is true by the construction of $G_s^I$, which maps each secure symbol to a unique symbol of a message vector in the initial configuration. Also, this implies that since $(m_j^F)_p$ has the unique secure symbol $s_{iq}$, $(m_j^F)_p = (m_i^I)_q$. Thus, the decoding procedure correctly decodes $s_{iq}$ since

$$(m_j^F)_p - (\kappa G_\kappa)_q = (m_i^I)_q - (\kappa G_\kappa)_q = (\boldsymbol{s}_i^I G_s^I)_q = s_{iq}$$

Since the constructed code uses the same conversion procedure as the underlying access optimal convertible code $(\mathcal{C}^I, \mathcal{C}^F)$, our construction also achieves access optimal conversion. Lastly, the proof for the construction when $\lambda^I < \lambda^F$ follows a symmetric argument. □

**Remark.** The field size requirement for the construction is the same as that of the access-optimal convertible code used in [12]. More specifically, the construction utilizes an $\ell$-nested code with field size at most linear in $\min\{k^I, k^F\}$. Thus, construction 11 has the same field size requirement as the utilized access-optimal convertible codes, and benefit from recent works improving the field size requirement of access-optimal convertible codes [3]. In terms of computational overhead, in addition to the decoding procedure of the underlying convertible code, there is an additional decoding step for the MDS nested code in our secure convertible code construction.

# Chapter 5

# Fine-grained Secure Convertible Codes

## 5.1 Increasing the Secrecy Capacity Upper bound

A significant constraint on the information capacity is the lack of information on where Eve's $\ell$ eavesdropped symbols lie. For instance, consider any $\ell$-secure $[n^I, k^I; n^F, k^F]$ convertible code. In the worst case, all of Eve's $\ell$ eavesdropped symbols may lie in a single initial or final codeword. Intuitively, this is reflected in the secrecy capacity upper bound $\min\{\lambda^I(k^I - \ell), \lambda^F(k^F - \ell)\}$, which can be interpreted as each initial and final codeword simultaneously containing $\ell$ eavesdropped symbols. A natural assumption which increases the secrecy capacity upper bound in this scenario is the knowledge of the number of symbols that are eavesdropped on each initial and final codeword. We denote a secure convertible code with this added assumption as a *fine-grained secure* convertible codes. In this chapter, we give a definition for fine-grained secure convertible codes, its secrecy capacity upper bound, and discussion of scenarios benefiting from fine-grained secrecy over traditional secrecy.

Recall the notation from previous chapters. The partitions $\mathbb{S}^I, \mathbb{S}^F \in \Pi(k_S)$, where $|\mathbb{S}^I| = \lambda^I$ and $|\mathbb{S}^F| = \lambda^F$ are the secure symbol partitions that specify the mapping of secure message symbols into the initial and final codewords. Likewise, for $i \in [\lambda^I]$, random variable $\boldsymbol{\mathcal{S}}_i^I$ corresponds to the secure message symbols of $i$'th initial codeword, and for $j \in [\lambda^F]$, random variable $\boldsymbol{\mathcal{S}}_j^F$ corresponds to the secure message symbols of the $j$'th final codeword. $\boldsymbol{\mathcal{X}}_{i,\mathbb{T}}^I$ is the random variable corresponding to the symbols of the $i$'th initial codeword projected onto the indices in $\mathbb{T} \subseteq [n^I]$. $\boldsymbol{\mathcal{X}}_{i,\mathbb{O}}^F$ is the random variable defined similarly for the $j$'th final codeword projected onto indices in $\mathbb{O} \subseteq [n^F]$. Then, for each $i \in [\lambda^I]$, the vector $\boldsymbol{x}_i^I \in \mathbb{F}^{n^I}$ is the $i$'th initial codeword and for each $j \in [\lambda^F]$ the vector $\boldsymbol{x}_j^F \in \mathbb{F}^{n^F}$ is the $j$'th final codeword.

We define fine-grained secure convertible codes as convertible codes secure up to specified number of eavesdropped symbols on each initial and final codeword.

**Definition 13.** A $(\mathbb{L}^I, \mathbb{L}^F, k_S)$- Fine-Grained Secure $[n^I, k^I; n^F, k^F]$ convertible code where

$$\mathbb{L}^I = \{\ell_i^I < k^I : i \in [\lambda^I]\} \text{ and}$$
$$\mathbb{L}^F = \{\ell_j^F < k^F : j \in [\lambda^F]\}$$

is a $[n^I, k^I; n^F, k^F]$ convertible code that can store a message $\boldsymbol{s} \in \mathbb{F}^{k_S}$ satisfying following decoding, fine-grained secrecy properties:

1. **Decoding (MDS property).** For each $i \in [\lambda^I]$ and any subset $\mathbb{B} \subset [n^I]$ of size $k^I$,

$$H(\boldsymbol{\mathcal{S}}_i^I | \boldsymbol{\mathcal{X}}_{i,\mathbb{B}}^I) = 0,$$

and for each $j \in [\lambda^F]$ and any subset $\mathbb{B} \subset [n^F]$ of size $k^F$,

$$H(\boldsymbol{\mathcal{S}}_j^F | \boldsymbol{\mathcal{X}}_{j,\mathbb{B}}^F) = 0.$$

2. **Fine-grained Secrecy.** For any $\mathbb{E}_1^I, \ldots, E_{\lambda^I}^I \subset [n^I]$, $\mathbb{E}_1^F, \ldots, \mathbb{E}_{\lambda^F}^F \subset [n^F]$ such that $|\mathbb{E}_i^I| \leq \ell_i^I$ and $|\mathbb{E}_j^F| \leq \ell_J^F$ for all $i \in [\lambda^I]$ and $j \in [\lambda^F]$,

$$H\left(\boldsymbol{\mathcal{S}} \,\middle|\, \bigwedge_{i\in[\lambda^I]} \boldsymbol{\mathcal{X}}_{\mathbb{E}_i^I}^I, \bigwedge_{j\in[\lambda^F]} \boldsymbol{\mathcal{X}}_{\mathbb{E}_j^F}^F\right) = H(\boldsymbol{\mathcal{S}}).$$

For fine-grained secure convertible codes, we can derive a similar upper bound on the secrecy capacity.

**Theorem 14** (Fine-Grained secrecy capacity upper bound of Convertible Codes)**.** For positive integers $k^I, n^I, k^F, n^F$, and $\mathbb{L}^I, \mathbb{L}^F$ such that

$$\mathbb{L}^I = \{\ell_i^I < k^I : i \in [\lambda^I]\} \text{ and}$$
$$\mathbb{L}^F = \{\ell_j^F < k^F : j \in [\lambda^F]\},$$

if $\mathcal{C}$ is an $(\mathbb{L}^I, \mathbb{L}^F)-$fine-grained-secure $[n^I, k^I; n^F, k^F]$ convertible code storing secure symbols $\boldsymbol{s} \in \mathbb{F}^{k_S}$, then

$$H(\boldsymbol{\mathcal{S}}) \leq \min\left\{\sum_{i=1}^{\lambda^I}(k^I - \ell_i^I), \sum_{j=1}^{\lambda^F}(k^F - \ell_j^F)\right\}.$$

*Proof.* Suppose $k^I \leq k^F$. For each $i \in [\lambda^I]$, suppose $\mathbb{E}_i \subset \mathbb{B}_i \subset [n^I]$ such that $|\mathbb{E}_i| = \ell_i^I$ and $|\mathbb{B}_i| = k^I$. Then,

$$\begin{aligned}
H(\boldsymbol{\mathcal{S}}_i^I) &= H(\boldsymbol{\mathcal{S}}_i^I | \boldsymbol{\mathcal{X}}_{i,\mathbb{E}_i}^I) - H(\boldsymbol{\mathcal{S}}_i^I | \boldsymbol{\mathcal{X}}_{i,\mathbb{B}_i}^I) \\
&= H(\boldsymbol{\mathcal{S}}_i^I | \boldsymbol{\mathcal{X}}_{i,\mathbb{E}_i}^I) - H(\boldsymbol{\mathcal{S}}_i^I | \boldsymbol{\mathcal{X}}_{i,\mathbb{E}_i}^I, \boldsymbol{\mathcal{X}}_{i,\mathbb{B}_i\setminus\mathbb{E}_i}^I\} \\
&= I(\boldsymbol{\mathcal{S}}_i^I; \boldsymbol{\mathcal{X}}_{i,\mathbb{B}_i\setminus\mathbb{E}_i}^I | \boldsymbol{\mathcal{X}}_{i,\mathbb{E}_i}^I) \\
&\leq H(\boldsymbol{\mathcal{X}}_{i,\mathbb{B}_i\setminus\mathbb{E}_i}^I | \boldsymbol{\mathcal{X}}_{i,\mathbb{E}_i}^I) \\
&\leq H(\boldsymbol{\mathcal{X}}_{i,\mathbb{B}_i\setminus\mathbb{E}_i}^I) \\
&\leq k^I - \ell_i^I,
\end{aligned}$$

where

$$H(\boldsymbol{\mathcal{S}}) = \sum_{i=1}^{\lambda^I} H(\boldsymbol{\mathcal{S}}_i^I) \leq \sum_{i=1}^{\lambda^I}(k^I - \ell_i^I).$$

The modification for the case when $k^F < k^I$ is symmetrical, and we obtain the desired bound. □

Note that if we take $\ell = \max\left\{\sum_{i\in[\lambda^I]} \ell_i^I, \sum_{j\in[\lambda^F]} \ell_j^F\right\}$ then we equivalently have that for fine-grained convertible codes, the secrecy capacity is upper bounded by $k_S \leq k - \ell$. That is, with fine-grained security, we remove the restrictive $\lambda^I$ or $\lambda^F$ term in the original secrecy capacity upper bound.

For $(\mathbb{L}^I, \mathbb{L}^F, k_S)$-fine-grained secure convertible codes that reach the upper bound on secrecy capacity, we drop the $k_S$ from the notation, simply denoting them as optimal $(\mathbb{L}^I, \mathbb{L}^F)$-*fine-grained secure convertible codes.*

## 5.2 Scenarios in the Fine-Grained Secrecy Model

In this section, we discuss plausible scenarios that may occur when a convertible code is compromised by an eavesdropper. We observe in each scenario that the number of secure symbols on a convertible code is undesirably small or even zero under the general secrecy model. In contrast, fine-grained convertible codes will have a significantly higher number of secure symbols.

### 5.2.1 Scenario # 1: Compromised Access Set

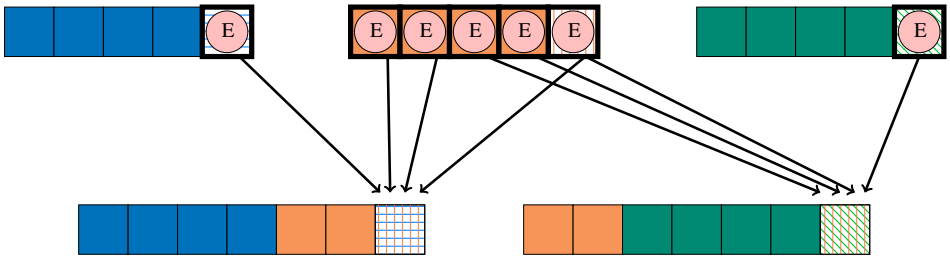


Figure 5.1: An Example of the 'Compromised Access Set' scenario on a $[5, 4; 7, 6]$ Convertible Code. All the bold-edge symbols accessed by the conversion procedure are eavesdropped symbols. The secrecy capacity in this scenario is $\min\{3(4-7), 2(6-7)\} = -9$, while the fine-grained secrecy capacity upper bound is $\min\{6, 12\} = 6$.

Recall that the general secure convertible code model introduced in Chapter 3 already captures the scenario when the codeword symbols accessed during the conversion process are compromised by an eavesdropper. By setting $\ell$ to be equal to the number of codeword symbols accessed during conversion, an $\ell$-secure convertible code is secure against the case where the specific codeword symbols accessed during conversion are eavesdropped. However, observe that the resulting number of secure symbols is at most $\min\{\lambda^I(k^I - \ell), \lambda^F(k^F - \ell)\}$ Thus, a secure convertible code handling the case when the access set is compromised by an eavesdropper has a small number of secure symbols, as pictured in Figure 5.1. In the given example, the upper bound on the secrecy capacity is less than $0$ i.e., there are no secure symbols that can possibly be stored in the convertible code.

We can improve on the number of secure symbols in the Fine-grained secrecy model. First, observe that in the previous example (Figure 5.1) the upper bound on the fine-grained secrecy

capacity is significantly greater than the general secrecy capacity. The fine-grained secrecy capacity upperbound is $6$, which allows for the possibility of having at most $6$ secure message symbols. This starkly contrasts the general secrecy capacity, which does not allow for even $1$ secure message symbol. Indeed, we can construct fine-grained secure convertible codes for the compromised access set scenario meeting the fine-grained secrecy upper bound.

The idea will be to follow a modification of the general secure convertible construction presented in Chapter $4$. For the construction when $\lambda^I \geq \lambda^F$, instead of applying a $\ell$-nested code to form each message vector in the initial configuration, where $\ell$ is the total number of eavesdropped symbols, we apply a $\ell_i^I$-nested code where $\ell_i^I$ is the number of eavesdropped symbols in the $i$'th initial codeword. Symmetrically, when $\lambda^I < \lambda^F$, we instead apply a $\ell_j^F$-nested code to form each message vector in the final configuration, where $\ell_j^F$ is the number of eavesdropped symbols in the $j$'th final codeword. We illustrate this procedure with the example below constructing the fine-grained secure convertible code pictured in Figure 5.1.

**Example: A $(\{1,5,1\},\{0,0\})$-fine-grained secure $[5,4;7,6]$ Convertible Code** Let $\boldsymbol{s} = s_1 \dots s_6 \in \mathbb{F}_7^6$ be the secure message symbols and let $\boldsymbol{\kappa} = \kappa_1 \dots \kappa_4 \in \mathbb{F}_7^4$ be the masking symbols. Consider a MDS $1$-nested $[4,4]$ code $\mathcal{D}_1^I$ and a MDS $4$-nested $[4,4]$ code $\mathcal{D}_2^I$ with respective generator matrix $G_1, G_2$ defined as

$$G_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}, G_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Then, set the message vectors in the initial configuration $\boldsymbol{m}_1^I, \boldsymbol{m}_2^I, \boldsymbol{m}_3^I \in \mathbb{F}_7^4$ as

$$\begin{aligned} \boldsymbol{m}_1^I &= \mathcal{D}_1^I\left(\begin{bmatrix} s_1 & s_2 & s_3 & \kappa_1 \end{bmatrix}\right) = \begin{bmatrix} \hat{s_1} & \hat{s_2} & \hat{s_3} & \kappa_1 \end{bmatrix}, \\ \boldsymbol{m}_2^I &= \mathcal{D}_4^I\left(\begin{bmatrix} \kappa_1 & \kappa_2 & \kappa_3 & \kappa_4 \end{bmatrix}\right) = \begin{bmatrix} \kappa_1 & \kappa_2 & \kappa_3 & \kappa_4 \end{bmatrix}, \\ \boldsymbol{m}_3^I &= \mathcal{D}_1^I\left(\begin{bmatrix} s_4 & s_5 & s_6 & \kappa_1 \end{bmatrix}\right) = \begin{bmatrix} \hat{s_4} & \hat{s_5} & \hat{s_6} & \kappa_1 \end{bmatrix}, \end{aligned}$$

where $\hat{s_i} = s_i + \kappa_1$.

Let $(\mathcal{C}^I, \mathcal{C}^F)$ be an access-optimal $[5,4;7,6]$ convertible code. The initial configuration codeword is set to

$$\boldsymbol{x}^I = \left(\mathcal{C}^I\left(\boldsymbol{m}_1^I\right), \mathcal{C}^I\left(\boldsymbol{m}_2^I\right), \mathcal{C}^I\left(\boldsymbol{m}_3^I\right)\right),$$

Using the conversion procedure of the convertible code $\left(\mathcal{C}^I, \mathcal{C}^F\right)$ on the initial codewords results in final codewords $\mathcal{C}^F\left(\boldsymbol{m}_1^F\right), \mathcal{C}^F\left(\boldsymbol{m}_2^F\right)$, where the message vectors in the final configuration $\boldsymbol{m}_1^F, \boldsymbol{m}_2^F$ are defined as

$$\begin{aligned} \boldsymbol{m}_1^F &= \begin{bmatrix} \hat{s_1} & \hat{s_2} & \hat{s_3} & \kappa_1 & \kappa_1 & \kappa_2 \end{bmatrix}, \\ \boldsymbol{m}_2^F &= \begin{bmatrix} \hat{s_4} & \hat{s_5} & \hat{s_6} & \kappa_1 & \kappa_3 & \kappa_4 \end{bmatrix}. \end{aligned}$$

The secure message symbols can be decoded from any final codeword $j = 1, 2$, and there are no eavesdropped symbols in the final configuration codeword.

### 5.2.2 Scenario # 2: General 'Worst-case Eavesdropping'

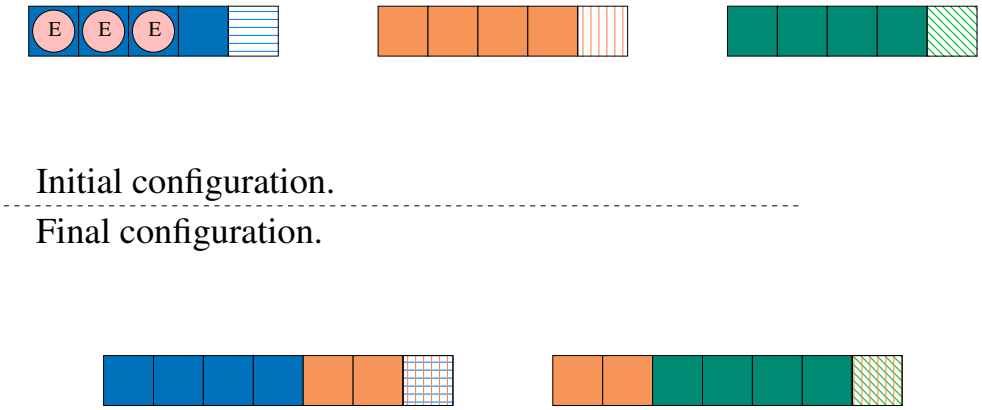


Figure 5.2: An example of a 'Worst-Case Eavesdropping' on a $[5, 4; 7, 6]$ Convertible Code. All Three Eavesdropped symbols lie in the first initial codeword. This example corresponds to a $(\{3, 0, 0\}, \{0, 0\})$-fine-grained secure convertible code.

In this scenario, all the eavesdropped symbols lie within a single initial or final codeword. The upper bound on the fine-grained secrecy capacity is significantly improved from the general secrecy capacity in this case. Illustrated in Figure 5.2, the general secrecy capacity upper bound is at most 3, while the fine-grained secrecy capacity is at most 9.

A fine-grained secure convertible code construction for this scenario follows from a similar procedure to the secure convertible code general construction. Just like in the previous scenario, the idea is that we only apply the nested code to the initial or final codeword which contain eavesdropped symbols. In the example in Figure 5.2, there are 9 secure symbols $\boldsymbol{s} \in \mathbb{F}^9$ and 3 masking symbols $\boldsymbol{\kappa} \in \mathbb{F}^3$. Construct the message vectors of the initial configuration such that the message vector for the first initial codeword contains the 3 masking symbols $\boldsymbol{\kappa}$ (the rest of the secure symbols are placed arbitrarily). Then, we apply a 3-nested $[4, 4]$ code only to the first message vector, and apply the convertible code as described previously.

### 5.2.3 Scenario # 3: 'Long-term' Eavesdropped Symbols

This scenario captures eavesdropped codeword symbols in the initial configuration codeword that persist through the conversion process. That is, the corresponding final codeword symbol after conversion will also be an eavesdropped symbol. We can model these long-term eavesdropped symbols using Fine-grained secure convertible codes. Figure 5.3 presents an example of when long-term eavesdropped symbols 'merge' i.e., when $\lambda^I \geq \lambda^F$. In contrast, Figure 5.4 models when long-term eavesdropped symbols 'split' i.e., when $\lambda^I < \lambda^F$.

We leave the construction of optimal fine-grained secure convertible codes as future work. We note that while we can use the general secure convertible code construction for these cases, the number of secure symbols is significantly much lower than the fine-grained secrecy capacity upper bound. For instance, in both of the examples presented in Figures 5.3 and 5.4, the general

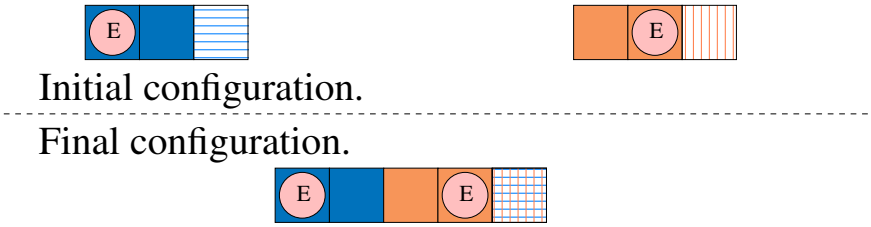


Figure 5.3: An example of 'Long-term' Eavesdropped Symbols within a $[3, 2; 5, 4]$ Convertible Code. The eavesdropped initial codeword symbols persist through the conversion procedure where their corresponding final codeword symbols are also eavesdropped symbols. This example corresponds to a $(\{1, 1\}, \{1\})$-fine-grained secure convertible code.

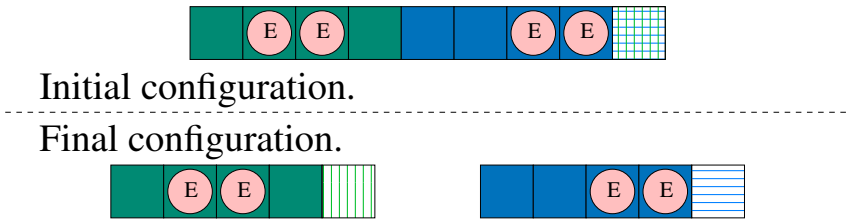


Figure 5.4: An example of 'Long-term' Eavesdropped Symbols within a $[9, 8; 5, 4]$ Convertible Code. This represents a 'split' case, where long-term eavesdropped symbols start on the same initial codeword and end on different final codewords. This example corresponds to a $(\{4\}, \{2, 2\})$-fine-grained secure convertible code.

construction cannot store any secure symbols, while the fine-grained secrecy capacity upper bounds are $4$ and $8$ respectively.

# Chapter 6

# Conclusion & Future Work

In this thesis, we introduce an information-theoretic secrecy model for convertible codes in the presence of eavesdroppers. We derived fundamental upper bounds on the number of message symbols that can be stored securely using convertible codes that provide security against eavesdroppers while maintaining access cost optimality of code conversions. We also presented explicit construction of optimal secure convertible codes meeting these bounds. Additionally, we discussed the fine-grained secrecy assumption where we additionally know the number of the eavesdropped symbols in each initial and final codeword. We derived stronger bounds beyond the original secrecy capacity and explored scenarios that would benefit from the additional assumption of fine-grained secrecy.

This work opens up several avenues for future work.

1. **Bandwidth-Optimal Convertible Codes.** The focus of this thesis was convertible codes in an access-cost setting. A natural extension is to consider the bandwidth-setting, which models and optimizes the bandwidth cost of conversion within distributed storage systems by allowing partial symbol downloads. In previous work, bandwidth-optimal convertible codes have been constructed [10, 11] by augmenting existing access-optimal convertible codes. While this same procedure can be done on our secure access-optimal convertible codes to convert it into a bandwidth-optimal convertible code, the resulting code may not necessarily preserve secrecy nor reach the secrecy capacity upperbound. As in the regenerating code setting [15, 17], the secrecy capacity may be lower due to the eavesdropper eavesdropping on the set of symbols downloaded for node recovery. To understand the secrecy capacity of convertible codes when eavesdroppers are allowed to access partial symbols, we will need to augment the bandwidth conversion model to include eavesdroppers. Thus, expanding secure convertible codes into the bandwidth setting is an interesting future direction.
2. **Fine-grained General Codes.** In the previous chapter, the secrecy capacity and specific constructions of fine-grained-secure convertible codes were explored. Constructions for general parameters remain a non-trivial challenge along with other unexplored scenario constructions.
3. **Active Adversaries.** In this thesis, we considered convertible codes that were secure against a *passive* eavesdropper that can read any $\ell$ convertible code symbols. We can

also consider convertible codes which are secure against *active* adversaries. Active adversaries can disrupt the conversion process by maliciously corrupting symbols. Active adversaries were previously studied in in the context of codes for storage systems under the setting of regenerating codes [15, 17], where constructions needed to accommodate for possibly corrupted symbols when undergoing symbol repair. The adversary's additional power is also reflected in the upperbound on the number of message symbols that can be securely stored, which is reduced as compared to the upper bound of the secrecy capacity against a passive eavesdropper. Understanding convertible codes that are secure against active eavesdroppers is a natural next step.

4. **Cryptographic Assumptions.** There is a broad, exciting line of work incorporating cryptography in traditional coding theory problems to provide constructions beyond worst-case parameter bounds [1, 6, 13, 18]. The adversary causing corruptions (e.g. error, erasure, or edit errors) in codewords is assumed to be computationally-bounded, allowing the use of cryptographic primitives that ultimately reduce the rate while improving the corruption tolerance of the code. Relaxing worst-case guarantees within distributed storage coding problems may lead to new breakthrough solutions, as they have in other coding theory problems. Specifically, for the conversion problem, we conjecture that if an eavesdropper reading any $\ell$ nodes is bounded to polynomial-time computations, one can construct $\ell$-secure convertible codes storing a number of secure symbols beyond the information-theoretic secrecy capacity upperbound derived in this thesis.

# Bibliography


[1] Mohammad Hassan Ameri, Alexander R Block, and Jeremiah Blocki. Memory-hard puzzles in the standard model with applications to memory-hard functions and resource-bounded locally decodable codes. In *International Conference on Security and Cryptography for Networks*, pages 45–68. Springer, 2022. 4

[2] Ning Cai and Raymond W. Yeung. Secure network coding on a wiretap network. *IEEE Transactions on Information Theory*, 57(1):424–435, 2011. doi: 10.1109/TIT.2010.2090197. 1

[3] Saransh Chopra, Francisco Maturana, and K. V. Rashmi. On low field size constructions of access-optimal convertible codes. In *2024 IEEE International Symposium on Information Theory (ISIT)*, pages 1456–1461, 2024. doi: 10.1109/ISIT57864.2024.10619440. 4.2.1

[4] Toni Ernvall, Salim El Rouayheb, Camilla Hollanti, and H. Vincent Poor. Capacity and security of heterogeneous distributed storage systems. *IEEE Journal on Selected Areas in Communications*, 31(12):2701–2709, 2013. doi: 10.1109/JSAC.2013.131210. 1

[5] Sanjay Ghemawat, Howard Gobioff, and Shun-Tak Leung. The google file system. *SIGOPS Oper. Syst. Rev.*, 37(5):29–43, oct 2003. ISSN 0163-5980. doi: 10.1145/1165389.945450. URL `https://doi.org/10.1145/1165389.945450`. 1

[6] Brett Hemenway and Rafail Ostrovsky. Public-key locally-decodable codes. In *Annual International Cryptology Conference*, pages 126–143. Springer, 2008. 4

[7] Cheng Huang, Huseyin Simitci, Yikang Xu, Aaron Ogus, Brad Calder, Parikshit Gopalan, Jin Li, and Sergey Yekhanin. Erasure coding in windows azure storage. In *2012 USENIX Annual Technical Conference (USENIX ATC 12)*, pages 15–26, Boston, MA, jun 2012. USENIX Association. ISBN 978-931971-93-5. URL `https://www.usenix.org/conference/atc12/technical-sessions/presentation/huang`. 1

[8] Saurabh Kadekodi, K. V. Rashmi, and Gregory R. Ganger. Cluster storage systems gotta have heart: improving storage efficiency by exploiting disk-reliability heterogeneity. In *Proceedings of the 17th USENIX Conference on File and Storage Technologies*, FAST'19, page 345–358, USA, 2019. USENIX Association. ISBN 9781931971485. 1

[9] Francisco Maturana and K. V. Rashmi. Convertible codes: Enabling efficient conversion of coded data in distributed storage. *IEEE Transactions on Information Theory*, 68(7): 4392–4407, 2022. doi: 10.1109/TIT.2022.3155972. 1, 2.3, 1, 2.3

[10] Francisco Maturana and K. V. Rashmi. Bandwidth cost of code conversions in the split regime. In *2022 IEEE International Symposium on Information Theory (ISIT)*, pages 3262–

3267, 2022. doi: 10.1109/ISIT50566.2022.9834604. 1, 2.3, 1

[11] Francisco Maturana and K. V. Rashmi. Bandwidth cost of code conversions in distributed storage: Fundamental limits and optimal constructions. *IEEE Transactions on Information Theory*, 69(8):4993–5008, 2023. doi: 10.1109/TIT.2023.3265512. 1, 2.3, 1

[12] Francisco Maturana, V. S. Chaitanya Mukka, and K. V. Rashmi. Access-optimal linear mds convertible codes for all parameters. In *2020 IEEE International Symposium on Information Theory (ISIT)*, pages 577–582, 2020. doi: 10.1109/ISIT44484.2020.9173947. 1, 2.3, 11, 4.2.1

[13] Rafail Ostrovsky, Omkant Pandey, and Amit Sahai. Private locally decodable codes. In *International Colloquium on Automata, Languages, and Programming*, pages 387–398. Springer, 2007. 4

[14] L. H. Ozarow and A. D. Wyner. Wire-tap channel ii. *AT&T Bell Laboratories Technical Journal*, 63(10):2135–2157, 1984. doi: 10.1002/j.1538-7305.1984.tb00072.x. 2.4, 2.4.1, 2, 3

[15] Sameer Pawar, Salim El Rouayheb, and Kannan Ramchandran. Securing dynamic distributed storage systems against eavesdropping and adversarial attacks. *IEEE Transactions on Information Theory*, 57(10):6734–6753, 2011. 1, 1, 3

[16] K. V. Rashmi, Nihar B. Shah, and Kannan Ramchandran. A piggybacking design framework for read-and download-efficient distributed storage codes. *IEEE Transactions on Information Theory*, 63(9):5802–5820, 2017. doi: 10.1109/TIT.2017.2715043. 2.3

[17] KV Rashmi, Nihar B Shah, Kannan Ramchandran, and P Vijay Kumar. Information-theoretically secure erasure codes for distributed storage. *IEEE Transactions on Information Theory*, 64(3):1621–1646, 2017. 1, 1, 3

[18] Ronen Shaltiel and Jad Silbak. Explicit list-decodable codes with optimal rate for computationally bounded channels. *Comput. Complex.*, 30(1), June 2021. ISSN 1016-3328. doi: 10.1007/s00037-020-00203-w. URL `https://doi.org/10.1007/s00037-020-00203-w`. 4

[19] Arunkumar Subramanian and Steven W McLaughlin. MDS codes on the erasure-erasure wiretap channel. *arXiv preprint arXiv:0902.3286*, 2009. 1, 2.4, 2.4.1, 3, 4, 5, 6, 7

[20] A. D. Wyner. The wire-tap channel. *The Bell System Technical Journal*, 54(8):1355–1387, 1975. doi: 10.1002/j.1538-7305.1975.tb02040.x. 2.4