Secure RAID Schemes for Distributed Storage

Wentao Huang and **Jehoshua Bruck**
California Institute of Technology, Pasadena, USA
{whuang,bruck}@caltech.edu

***Abstract*—We propose secure RAID, i.e., low-complexity schemes to store information in a distributed manner that is resilient to node failures and resistant to node eavesdropping. We generalize the concept of systematic encoding to secure RAID and show that systematic schemes have significant advantages in the efficiencies of encoding, decoding and random access. For the practical high rate regime, we construct three XOR-based systematic secure RAID schemes with optimal encoding and decoding complexities, from the EVENODD codes and B codes, which are array codes widely used in the RAID architecture. These schemes optimally tolerate two node failures and two eavesdropping nodes. For more general parameters, we construct efficient systematic secure RAID schemes from Reed-Solomon codes. Our results suggest that building "keyless", information-theoretic security into the RAID architecture is practical.**

I. Introduction

In the RAID architecture [3], information is stored distributively among multiple nodes, such as an array of disks or a cluster of networked computers, in a redundant manner that is resilient to individual node failures. RAID improves the reliability, availability and performance of the system and has seen extensive applications over the past decades [3].

As distributed storage systems are increasingly being used to store critical and sensitive data, the challenge of protecting data privacy is imminent. This paper studies the design of distributed storage systems that are not only failure-resilient, but also resistant to adversarial eavesdropping of individual nodes. Specifically, we study the problem of storing a message among n nodes such that any $n-r$ nodes can decode the message but any coalition of z nodes cannot infer any information about the message. This problem was initially studied in the literature under the context of secret sharing, and space-optimal schemes are known such as Shamir's scheme [13] and its generalization [1]. More recently, there has been considerable interest in improving the efficiency of the schemes during the repair process (i.e., recovering from node failures), in terms of communication bandwidth and locality, e.g., [10], [12], [11].

However, application of existing schemes to practical distributed storage systems has been limited by their complexities [7]. Specifically, current schemes have much higher encoding and decoding complexities than the erasure codes employed in practice, that offer protection against failure but not against eavesdropping. The reasons are twofold: Firstly, erasure codes for distributed storage are typically encoded systematically so that the information symbols appear "in the clear" in the codeword. This trivializes decoding when no erasure occurs and significantly simplifies encoding. In comparison, an eavesdropping-resistant secure scheme does not allow information symbols to appear in the clear and thus the encoding/decoding complexity is increased. Secondly, while there have been extensive studies on optimizing the encoding/decoding complexity of erasure codes and numerous good constructions are known [2], [15], very little is known about how to design secure schemes with similar optimality.

This paper proposes low-complexity schemes, termed *secure RAID schemes*, that are resilient to node failures and resistant to node eavesdropping. To the best of our knowledge, they are the first schemes that have comparable computational and implementation complexities as practical erasure codes such as the EVENODD [2] and Reed-Solomon (RS) codes [8]. They are also the first schemes that are shown to have optimal encoding/decoding complexities.

We highlight two ideas from our constructions: Firstly, we generalize the concept of systematic encoding to the secure setting. Refer to Fig. 1 for an example of a systematic scheme, which can optimally tolerate two node erasures and two eavesdropping nodes. Secondly, we leverage the results on efficient erasure codes, notably on array codes, and construct secure schemes from them and their dual codes. Specifically, the codeword of an array code is a $t \times n$ array; each node stores a column of the array so erasure and distance are defined column-wise. Well-known MDS array codes suitable for RAID include the EVENODD [2] and B [15] codes. Both of them are XOR-based high-rate codes that can tolerate 2 erasures with optimal computation in the sense that their generator matrices are "low-density" (sparse), and so encoding requires an optimal or almost optimal number of XOR operations.

We make several contributions in designing array-based secure RAID schemes. We study the density of the generator matrix of secure RAID schemes, which characterizes the number of operations required by encoding/decoding, and prove a lower bound. We construct three families of optimal secure RAID schemes based on the B and EVENODD codes and their dual codes. Refer to Fig. 1 for an example. The schemes are XOR-based, space-optimal, and have low or lowest density generator matrices. Specifically, the schemes can correct $r \leq 2$ node erasures and resist $z \leq 2$ eavesdropping nodes. In these schemes, encoding one bit of information on average requires approximately $r + z = 4$ XORs and decoding one bit of information when no erasure occurs on average requires approximately $z = 2$ XORs. We show that these encoding and decoding complexities are optimal or almost optimal. Finally, for general parameters n, r and z, we present a systematic, space-optimal scheme based on RS codes, and show that its computational complexity is significantly better than Shamir's scheme and its generalization [13], [1]. The latter two schemes are also related to RS codes [9] but are not systematic.

Our results suggest that building "keyless", information-theoretic security into the RAID architecture is practical. Particularly, for Reed-Solomon, EVENODD or B coded distributed storage systems, extending them to employ the proposed secure RAID schemes requires only minor modification to the implementation, with small computational and therefore performance overhead. Due to space limitation we omit the proofs in this paper and defer them to the longer version [4].

Node 1	Node 2	Node 3	Node 4	Node 5	Node 6
u_1	u_2	u_3	u_4	u_5	u_6
$u_3 \oplus u_5 \oplus m_1$	$u_6 \oplus u_3 \oplus m_2$	$u_2 \oplus u_1 \oplus m_3$	$u_5 \oplus u_6 \oplus m_4$	$u_1 \oplus u_4 \oplus m_5$	$u_4 \oplus u_2 \oplus m_6$
$u_2 \oplus u_6 \oplus m_3 \oplus m_5$	$u_4 \oplus u_5 \oplus m_6 \oplus m_3$	$u_6 \oplus u_4 \oplus m_2 \oplus m_1$	$u_1 \oplus u_3 \oplus m_5 \oplus m_6$	$u_3 \oplus u_2 \oplus m_1 \oplus m_4$	$u_5 \oplus u_1 \oplus m_4 \oplus m_2$

Fig. 1: A secure RAID scheme constructed from the B codes [15]. Symbols are bits and operations are XORs. $m_1, ..., m_6$ are message bits and $u_1, ..., u_6$ are random key bits. The scheme is able to correct two node erasures and is secure against two eavesdropping nodes. The scheme is optimal in several senses. It has optimal rate and optimal field size. It follows a generalized *systematic* form: all keys are stored uncoded in the first row; all message bits are stored uncoded in the second row, each padded (XORed) by an optimal number of two keys necessary to defeat two eavesdropping nodes; and the third row is redundant. The systematic form implies optimal decoding complexity as the message bits can be decoded by canceling the least amount of keys. The scheme is also optimal in terms of encoding complexity: every key and message bit is checked by an optimal number of two parities in the redundant (third) row necessary to correct two erasures. The scheme is constructed in Section III-D. Two more families of almost optimal schemes, which require slightly more computation but are more flexible in length, are constructed in Section III-A and III-C. The generalized form of systematic encoding is defined in Section III-B.

II. Definitions and Converse

Let $\mathcal{Q}$ be a generic alphabet and let $[n] = \{1, ..., n\}$. For an index set $I \subset [n]$ and a vector $\boldsymbol{c} = (c_1, ..., c_n)$, let $\boldsymbol{c}_I = (c_i)_{i \in I}$. An $(n, k, r, z)_{\mathcal{Q}}$ secure RAID scheme is a randomized encoding function F that maps a (secret) message $\boldsymbol{m} \in \mathcal{Q}^k$ and a uniformly distributed vector $\boldsymbol{u} \in \mathcal{Q}^v$, also referred to as *keys*, to the codeword $\boldsymbol{c} = F(\boldsymbol{m}, \boldsymbol{u}) \in \mathcal{Q}^n$, such that:

1) (Reliability) $\forall I \subset [n], |I| \geq n - r : \ H(\boldsymbol{m}|\boldsymbol{c}_I) = 0$, implying a decoding function D_I such that $D_I(\boldsymbol{c}_I) = \boldsymbol{m}$.
2) (Secrecy) $\forall I \subset [n], |I| \leq z : \ I(\boldsymbol{m}; \boldsymbol{c}_I) = 0$.

Such schemes are referred to as the threshold secret sharing schemes [13], [1] in the literature. In this paper we focus on designing low-complexity schemes suitable for distributed storage, notably for the RAID architectures, and name them *secure RAID schemes*. We focus on two kinds of linear schemes, namely *scalar* schemes and *array* schemes. For a scalar scheme, $\mathcal{Q}$ is a finite field $\mathbb{F}_q$ and the encoding function F is linear over $\mathbb{F}_q$. For an array scheme, $\mathcal{Q}$ is a vector space $\mathbb{F}_q^t$ and $\boldsymbol{m}$, $\boldsymbol{u}$ can be interpreted as vectors over $\mathbb{F}_q$ of length tk and tv. When this interpretation is made, $\boldsymbol{m}$ and $\boldsymbol{u}$ are denoted by $\bar{\boldsymbol{m}}$ and $\bar{\boldsymbol{u}}$. The encoding function is linear over $\mathbb{F}_q$, taking $\bar{\boldsymbol{m}}$ and $\bar{\boldsymbol{u}}$ as inputs. The output is a $t \times n$ array with entries $c_{i,j}$ over $\mathbb{F}_q$. A column of the array correspond to an entry of $\boldsymbol{c}$ over $\mathbb{F}_q^t$, and note that under the array representation erasure and eavesdropping are column-wise. Alternatively, the codeword is denoted by $\bar{\boldsymbol{c}}$ when regarded as a vector over $\mathbb{F}_q$ of length tn, i.e., $\bar{\boldsymbol{c}} = (c_{1,1}, ..., c_{t,1}, ..., c_{1,n}, ..., c_{t,n})$. Scalar schemes are special cases of array schemes with $t = 1$, and without loss of generality, we deal with array schemes. An $[n, k]_{\mathbb{F}_q^t}$ *array code* $\mathcal{C}$ of minimum distance $d_{\min}(\mathcal{C}) = r + 1$ is equivalent to an $(n, k, r, 0)_{\mathbb{F}_q^t}$ secure RAID scheme.

The *rate* of an (n, k, r, z) secure RAID scheme is k/n. The optimal rate is $\frac{n-r-z}{n}$ and the maximum message size is $k = n - r - z$ [5]. A scheme is associated with an encoding algorithm and multiple decoding algorithms. The encoding algorithm is used to evaluate the encoding function F. The decoding algorithms are used to evaluate the decoding functions D_I for $|I| \geq n - r$, referred to as the *systematic decoding algorithm* when $|I| = n$ and the *erasure decoding algorithm* when $|I| < n$. Define the *generator matrix* of a secure RAID scheme to be the $(v+k)t \times nt$ matrix G over $\mathbb{F}_q$ such that $(\bar{\boldsymbol{u}}, \bar{\boldsymbol{m}})G = \bar{\boldsymbol{c}}$. We refer to the first vt rows of G as the *key rows*, and the remaining kt rows as the *message rows*. Define the *density* of a vector or matrix to be the number of non-zero entries. We are interested in designing secure RAID schemes with low-density generator matrices. Such schemes require a small number of operations in encoding/decoding. We first prove several useful converse results. The following one addresses the amount of required randomness.

Theorem 1. *Any linear rate-optimal $(n, k, r, z)_{\mathbb{F}_q^t}$ secure RAID scheme uses at least zt keys, and is equivalent to a scheme that uses exactly zt keys over $\mathbb{F}_q$.*

Therefore a rate-optimal $(n, k, r, z)_{\mathbb{F}_q^t}$ scheme uses exactly zt keys, and its generator matrix G has size $(z + k)t \times nt$.

Theorem 2. *Consider the generator matrix of a rate-optimal (n, k, r, z) scheme, the density of a key row is at least $n-z+1$, and the density of a message row is at least $r + 1$.*

From Theorem 2 we obtain a lower bound on the encoding complexity of a XOR-based scheme, i.e., schemes with $q = 2$.

Corollary 1. *Encoding a rate-optimal $(n, k, r, z)_{\mathbb{F}_2^t}$ scheme requires at least $r + z + \frac{rz-z}{n-r-z}$ XORs per message bit.*

III. Construction of Secure RAID Schemes

A. Secure EVENODD

We design a family of low-complexity XOR-based schemes from the EVENODD codes. Let p be a prime, the EVENODD code is a $[p + 2, p]$ MDS array code over $\mathbb{F}_2^{p-1}$ with a low density generator matrix [2]. Refer to Fig. 2 for an example of $p = 5$. We describe our construction idea using this example. Denote the code in Fig. 2 by $\mathcal{C}_2$, which corrects 2 column erasures. To build secrecy into $\mathcal{C}_2$, consider its dual $\mathcal{C}_2^\perp$, obtained by switching the roles of information and parity bit, i.e., in Fig. 2 an information bit $c_{i,6}$ is checked by (parity) entries labeled by i in the top plot, and $c_{i,7}$ is checked by entries labeled by i and S in the bottom. Since $\mathcal{C}_2$ is MDS, so is $\mathcal{C}_2^\perp$. $\mathcal{C}_2^\perp$ is a $[p+2, 2]$ code for secrecy in the sense that if we encode two columns of keys as information bits by $\mathcal{C}_2^\perp$ and pad (XOR) this key array to a message array, then any two columns in the sum array reveal no information about the message. Now we have two efficient codes for reliability and secrecy, respectively. The challenge is to integrate them into a single scheme that is both reliable and secure. The straightforward approach for combining the two codes typically fails. We prove in the sequel that a sufficient condition for the correctness of the scheme is that the code for secrecy, denoted by $\mathcal{C}_1$, is a subcode of $\mathcal{C}_2$ (the code for reliability). In our example, $\mathcal{C}_2^\perp$ is not a subcode of $\mathcal{C}_2$. However, switch column 1 and 6 in $\mathcal{C}_2^\perp$ to obtain $\mathcal{C}_1$ (encoding described in Fig. 3), then we can show that $\mathcal{C}_1$ meets the subcode property. Based on $\mathcal{C}_1$ and $\mathcal{C}_2$ we construct a secure RAID scheme as follows. Generate

1	1	1	1	1	$c_{1,6}$	
2	2	2	2	2	$c_{2,6}$	
3	3	3	3	3	$c_{3,6}$	
4	4	4	4	4	$c_{4,6}$	

1	2	3	4	S		$c_{1,7}$
2	3	4	S	1		$c_{2,7}$
3	4	S	1	2		$c_{3,7}$
4	S	1	2	3		$c_{4,7}$

Fig. 2: $[7, 5]$ EVENODD code. Codeword is a 4×7 array. The first 5 columns store information bits. Parity bit $c_{i,6}$ is the XOR of all entries labeled by i in the top plot. Parity bit $c_{i,7}$ is the XOR of all entries labeled by i and all entries labeled by S in the bottom plot.

1	1	1	1	1	1	
2	2	2	2	2	2	
3	3	3	3	3	3	
4	4	4	4	4	4	

	2	3	4	Σ	1	1
	3	4	Σ	1	2	2
	4	Σ	1	2	3	3
	Σ	1	2	3	4	4

Fig. 3: Encoding of keys in the (7,3,2,2) secure EVENODD, which is exactly the encoding of $\mathcal{C}_1$. $i = 1, ..., 4$ in the top (or bottom) array represents that a key bit $u_{i,1}$ (or $u_{i,2}$) is added to the corresponding entry in the codeword array; and Σ represents that $\bigoplus_{i=1}^{4} u_{i,2}$ is added. Note that the padding pattern is almost optimal, in the sense that most entries are padded by only two keys and that when more than two keys are padded, Σ only needs to be computed once.

two columns of random keys; encode the keys by $\mathcal{C}_1$ but skip the last two columns of the codeword; XOR three columns of message bits with the 3-rd to 5-th columns of the key array; finally complete the last two columns by encoding $\mathcal{C}_2$. Note that the first 2 columns store only keys, the next 3 columns store uncoded message bits padded by keys, and the last two columns are redundant. The encoding of keys is shown in Fig. 3. The scheme corrects 2 erasures, and because $\mathcal{C}_1 \subset \mathcal{C}_2$, the encoding of keys in the last 2 columns is consistent with $\mathcal{C}_1$, implying secrecy against 2 eavesdropping nodes. Hence we have the $(7, 3, 2, 2)$ secure EVENODD scheme.

The construction technique can be readily generalized to any prime p. For an integer a, denote by $\langle a \rangle$ the unique integer m, $0 \leq m < p$, such that $a \equiv m \pmod{p}$.

Construction 1. (EVENODD Code [2]) *Let p be a prime, and $m_{i,j}$, $i \in [p-1]$, $j \in [p]$ be the message bits. The codeword of EVENODD forms a $(p-1) \times (p+2)$ array. The first p columns of the array are the systematic symbols, i.e., for $i \in [p-1]$, $j \in [p]$, $c_{i,j} = m_{i,j}$. The last two columns are redundant symbols, i.e., for $i \in [p-1]$, $c_{i,p+1} = \bigoplus_{l=1}^{p} m_{i,l}$ and $c_{i,p+2} = S + \left(\bigoplus_{l=1}^{p} m_{\langle i+1-l \rangle, l}\right)$, where $S = \bigoplus_{l=2}^{p} m_{\langle 1-l \rangle, l}$, and for the ease of notation we define $m_{0,j} = 0$.*

Construction 2. (Secure EVENODD) *Let p be a prime. For $i \in [p-1]$, $j \in [p-2]$ and $l = 1, 2$, let $m_{i,j}$ be the message bits, and let $u_{i,l}$ be the uniformly distributed key bits. The codeword of secure EVENODD forms a $(p-1) \times (p+2)$ array. The first two columns of the array store the key bits, i.e., $c_{i,1} = u_{i,1}$ for $i \in [p-1]$, and denote $u_{\Sigma,2} = \bigoplus_{l=1}^{p-1} u_{l,2}$,*

$$c_{i,2} = \begin{cases} u_{i,1} \oplus u_{i+1,2} & i = 1, ..., p-2 \\ u_{i,1} \oplus u_{\Sigma,2} & i = p-1 \end{cases}$$

An entry in the 3-rd to p-th columns stores a message bit padded by two or occasionally more keys, i.e., for $j = 3, ..., p$,

$$c_{i,j} = \begin{cases} u_{i,1} \oplus u_{\langle i+j-1 \rangle, 2} \oplus m_{i,j-2} & i + j \neq p+1 \\ u_{i,1} \oplus u_{\Sigma,2} \oplus m_{i,j-2} & i + j = p+1 \end{cases}$$

The last two columns are redundant symbols computed by encoding the EVENODD code, regarding the first p columns of the array as information symbols.

Theorem 3. *For any prime p, secure EVENODD is a $(p + 2, p - 2, 2, 2)_{\mathbb{F}_2^{p-1}}$ secure RAID scheme.*

Systematic decoding the scheme is straightforward by first decoding the keys from the first two columns and then canceling them from the 3-rd to p-th columns. In case of erasures/error, the erasure/error decoding algorithm of EVENODD [2] is invoked, followed by systematic decoding. Encoding the scheme requires on average $4 + \frac{3}{p-2} + \frac{2}{p-1}$ XORs per message bit. Systematic decoding the scheme requires on average $2 + \frac{1}{p-2} + \frac{1}{p-1}$ XORs per message bit. Note that encoding each message bit requires at least $4 + \frac{2}{p-2}$ XORs by Corollary 1. Moreover, to be secure against $z = 2$ eavesdroppers, each message bit has to be padded by at least two keys, and different message bits must not be padded by the same pair of keys, so decoding each message bit requires at least 2 XORs. Hence secure EVENODD has almost optimal encoding and systematic decoding complexities.

B. Systematic Schemes and Construction Method

Codes for distributed storage are typically systematic, i.e., message symbols appear uncoded in the codeword. Systematic codes have important advantages in the efficiencies of encoding, decoding and random access. For secure RAID schemes, conventional systematic encoding is forbidden by the secrecy requirement. However, Fig. 1 and Construction 2 suggest that a generalized form of systematic encoding offers similar efficiency as systematic codes. Formally, we say that a secure RAID scheme is *systematic* if (1) the keys $\bar{u}_1, ..., \bar{u}_{tv}$ are stored in the uncoded form in the codeword $\bar{\boldsymbol{c}}$, and (2) the message symbols $\bar{m}_1, ..., \bar{m}_{tk}$ are stored in the uncoded form in the codeword, each padded by a function of z keys, which is the minimal number of keys necessary to resist z eavesdropping nodes. These $tv+tk$ symbols are called the *systematic symbols*. Refer to Fig. 1 for an example of systematic schemes. Secure EVENODD is also essentially systematic except that a few message bits are padded by more than z keys.

We introduce a method to design systematic secure RAID schemes. The method falls under the general framework of coset coding, which dates back to [14] on the wiretap channel. However here we put special emphasis on designing efficient and systematic schemes in the context of secure RAID. Consider an $[n, k_1]_{\mathbb{F}_q^t}$ code $\mathcal{C}_1$ and an $[n, k_2]_{\mathbb{F}_q^t}$ code $\mathcal{C}_2$, such that codewords of $\mathcal{C}_1$ are codewords of $\mathcal{C}_2$, i.e., $\mathcal{C}_1 \subset \mathcal{C}_2$. Encode $\mathcal{C}_2$ systematically and denote the index set of the systematic

symbols in the codeword by I_2. Encode $\mathcal{C}_1$ systematically such that the index set I_1 of its systematic symbols satisfies $I_1 \subset I_2$ (which is possible as $\mathcal{C}_1 \subset \mathcal{C}_2$). Alternatively, we can encode $\mathcal{C}_1$ in more flexible ways as long as there is a set of entries I_1 in the codeword such that $I_1 \subset I_2$ and that $\mathcal{C}_1$ can be decoded from the entries in I_1. The secure RAID scheme has 2 steps.

Step 1: Draw $|I_1| = tk_1$ random keys from $\mathbb{F}_q$. Encode $\mathcal{C}_1$ by regarding the keys as information symbols. Puncture (delete) all entries in the codeword that is not in I_2. Denote the punctured codeword by $\boldsymbol{d}$. Note that $|\boldsymbol{d}| = |I_2| = tk_2$.

Step 2: Let $\bar{\boldsymbol{m}}$ be the message vector of length $t(k_2 - k_1)$ over $\mathbb{F}_q$. Note that $|\bar{\boldsymbol{m}}| = |I_2 \backslash I_1|$. Augment $\bar{\boldsymbol{m}}$ to have the same length as $\boldsymbol{d}$ by inserting 0 in entries indexed by I_1. Sum $\boldsymbol{d}$ and the augmented $\bar{\boldsymbol{m}}$ to obtain $\boldsymbol{e}$. Encode $\mathcal{C}_2$ by regarding $\boldsymbol{e}$ as information symbols to obtain $\bar{\boldsymbol{c}}$. $\bar{\boldsymbol{c}}$ is a length-tn vector over $\mathbb{F}_q$, and is the output codeword of the secure RAID scheme.

Theorem 4. *Let $\mathcal{C}_1$ be an $[n, k_1]$ code and $\mathcal{C}_2$ be an $[n, k_2]$ code, both over $\mathbb{F}_q^t$, such that $\mathcal{C}_1 \subset \mathcal{C}_2$. Then the described encoding scheme is an $(n, k_2 - k_1, r, z)$ secure RAID scheme over $\mathbb{F}_q^t$, where $r = d_{\min}(\mathcal{C}_2) - 1$, $z = d_{\min}(\mathcal{C}_1^{\perp}) - 1$.*

An important special case is that $\mathcal{C}_1$ and $\mathcal{C}_2$ are MDS.

Corollary 2. *If $\mathcal{C}_1$ and $\mathcal{C}_2$ are MDS, then the described scheme is an $(n, k = k_2 - k_1, r = n - k_2, z = k_1)$ secure RAID scheme. Particularly, the scheme has optimal rate.*

The construction method results in schemes that are almost systematic (except that a message symbol may be padded by more than z keys), where I_1 is the systematic key symbols, and $I_2 \backslash I_1$ is systematic message symbols. This systematic form connects the computational complexity of the scheme to that of the codes. The encoding complexity of the scheme is essentially the complexity of encoding $\mathcal{C}_1$ and $\mathcal{C}_2$. A simple systematic decoding algorithm for the scheme is to compute $\boldsymbol{d}$ by encoding $\mathcal{C}_1$ and then cancel it from $\boldsymbol{e}$ to obtain $\bar{\boldsymbol{m}}$, hence the complexity is dominated by encoding $\mathcal{C}_1$. The erasure decoding algorithm first corrects the erasures by invoking the erasure correction algorithm of $\mathcal{C}_2$, and then invokes the systematic decoding algorithm. So the complexity is essentially the complexity of (erasure) decoding $\mathcal{C}_2$ plus encoding $\mathcal{C}_1$. In words, to construct efficient secure RAID schemes, it suffices to find a pair of MDS codes $\mathcal{C}_1, \mathcal{C}_2$ of appropriate rates such that $\mathcal{C}_1 \subset \mathcal{C}_2$, and that $\mathcal{C}_1$ can be efficiently encoded, and that $\mathcal{C}_2$ can be efficiently encoded and decoded.

A key idea in our constructions is to design $\mathcal{C}_2$ based on MDS array codes and design $\mathcal{C}_1$ based on their dual codes. This is because the array codes and their duals 1) are both MDS, so that the resulting scheme is rate-optimal; 2) have high and low rate, respectively, so that the scheme has high rate; 3) both have low or lowest density generator matrices, implying optimal or almost optimal encoding complexity, so that the scheme is efficient. However, array codes and their duals are rarely known to contain each other. Surprisingly, such as in the EVENODD case, we can often modify the codes appropriately to meet the containment condition, while not compromising their complexity and distance. We construct two more families of optimal schemes in the sequel with this idea.

Finally, the construction method is also promising in *implementation complexity*. The encoder of the secure RAID scheme consists of the encoders of $\mathcal{C}_1$ and $\mathcal{C}_2$. The decoder consists of the encoder of $\mathcal{C}_1$ (for systematic decoding) and the decoder of $\mathcal{C}_2$ (for correcting erasures). Hence if $\mathcal{C}_1$ and $\mathcal{C}_2$ are amenable to implementation then so is the secure RAID scheme.

C. Secure RAID from B Codes

We design a family of efficient XOR-based schemes from the B codes, which are equivalent to perfect one-factorization of complete graphs [15], [16]. For prime p, the perfect one-factorization of K_{p+1}, the complete graph of $p+1$ vertices, is known and geometrically defines a family of B codes. We present a simplified algebraic description equivalent to this family of codes, which is useful in later constructions. For prime p, let $t = \frac{p-1}{2}$, the dual B code is a $[p-1, 2]_{\mathbb{F}_2^t}$ MDS array code. Recall the definition of $\langle \cdot \rangle$ from Section III-A.

Construction 3. (Dual B Code). *Let p be a prime, $t = \frac{p-1}{2}$ and let $m_1, ..., m_{p-1}$ be the message bits. The codeword of the dual B code forms a $t \times (p-1)$ array. The first row of the array consists of the systematic symbols, i.e., $c_{1,j} = m_j$, for $j = 1, ..., p-1$. The 2-nd to t-th rows are redundant symbols, i.e., $c_{i,j} = m_{\langle i \cdot j \rangle} \oplus m_{\langle (1-i) \cdot j \rangle}$, for $i = 2, ..., t$, $j = 1, ..., p-1$.*

Theorem 5. *The dual B codes in Construction 3 are MDS.*

From the dual B code we immediately obtain the B code.

Construction 4. (B Code). *Let p be a prime, $t = \frac{p-1}{2}$ and let $m_{i,j}$, $i \in [t-1]$, $j \in [p-1]$ be the message bits. The codeword of the B code forms a $t \times (p-1)$ array. The first $t-1$ rows of the array consists of the systematic symbols, i.e., $c_{i,j} = m_{i,j}$, for $i \in [t-1]$, $j \in [p-1]$. The t-th row consists of the redundant symbols, i.e., $c_{t,j} = \bigoplus_{k=1}^{t-1} \left(m_{k, \langle \frac{j}{k+1} \rangle} \oplus m_{k, \langle -\frac{j}{k} \rangle} \right)$, for $j \in [p-1]$.*

We present a secure RAID scheme based on the B code.

Construction 5. (Secure B). *Let p be a prime and $t = \frac{p-1}{2}$. Let $u_1, ..., u_{p-1}$ be the uniformly distributed key bits and let $m_{i,j}$, $i \in [t-2]$, $j \in [p-1]$ be the message bits. The codeword of secure B forms a $t \times (p-1)$ array. The first row of the array consists of the (relaxed) systematic key symbols, i.e., $c_{1,j} = u_j \oplus u_{\langle 2 \cdot j \rangle} \oplus u_{\langle -j \rangle}$, $j \in [p-1]$. The 2-nd to $(t-1)$-th rows are the systematic message symbols, i.e., $c_{i,j} = u_{\langle (i+1) \cdot j \rangle} \oplus u_{\langle -i \cdot j \rangle} \oplus m_{i-1,j}$, for $i \in [2, t-1]$, $j \in [p-1]$. The t-th row consists of the redundant symbols, which are computed by encoding the B code, regarding the first $(t-1)$-rows of the array as information symbols.*

Theorem 6. *Secure B is a $(p-1, p-5, 2, 2)_{\mathbb{F}_2^t}$ secure RAID scheme, for any prime p and $t = \frac{p-1}{2}$.*

An example of the scheme is shown in Fig. 4. Similar to previous discussion, the construction idea is to let $\mathcal{C}_2$ be the B code and design $\mathcal{C}_1$ to take a form similar to the dual B code, because it is low rate, MDS, and has optimal encoding complexity. However, the dual B code is not contained in the B code, and we need to design $\mathcal{C}_1$ carefully to meet $\mathcal{C}_1 \subset \mathcal{C}_2$.

We describe an efficient systematic decoding algorithm for the scheme in [4, Algorithm 1]. In case of erasures/error, the decoding algorithm of the B code [15] is invoked, followed by systematic decoding. Encoding the scheme requires on average $4 + \frac{6}{p-5}$ XORs per message bit. Systematic decoding the scheme requires on average $2 + \frac{3}{p-5}$ XORs per message bit. Encoding each message bit requires at least $4 + \frac{2}{p-5}$ XORs

Node 1	Node 2	Node 3	Node 4	Node 5	Node 6
$u_1 \oplus u_2 \oplus u_6$	$u_2 \oplus u_4 \oplus u_5$	$u_3 \oplus u_6 \oplus u_4$	$u_4 \oplus u_1 \oplus u_3$	$u_5 \oplus u_3 \oplus u_2$	$u_6 \oplus u_5 \oplus u_1$
$u_3 \oplus u_5 \oplus m_1$	$u_6 \oplus u_3 \oplus m_2$	$u_2 \oplus u_1 \oplus m_3$	$u_5 \oplus u_6 \oplus m_4$	$u_1 \oplus u_4 \oplus m_5$	$u_4 \oplus u_2 \oplus m_6$
$u_\Sigma \oplus u_1 \oplus$ $u_4 \oplus m_3 \oplus m_5$	$u_\Sigma \oplus u_2 \oplus$ $u_1 \oplus m_6 \oplus m_3$	$u_\Sigma \oplus u_3 \oplus$ $u_5 \oplus m_2 \oplus m_1$	$u_\Sigma \oplus u_4 \oplus$ $u_2 \oplus m_5 \oplus m_6$	$u_\Sigma \oplus u_5 \oplus$ $u_6 \oplus m_1 \oplus m_4$	$u_\Sigma \oplus u_6 \oplus$ $u_3 \oplus m_4 \oplus m_2$

Fig. 4: The (6,2,2,2) secure B scheme. $u_\Sigma = \bigoplus_{i=1}^{p-1} u_i$. The first row stores the (relaxed) systematic key bits, the middle row(s) stores the systematic message bits, and the last row is redundant. The scheme is optimal in the middle row(s), because each message bit is padded by exactly two keys necessary for secrecy. Furthermore, the scheme is almost optimal in the last row, because each parity must involve at least two keys for secrecy and two message bits for reliability. Hence a parity involves only one more special key u_Σ, and takes one more XOR than optimal. The scheme is slightly suboptimal in the first row of keys. However encoding this row takes $2(p-1)$ XORs which is insignificant when amortized over the $\frac{p^2-6p+5}{2}$ message bits; and decoding the keys from this row is also efficient, see [4, Algorithm 1].

by Corollary 1, and decoding each message bit requires at least 2 XORs. Hence the secure B scheme has almost optimal encoding and systematic decoding complexities.

D. Optimal secure RAID scheme from B codes

In this subsection we construct a family of strictly optimal schemes from the B codes. Let p be a prime, $t = \frac{p-1}{2}$, and $\sigma : [t] \to [t]$ be a permutation. We say that σ is *proper* with respect to p if $\sigma(1) \neq t$ and that for every codeword $C = (c_{i,j})$ of the dual B code, $(c_{\sigma(i),j})$ is a codeword of the B code.

Construction 6. (Optimal Secure B) *Let p be a prime, $t = \frac{p-1}{2}$, and σ be a proper permutation. Let $u_1, ..., u_{p-1}$ be uniformly distributed key bits. The codeword of optimal secure B forms a $t \times (p-1)$ array. The first $t-1$ rows are the systematic key and message symbols, computed as follows. Denote by $C' = c'_{i,j}$ the codeword of the dual B code computed by encoding the u_j's as information symbols and let $i^* = \sigma(1)$, then $c_{i^*,j} = u_j$, $j \in [p-1]$; for $i \neq i^*, i \in [t-1]$, $j \in [p-1]$, $c_{i,j} = c'_{\sigma(i),j} \oplus m_{i,j}$, where the $m_{i,j}$'s are the message bits. The t-th row consists of the redundant symbols, computed by encoding the B code regarding the first $(t-1)$-rows of the array as information symbols.*

Theorem 7. *Construction 6 is a $(p-1, p-5, 2, 2)_{\mathbb{F}_2^t}$ secure RAID scheme, with optimal key and message row density.*

Fig. 1 shows an example of the scheme where σ is $(1)(23)$. In general, the scheme requires an optimal number of $4 + \frac{2}{p-5}$ XORs to encode each message bit and an optimal number of 2 XORs to (systematic) decode each message bit.

We are not aware of how to construct proper permutations for an arbitrary prime. However we can efficiently check whether a given permutation is proper by [4, Lemma 5]. Therefore a proper σ, if exists, can be found by exhaustive search. Proper σ with respect to prime p, $7 \leq p \leq 53$, are listed in [4, Table I].

E. Secure RAID from Reed-Solomon Codes

We describe a rate-optimal systematic secure RAID scheme for arbitrary n, r, z. It is well known that RS codes are MDS and that low rate RS codes are subcodes of high rate RS codes [8]. To construct a (n, k, r, z) scheme, let $\mathcal{C}_2$ be a $[n, k]$ systematic RS code; let $\mathcal{C}_1$ be a $[n, z]$ systematic RS code that is contained in $\mathcal{C}_2$, and such that the systematic entries of $\mathcal{C}_1$ coincide with $\mathcal{C}_2$. A systematic scheme can be constructed as described in Section III-B. Encoding the scheme is essentially encoding $\mathcal{C}_1$ and $\mathcal{C}_2$, which takes $O((r+z)(n-r))$ operations over $\mathbb{F}_q$; systematic decoding the scheme is essentially encoding $\mathcal{C}_1$, which takes $O(z(n-z-r))$ operations; erasure/error decoding is done by erasure/error decoding $\mathcal{C}_2$ using the the Berlekamp-Massey algorithm [8], which takes $O(rn)$ operations, followed by systematic decoding.

In comparison, consider Shamir's (ramp) scheme [13], [1]. Encoding it requires evaluating a polynomial at n points which takes $O(n(n-r))$ operations; decoding it requires polynomial interpolation which takes $O((n-r)^2)$ operations by Lagrange interpolation. The proposed systematic scheme has significantly better complexity than Shamir's scheme. Particularly, in the high rate regime that r and z are fixed and n grows, encoding and systematic decoding the systematic scheme both take $O(n)$ operations, whereas encoding and decoding Shamir's scheme both take $O(n^2)$ operations. Note that though (asymptotically) efficient $O(n \log n)$ algorithms are known for encoding and decoding Shamir's scheme, they have large overhead factors and are not commonly used in practice [6]. Finally, the systematic scheme also support efficient random access, while Shamir's scheme does not.

REFERENCES

[1] G. R. Blakley and C. Meadows, "Security of ramp schemes," *Advances in Cryptology - CRYPTO*, vol. 196, pp. 242–268, 1985.

[2] M. Blaum, J. Brady, J. Bruck, and J. Menon, "EVENODD: an efficient scheme for tolerating double disk failures in RAID architectures," *IEEE Transactions on Computers*, vol. 44, no. 2, pp. 192–202, 1995.

[3] P. M. Chen, E. K. Lee, G. a. Gibson, R. H. Katz, and D. a. Patterson, "RAID: high-performance, reliable secondary storage," *ACM Computing Surveys*, vol. 26, no. 2, pp. 145–185, 1994.

[4] W. Huang and J. Bruck, "Secure RAID schemes for distributed storage," http://www.paradise.caltech.edu/papers/etr132.pdf.

[5] W. Huang, M. Langberg, J. Kliewer, and J. Bruck, "Communication efficient secret sharing," *arXiv:1505.07515*, 2015.

[6] D. Knuth, *The Art of Computer Programming*. Addison-Wesley, 1998.

[7] J. Kurihara, S. Kiyomoto, K. Fukushima, and T. Tanaka, "A new (k, n)-threshold secret sharing scheme and its extension," *ISC*, 2008.

[8] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes*. North Holland Publishing, 1977.

[9] R. J. McEliece and D. V. Sarwate, "On sharing secrets and Reed-Solomon codes," *Commun ACM*, vol. 24, no. 9, pp. 583 – 584, 1981.

[10] S. Pawar, S. E. Rouayheb, and K. Ramchandran, "Securing dynamic distributed storage systems against eavesdropping and adversarial attacks," *IEEE Trans. Info Theory*, vol. 57, no. 10, pp. 6734 – 6753, 2011.

[11] A. S. Rawat, N. S. Onur Ozan Koyluoglu and, and S. Vishwanath, "Optimal locally repairable and secure codes for distributed storage systems," *IEEE Trans. Info Theory*, vol. 60, no. 1, pp. 212–236, 2014.

[12] N. B. Shah, K. V. Rashmi, and P. V. Kumar, "Information-theoretically secure regenerating codes for distributed storage," in *Globecom*, 2011.

[13] A. Shamir, "How to share a secret," *CACM*, vol. 22, no. 11, 1979.

[14] A. Wyner, "The wire-tap channel," *Bell Syst. Tech. J.*, 1975.

[15] L. Xu, V. Bohossian, J. Bruck, and D. G. Wagner, "Low-density MDS codes and factors of complete graphs," *IEEE Transactions on Information Theory*, vol. 45, no. 6, pp. 1817–1826, 1999.

[16] G. Zaitsev, V. Zinov'ev, and N. Semakov, "Minimum-check- density codes for correcting bytes of errors, erasures, or defects," *Probl. Inform. Transm.*, vol. 19, no. 3, pp. 197–204, 1983.