Correspondence

Collusion-Secure and Cost-Effective Detection of Unlawful Multimedia Redistribution

Francesc Sebé and Josep Domingo-Ferrer

***Abstract*—Intellectual property protection of multimedia content is essential to the successful deployment of Internet content delivery platforms. There are two general approaches to multimedia copy protection: copy prevention and copy detection. Past experience shows that only copy detection based on mark embedding techniques looks promising. Multimedia fingerprinting means embedding a different buyer-identifying mark in each copy of the multimedia content being sold. Fingerprinting is subject to collusion attacks: a coalition of buyers collude and follow some strategy to mix their copies with the aim of obtaining a mixture from which none of their identifying marks can be retrieved; if their strategy is successful, the colluders can redistribute the mixture with impunity. A construction is presented in this paper to obtain fingerprinting codes for copyright protection which survive any collusion strategy involving up to three buyers (3-security). It is shown that the proposed scheme achieves 3-security with a codeword length dramatically shorter than the one required by the general Boneh-Shaw construction. Thus the proposed fingerprints require much less embedding capacity. Due to their own clandestine nature, collusions tend to involve a small number of buyers, so that there is plenty of use for codes providing cost-effective protection against collusions of size up to three.**

***Index Terms*—Buyer collusion, electronic copyright protection, fingerprinting, Internet and telecom applications and services, service creation platforms and enabling technologies, watermarking.**

I. Introduction

If multimedia content delivery services are to take off over the Internet, delivery platforms should guarantee intellectual property protection. There are two general approaches to protecting the copyright of multimedia content: one is *a priori* and consists of trying to prevent illegal copies from being made; the other is *a posteriori*, *i.e.* it tries to detect illegal copying once it has taken place. In view of the past experience in failure of copy prevention systems (the most recent being the DVD copy prevention failure, see [3]), only copy detection seems to have reasonable chances of success. Copy detection is based on mark embedding: the merchant embeds an imperceptible mark into the content before selling it [5]. There are two kinds of mark: watermarks and fingerprints. A watermark is a message that allows ownership of the marked content to be proven, whereas a fingerprint allows buyer identification [7]. Thus, fingerprints can be used to trace illegal redistributors: once a redistributed copy is found, the buyer who legally purchased it can be identified, and this legal buyer is necessarily the (first) illegal redistributor.

Collusion attacks are not an issue for watermarking (all marked copies being identical), but should be considered in the case of fingerprinting. In a collusion attack, a coalition of dishonest buyers compare their copies in order to locate differences between them and follow a strategy to fabricate a new copy of the content whose mark is either no longer recoverable or does not allow identification of any of the colluders. If the collusion strategy is successful, the fabricated content can be redistributed with impunity.

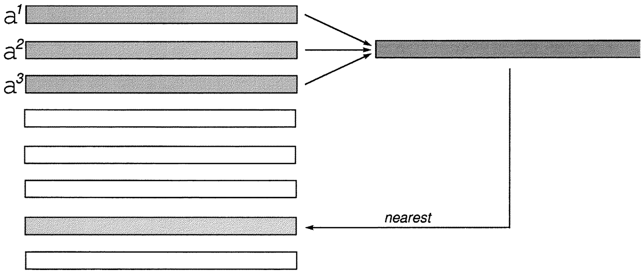


Fig. 1. Successful collusion.

In [1] and [2], the concept of fingerprinting secure against buyer collusions is introduced. A general construction is given to obtain fingerprinting codes secure against collusions of up to c buyers (c-secure codes). For N possible buyers and given $\epsilon > 0$, $L = 2c\log(2N/\epsilon)$ and $d = 8c^2\log(8cL/\epsilon)$ a code with N codewords of length

$$l = 2Ldc = 32c^4 \log\left(\frac{2N}{\epsilon}\right)\log\left(\frac{8cL}{\epsilon}\right) \tag{1}$$

is constructed which allows one of the colluders to be identified with probability $1-\epsilon$ (the fingerprint embedded in each copy sold is a different codeword of the fingerprinting code). The authors also show that, for $c \geq 2$ and $N \geq 3$, it is not possible to obtain c-secure codes where colluders are identified with probability 1.

In [4] it is shown that, for $c = 2$, collusion security can be obtained using the error-correcting capacity of dual Hamming codes. In this way, 2-secure fingerprinting codes are obtained which are much shorter than 2-secure codes obtained via the general construction [1], [2]. There are two advantages in using a shorter fingerprinting code: 1) embedding a fingerprint requires less embedding capacity (in other words, it is more imperceptible); 2) since less bits must be embedded and retrieved, the embedding and retrieval of fingerprints is more cost-effective.

We show in this paper that, for $c = 3$, it is also possible to come up with collusion-secure fingerprinting codes much shorter than 3-secure codes obtained from the general construction [1], [2]. The basic idea is to compose a new kind of code, which we call *scattering code*, with a dual Hamming code.

Section II contains an overview of our proposal. Section III presents some results on dual Hamming codes. Section IV presents a set of lemmas on the probability of successful collusion as a function of the strategy of colluders. The construction and decoding of scattering codes are introduced in Section V. Section VI explains how to generate fingerprinting codes secure against collusions of up to three buyers by composing a scattering code with a dual Hamming code. Section VII concludes by presenting some numerical results. Finally, the Appendix contains proofs for all but two lemmas presented in the paper (the two omitted proofs are nearly trivial).

II. Overview

Binary dual Hamming codes can be used to build fingerprints resistant against collusions of up to three buyers in the way proposed in this

Manuscript received August 31, 2002; revised June 3, 2003. This work was supported in part by the European Commission under project IST-2001-32012 “CO-ORTHOGONAL” and by the Spanish Ministry of Science and Technology and the European FEDER fund through Project TIC2001-0633-C03-01 “STREAMOBILE.” This paper was recommended by Guest Editors W. Pedrycz and A. Vasilakos.

F. Sebé and J. Domingo-Ferrer are with the Department of Computer Engineering and Maths, Universitat Rovira i Virgili, E-43007 Tarragona, Catalonia, Spain (e-mail: fsebe@etse.urv.es; jdomingo@etse.urv.es).

Digital Object Identifier 10.1109/TSMCC.2003.817351

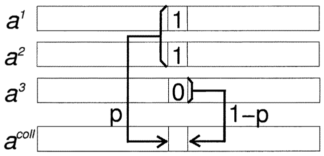


Fig. 2. The p-majority collusion strategy.

paper. A different codeword of such a code is assigned to each buyer as fingerprint; to recover the fingerprint, minimum distance decoding is used. Thus, colluders succeed if they manage to mix their copies to obtain a copy containing an embedded word such that the closest codeword to that word is a codeword assigned as a fingerprint to a buyer different from the colluders.

We model collusion attacks as so-called "p-majority" strategies. In such a strategy, three colluders wishing to mix their copies choose, for the positions where their codewords differ, the majority bit with probability p. We prove in this paper that, if the colluders follow a p-majority strategy with p close to one, the probability of identifying the three of them can be made arbitrarily close to one with the sole use of codewords of a binary dual Hamming code as fingerprints.

The problem is that p is a parameter chosen by the colluders, so they are likely to use the best choice for them, which is $p = 0$ (in fact a small p is enough for them to stay undetected with great probability). To remedy this, we propose to construct fingerprints by composing a special new kind of codes, called scattering codes and described in this paper, with dual binary Hamming codes. With such a composition, the merchant can make sure that, regardless of the p-majority strategy used by colluders to mix their composed codewords, the result of decoding the mixed composed codeword will be a word generated by a $p(v)$-majority strategy, where the probability $p(v)$ is controlled by the merchant and can be made arbitrarily close to one. Thus, the probability that the three colluders can be identified can be made arbitrarily close to one by our construction.

III. Dual Binary Hamming Codes

The dual code of a binary Hamming code (denoted by $DH(n)$) is a binary code with 2^n codewords of length $N = 2^n - 1$ such that the distance between any two codewords is 2^{n-1}. A few definitions and useful properties related to such codes are presented next.

Definition 1: Let a^1, a^2, a^3 be three codewords of a $DH(n)$ code, *i.e.*, $a^i = a_1^i a_2^i \cdots a_N^i$. Define $inv(a^1, a^2, a^3)$ to be the set of invariant positions between all three codewords, that is, those bit positions in which all three codewords have the same bit value. Formally speaking

$$inv(a^1, a^2, a^3) = \left\{i, 1 \leq i \leq N,\ a_i^1 = a_i^2 = a_i^3\right\}.$$

Definition 2: Let a^1, a^2, a^3 be three codewords of a $DH(n)$ code. Define $minor(a^1; a^2, a^3)$ to be the set of bit positions in which a^1 has a value different from the values in a^2 and a^3 (for such positions, $a_i^2 = a_i^3$). Formally expressed

$$minor(a^1; a^2, a^3) = \left\{i, 1 \leq i \leq N, a_i^1 \neq a_i^2, a_i^1 \neq a_i^3\right\}.$$

Lemma 1: Let a^1, a^2, a^3 be three codewords of a $DH(n)$ code and let $|\cdot|$ denote the bitlength operator. Then it holds that $|\mathrm{inv}(\mathrm{a}^1, \mathrm{a}^2, \mathrm{a}^3)| = 2^{\mathrm{n}-2} - 1$, $|\mathrm{minor}(a^1; a^2, a^3)| = 2^{n-2}$, $|\mathrm{minor}(a^2; a^1, a^3)| = 2^{n-2}$ and $|\mathrm{minor}(a^3; a^1, a^2)| = 2^{n-2}$.

	$inv123$	$min123$	$min213$	$min312$
a^1	0000000	11111111	00000000	11111111
a^2	0000000	00000000	11111111	11111111
a^3	0000000	00000000	00000000	00000000

TABLE I

Expected Number of Differing Bits Between a Word a^{coll} Generated Using a p-Majority Strategy and any of the Colluders' Codewords. The Code is a $DH(6)$

p	0	0.2	0.4	0.6	0.8	1
$E(d_1)$	32	28.8	25.6	22.4	19.2	16

Example 1: The following are three codewords of a $DH(5)$ code. In the table above, $\mathrm{inv123}$ stands for $\mathrm{inv}(\mathrm{a}^1, \mathrm{a}^2, \mathrm{a}^3)$, $\mathrm{min123}$ stands for minor$(a^1; a^2, a^3)$, $\mathrm{min213}$ stands for minor$(a^2; a^1, a^3)$ and $\mathrm{min312}$ stands for minor$(a^3; a^1, a^2)$. The codeword length is $2^5 - 1 = 31$. Now, $|\mathrm{inv}(\mathrm{a}^1, \mathrm{a}^2, \mathrm{a}^3)| = 2^{5-2} - 1 = 7$, $|\mathrm{minor}(a^1; a^2, a^3)| = |\mathrm{minor}(a^2; a^1, a^3)| = |\mathrm{minor}(a^3; a^1, a^2)| = 2^{5-2} = 8$.

Lemma 2: Let a^1, a^2, a^3 be three codewords of a $DH(n)$ code, then it holds that:

1) there exists one and only one codeword $a^z \in DH(n)\backslash\{a^1, a^2, a^3\}$ such that $a_i^z = a_i^1 = a_i^2 = a_i^2$, $\forall\ i \in \mathrm{inv}(\mathrm{a}^1, \mathrm{a}^2, \mathrm{a}^3)$. Furthermore, $a_i^z = a_i^1, \forall i \in$ minor$(a^1; a^2, a^3)$, $a_i^z = a_i^2$, $\forall\ i \in$ minor$(a^2; a^1, a^3)$ and $a_i^z = a_i^3$, $\forall\ i \in$ minor$(a^3; a^1, a^2)$.
2) remaining codewords satisfy that $\forall\ a^j \in DH(n)\backslash\{a^1, a^2, a^3, a^z\}$, $d_{inv(a^1,a^2,a^3)}(a^j, a^1) = d_{minor(a^1;a^2,a^3)}(a^j, a^1) = d_{minor(a^2;a^1,a^3)}(a^j, a^1) = d_{minor(a^3,a^1,a^2)}(a^j, a^1) = 2^{n-3}$, where $d_P(x, y)$ denotes Hamming distance between codewords x and y restricted to bit positions in P. The same distances hold with respect to a^2 and a^3.

Example 2: The table following shows the unique codeword a^z corresponding to three particular codewords a^1, a^2, a^3 of a $DH(5)$ code:

	$inv123$	$min123$	$min213$	$min312$
a^1	0000000	11111111	00000000	11111111
a^2	0000000	00000000	11111111	11111111
a^3	0000000	00000000	00000000	00000000
a^z	0000000	11111111	11111111	00000000

In the previous table, $\mathrm{inv123}$, $min123$, $\mathrm{min213}$ and $\mathrm{min312}$ have the same meanings as in the table of Example 1. It can be seen that $a_i^z = a_i^1 = a_i^2 = a_i^3$, $\forall\ i \in \mathrm{inv}(\mathrm{a}^1, \mathrm{a}^2, \mathrm{a}^3)$. Also, $a_i^z = a_i^1$, $\forall\ i \in$ minor$(a^1; a^2, a^3)$, $a_i^z = a_i^2$, $\forall\ i \in$ minor$(a^2; a^1, a^3)$ and $a_i^z = a_i^3, \forall\ i \in$ minor$(a^3; a^1, a^2)$.

Example 3: The table following displays three codewords a^1, a^2, a^3 of a $DH(5)$ code and another codeword $a^i \in DH(5)\backslash\{a^1, a^2, a^3, a^z\}$. The meaning of $\mathrm{inv123}$, $\mathrm{min123}$, $\mathrm{min213}$ and $\mathrm{min312}$ is as above.

	$inv123$	$min123$	$min213$	$min312$
a^1	0000000	11111111	00000000	11111111
a^2	0000000	00000000	11111111	11111111
a^3	0000000	00000000	00000000	00000000
a^i	0001111	00001111	00001111	00001111

It can be seen that $d_{\mathrm{inv}(\mathrm{a}^1,\mathrm{a}^2,\mathrm{a}^3)}(a^i, a^1)= d_{\mathrm{minor}(\mathrm{a}^1;\mathrm{a}^2,\mathrm{a}^3)}(a^i, a^1)= d_{\mathrm{minor}(\mathrm{a}^2;\mathrm{a}^1,\mathrm{a}^3)}(a^i, a^1)= d_{\mathrm{minor}(\mathrm{a}^3,\mathrm{a}^1,\mathrm{a}^2)}(a^i, a^1)= 2^{n-3} = 4$. The same distances hold between a^i and a^2, a^3.

IV. 3-Collusions Over $DH(n)$

A. Detectable Positions

Let us assume that three dishonest buyers c^1, c^2, c^3 compare their copies of the same multimedia content. According to the marking assumption [1], they can only modify the embedded marks in those *detectable* positions, *i.e.*, those where not all three marks take the same bit value. In those positions, the colluders can set the corresponding bit to "0," "1," or "unreadable." In this way, we conclude that, if three different buyers are assigned codewords a^1, a^2, and a^3 of a $DH(n)$ code, the result of their collusion will be a word a^{coll} where no bit has been modified in the $2^{n-2}-1$ positions in $inv(a^1, a^2, a^3)$. On the other hand, colluders will be able to detect and identify positions in minor($a^1; a^2, a^3$) as the bit positions of those content fragments which are identical between the copies of c^2 and c^3 and different from the copy of c^1. In a similar way, minor($a^2; a^1, a^3$) and minor($a^3; a^1, a^2$) can be identified as well.

B. Decoding by Minimum Distance

As said above, colluders can generate a new object whose embedded codeword may have been altered in detectable positions. In this way, it is possible that the word retrieved from a collusion-generated object does not correspond to any $DH(n)$ codeword. In these situations, the recovered word will be error-corrected by minimum distance.

Thus, in order for a collusion to be successful, colluders c^1, c^2, c^3 (whose assigned codewords are a^1, a^2, a^3) must generate, by mixing fragments of their copies, a word such that the closest codeword in the $DH(n)$ code is not in $\{a^1, a^2, a^3\}$ (see Fig. 1). In this way, another buyer will be accused in lieu of the colluders. Note that we are assuming colluders do not generate "unreadable" positions when colluding over $DH(n)$ codewords. It will be shown later that our construction actually prevents unreadable positions from being fed by colluders to the dual Hamming decoder.

C. Objective of Colluders

As decoding is done by minimum distance, the objective of colluders is to come up with an object whose embedded word is as distant as possible from their assigned codewords.

Intuitively, it can be realized that all colluders must contribute the same number of bits from their corresponding codewords. Otherwise, the collusion-generated word would be closer to the codewords of those colluders having contributed more bits.

Definition 3: A p-*majority* collusion strategy is one in which colluders choose with probability p the majority bit value in positions minor($a^i; a^j, a^k$) (that is, the bit values in a^j or a^k) (See Fig. 2).

It can be seen that a word generated using a p-majority strategy on $a^1, a^2, a^3 \in DH(n)$ is expected to be at the same distance from a^1, a^2 and a^3.

D. Distance From a Collusion-Generated Word to Colluders' Codewords

Lemma 3: Let a^{coll} be a word that has been generated using a p-majority collusion strategy between three codewords $a^1, a^2, a^3 \in DH(n)$. It holds that $d_1 = d(a^{coll}, a^i) = K_1, \forall\, i = 1$, 2, 3 with

$$\begin{aligned} p_1(k) =& p(K_1 = k) = \\ =& \sum_{t=\max(0,k-2^{n-1})}^{\min(k,2^{n-2})} b(t; 2^{n-2}, p) b(k-t; 2^{n-1}, 1-p) \end{aligned}$$

where $b(x_1; x_2, x_3)$ is the binomial probability function (x_2 is the number of trials, x_3 the success probability per trial and x_1 is the number of successful trials).

Remarks: The total number of differing bits is the addition of two binomial random variables. We use this fact to compute its expected value as

$$E(d_1) = p \cdot 2^{n-2} + (1-p)2^{n-1} = 2^{n-1} - p \cdot 2^{n-2}.$$

TABLE II
EXPECTED NUMBER OF DIFFERING BITS BETWEEN A WORD $a^{\rm coll}$ GENERATED USING A p-MAJORITY STRATEGY AND THE NEAREST [$E(d_2)$] AND THE FARTHEST [$E(d_3)$] COLLUDER CODEWORD. THE CODE IS A $DH(6)$

p	0	0.2	0.4	0.6	0.8	1
$E(d_2)$	32	26.5	22.7	19.5	16.9	16
$E(d_3)$	32	31.12	28.46	25.27	21.54	16

TABLE III
EXPECTED NUMBER OF DIFFERING BITS BETWEEN A WORD $a^{\rm coll}$ GENERATED USING A p-MAJORITY STRATEGY ($p > 0.\hat{6}$) AND THE NEAREST CODEWORD NOT IN THE COLLUSION. THE CODE IS A $DH(6)$

p	$0.\hat{6}$	0.8	1
$E(d_6)$	24.5	25.6	32

As can be seen in Table I, the expected number of differing bits between the word a^{coll} generated by a collusion and any of the colluders' codewords [$a^1, a^2, a^3 \in DH(6)$] decreases as the value p gets closer to 1 (a^{coll} gets closer to a^1, a^2, a^3).

Lemma 4: Let a^{coll} be a word generated using a p-majority collusion strategy between three codewords $a^1, a^2, a^3 \in DH(n)$. It holds that $d_2 = \min_{i=1,2,3} d(a^{coll}, a^i) = K_2$ with

$$p_2 = p(K_2 = k) = \sum_{i=1}^{3} \binom{3}{i} p_1(k)^i \left[\sum_{k'>k} p_1(k')\right]^{3-i}.$$

Lemma 5: Let a^{coll} be a word generated using a p-majority collusion strategy between three codewords $a^1, a^2, a^3 \in DH(n)$. It holds that $d_3 = \max_{i=1,2,3} d(a^{coll}, a^i) = K_3$ with

$$p_3 = p(K_3 = k) = \sum_{i=1}^{3} \binom{3}{i} p_1(k)^i \left[\sum_{k'<k} p_1(k')\right]^{3-i}.$$

See Table II, for a numerical example of d_2 and d_3.

E. Distance From a Collusion-Generated Word to Codewords Not in the Collusion

Lemma 6: Let a^{coll} be a word generated using a p-majority strategy between three codewords $a^1, a^2, a^3 \in DH(n)$ and let a^z be the only codeword in $DH(n)\backslash\{a^1, a^2, a^3\}$ with $a_i^z = a_i^1 = a_i^2 = a_i^3, \forall\, i \in inv(a^1, a^2, a^3)$ (the existence and uniqueness of a^z are guaranteed by Lemma 2). Then, $d_4 = d(a^z, a^{coll}) = K_4$ with

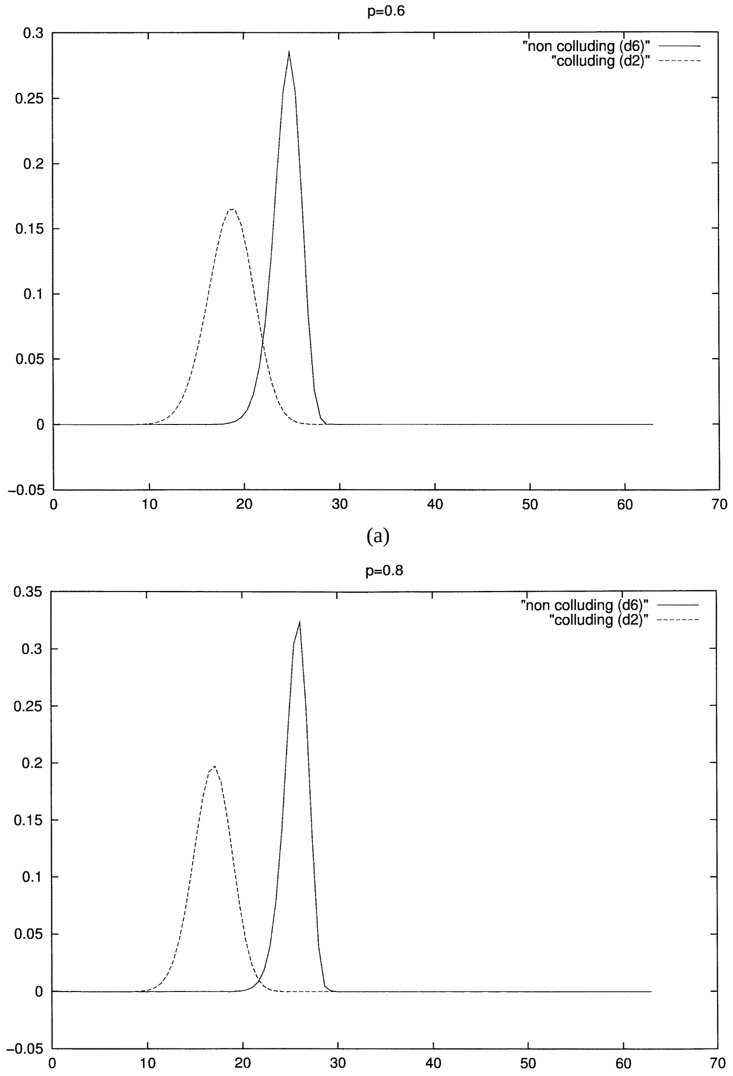
$$p_4(k) = p(K_4 = k) = b(k; 3 \cdot 2^{n-2}, p).$$

Remarks: The expected number of differing bits between a^z and a^{coll} is

$$E(d_4) = p \cdot 3 \cdot 2^{n-2}$$

Lemma 7: Let a^{coll} be a word generated using a p-majority strategy between three codewords $a^1, a^2, a^3 \in DH(n)$ and let a^z be the only codeword in $DH(n)\backslash\{a^1, a^2, a^3\}$ with $a_i^z = a_i^1 = a_i^2 = a_i^3, \forall i \in inv(a^1, a^2, a^3)$. Then, for any codeword $a \in DH(n)\backslash\{a^1, a^2, a^3, a^z\}$ it holds that $d_5 = d(a, a^{coll}) = 2^{n-3} + K_5$ with

$$\begin{aligned} p_5(k) =& p(K_5 = k) \\ =& \sum_{t=\max(0,k-3\cdot^{n-1})}^{\min(k,3\cdot^{n-3})} \\ & b(t; 3 \cdot 2^{n-3}, 1-p) b(k-t; 3 \cdot 2^{n-3}, p). \end{aligned}$$



(a)

(b)

Fig. 3. Distribution of d_2 and d_6 for (a) $p = 0.\hat{6}$ and (b) $p = 0.8$. The code is a $DH(6)$.

Remarks: The expected number of differing bits between a and a^{coll} is

$$E(d_5) = 2^{n-3} + 3 \cdot 2^{n-3}(1-p) + 3 \cdot 2^{n-3}p = 2^{n-1}.$$

For the sake of simplicity, let us assume in what follows that d_4 is distributed like d_5. Since for $p > 0.\hat{6}$ the number of differing bits expected for d_4 is greater than the number of differing bits expected for d_5 $(E(d_4) > E(d_5) \Leftrightarrow p \cdot 3 \cdot 2^{n-2} > 2^{n-1} \Leftrightarrow p > 0.\hat{6})$, such a distributional assumption will cause actual security to be even slightly higher than computed in what follows.

Lemma 8: Let a^{coll} be a word generated using a p-majority strategy $(p > 0.\hat{6})$ between three codewords $a^1, a^2, a^3 \in DH(n)$. It holds that $d_6 = \min_{i \notin \{1,2,3\}}\{d(a^{coll}, a^i)\} = 2^{n-3} + K_6$, with

$$\begin{aligned} p_6(k) =& p(K_6 = k) \\ =& \sum_{i=1}^{2^n-3} \binom{2^n-3}{i} p_5(k)^i \left[\sum_{k'>k} p_5(k')\right]^{2^n-3-i}. \end{aligned}$$

See Table III for a numerical example of d_6.

As can be seen in Fig. 3, when $p > 0.\hat{6}$, d_2 tends to take smaller values than d_6. This means that, with high probability, the codeword in $DH(n)$ nearest to the collusion-generated word is a colluder codeword.

F. Identifying Colluders' Codewords

Lemma 9: Let a^{coll} be a word generated using a p-majority strategy $(p > 0.\hat{6})$ between three codewords $a^1, a^2, a^3 \in DH(n)$. The probability that the codeword in $DH(n)$ nearest to a^{coll} is *not* in $\{a^1, a^2, a^3\}$ is expressed by

$$\epsilon = \sum_{k=0}^{2^n-1} p(d_2 = k)p(d_6 \le k).$$

ϵ is the probability that decoding a^{coll} yields as a result a codeword different from any of the colluders' codewords, that is, the probability of an honest buyer being unjustly accused instead of the colluders. The formula above for ϵ is straightforward from the definitions of d_2 and d_6, so its proof is omitted in the Appendix.

Remarks: It can be observed from Table IV that, as n increases and p approaches 1, the probability ϵ of accusing an innocent buyer can be made arbitrarily close to 0.

The following result follows from the definitions of d_3 and d_6, so its proof is omitted in the Appendix as well.

Lemma 10: Let a^{coll} be a word generated using a p-majority strategy $(p > 0.\hat{6})$ between three codewords $a^1, a^2, a^3 \in DH(n)$.

TABLE IV
PROBABILITY ϵ OF SUCCESS OF A 3-COLLUSION IN $DH(7)$ AND $DH(8)$ FOR SEVERAL VALUES OF p

	p			
	0.7	0.8	0.9	1.0
$DH(7)$	$0.14 \cdot 10^{-3}$	$0.14 \cdot 10^{-6}$	$0.77 \cdot 10^{-14}$	0.0
$DH(8)$	$0.10 \cdot 10^{-7}$	$0.15 \cdot 10^{-13}$	$0.70 \cdot 10^{-28}$	0.0

The probability that the three codewords in $DH(n)$ nearest to a^{coll} are $\{a^1, a^2, a^3\}$ is expressed by

$$\sum_{k=0}^{2^n-1} p(d_3 = k)p(d_6 > k).$$

Remarks: It can be observed from Table V that as n increases and p approaches 1, the probability of not identifying all three colluders can be made arbitrarily close to 0.

The problem is that the parameter p defining the collusion strategy is chosen by the colluders, which Implies they can take $p = 0$ to make sure they are not identified! In Section V, a new kind of codes named scattering codes are presented. These codes are used in Section VI to prevent colluders from avoiding identification in this way.

V. SCATTERING CODES

A. Construction

A *scattering code* $SC(d,t)$ with parameters (d,t) can be defined as a binary code consisting of $2t$ codewords of length $(2t+1)d$ constructed as follows.

1) Construction starts with generation of $SC(1,t)$.
   a) i-th codeword for $1 \leq i \leq t$ is constructed by setting the first and the $(i+1)$-th bits of the codeword to "1." The remaining bits are set to "0."
   b) i-th codeword for $t+1 \leq i \leq 2t$ is constructed by setting the $(i+1)$-th bit of the codeword to "1." The remaining bits are set to "0."
2) Code $SC(d,t)$ is generated by replicating d times every column of $SC(1,t)$. Define a *block* to be a group of d replicated columns.
3) By convention, the first t codewords of $SC(d,t)$ are defined to encode a "1" and the last t codewords are defined to encode a "0". The first block of the code is called "Zone-A," the next t blocks are called "Zone-B" and the last t blocks are called "Zone-C."

Using a Scattering Code, a "1" is encoded by randomly choosing one of the first t codewords and a "0" is encoded by randomly choosing one of the last t codewords (Table VI shows the codewords of a $SC(2,3)$ code).

B. Decoding

A scattering code is decoded by using the first applicable rule among the following ordered list.

1) If all bits in "Zone-A" are "1" and all bits in "Zone-C" are "0," decode as "1."
2) If all bits in "Zone-A" are "0" and all bits in "Zone-B" are "0," decode as "0."
3) If in two blocks of "Zone-B" there is at least one bit in each with value "1," decode as "1."
4) If in two blocks of "Zone-C" there is at least one bit in each with value "1," decode as "0."
5) If there are more "1" bits than "0" bits in "Zone-A," decode as "1."
6) If there are more "0" bits than "1" bits in "Zone-A," decode as "0."
7) Decode as "Unreadable"

TABLE V
PROBABILITY OF *NOT* IDENTIFYING ALL THREE COLLUDERS IN $DH(7)$ AND $DH(8)$ FOR SEVERAL VALUES OF p

	p			
	0.7	0.8	0.9	1.0
$DH(7)$	$0.5 \cdot 10^{-1}$	$0.1 \cdot 10^{-2}$	$0.25 \cdot 10^{-7}$	0.0
$DH(8)$	$0.26 \cdot 10^{-3}$	$0.6 \cdot 10^{-7}$	$0.2 \cdot 10^{-16}$	0.0

TABLE VI
CODEWORDS OF A SCATTERING CODE $SC(2,3)$

Encodes	Zone-A	Zone-B			Zone-C		
'1'	11	11	00	00	00	00	00
	11	00	11	00	00	00	00
	11	00	00	11	00	00	00
'0'	00	00	00	00	11	00	00
	00	00	00	00	00	11	00
	00	00	00	00	00	00	11

Note: It is easy to see that an odd value for d makes Rule 7 unreachable, thus causing a "0" or "1" to be always returned.

Lemma 11: Let b^{coll} be a word generated by using a p-majority strategy between three codewords $b^1, b^2, b^3 \in SC(d,t)$ encoding the same bit value v. Then, b^{coll} decodes as v with probability 1.

Lemma 12: Let b^{coll} be a word generated using a p-majority strategy between three codewords $b^1, b^2, b^3 \in SC(d,t)$, with two of them ($b^1$ and b^2) encoding a value v and the other (b^3) a value $\overline{v}$. Then, the probability that b^{coll} decodes as v is given by

$$p(v) = \left(1 - \frac{1}{t}\right) p_{dif}(v) + \frac{1}{t} p_{coi}(v)$$

where $p_{dif}(v)$ is the probability of decoding as v when $b^1 \neq b^2$ and can be computed as $p_{dif}(v) = 1 - p_{dif}(\overline{v})$ (we assume d to have an odd value) and

$$\begin{aligned} p_{dif}(\overline{v}) =& (1-p)^d p^{2d} \\ &+ 2 \cdot p^d (1 - p^d) \sum_{k=0}^{\lfloor d-1/2 \rfloor} b(k; d, p) \\ &+ p^{2d} \sum_{k=1}^{\lfloor d-1/2 \rfloor} b(k; d, p) \end{aligned}$$

and $p_{coi}(v)$ is the probability of decoding as v when $b^1 = b^2$ and can be computed as

$$\begin{aligned} p_{coi}(v) =& p^{2d} \\ &+ (1 - p^d) \sum_{k=\lfloor d+2/2 \rfloor}^{d} b(k; d, p) \\ &+ p^d \sum_{k=\lfloor d+2/2 \rfloor}^{d-1} b(k; d, p). \end{aligned}$$

See Fig. 4 for a plot of $p(v)$ as a function of p for different scattering codes.

VI. 3-SECURE CODES

A. Construction

For $N = 2^n$ buyers, each buyer c^i is assigned a different codeword $a^i \in DH(n)$. Rather than directly embedding a^i in the content to be sold, the merchant generates a codeword A^i by composing a scattering code $SC(d,t)$ with a^i. Such a composition is performed by replacing each bit of a^i with a codeword in $SC(d,t)$ that encodes the value of that bit of a^i. In this way, the codeword A^i will have bitlength

$$l = (N-1)(2t+1)d. \tag{2}$$

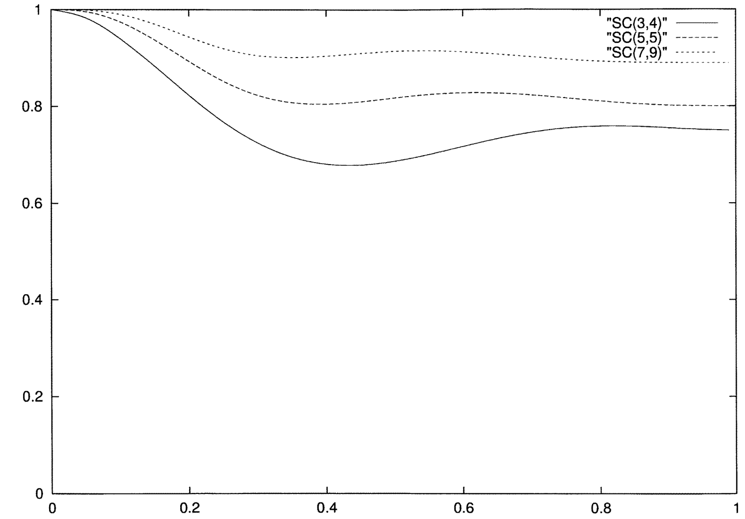


Fig. 4. For different values of d and t, the graphic depicts the probability of decoding the majority value $p(v)$ as a function of the p-majority strategy applied.

The merchant then permutes the bits in A^i using a pseudo-random permutation seeded by a secret key known only to the merchant. The same permutation is applied to all codewords. A^i. Fig. 5 graphically depicts the construction described in this section. Finally, the merchant embeds the permuted version of A^i in the content being sold.

B. Three-Collusions

Let us suppose three dishonest buyers c^1, c^2, c^3 are assigned three codewords A^1, A^2, A^3 which have been built by: 1) composing a scattering code with three different codewords $a^1, a^2, a^3 \in DH(n)$; 2) permuting the bits of the composed codewords.

By comparison of their copies, the colluding dishonest buyers can identify minor$(A^1; A^2, A^3)$, minor$(A^2; A^1, A^3)$ and minor$(A^3; A^1, A^2)$. But as the bits of A^i have been secretly permuted, the colluders cannot find out which bit of A^i corresponds to which bit of a^i. Thus, the colluders cannot identify minor$(a^1; a^2, a^3)$, minor$(a^2; a^1, a^3)$ nor minor$(a^3; a^1, a^2)$. Therefore, the only way for colluders to generate a A^{coll} is a to follow a p-majority strategy.

According to Lemma 9, all bits at positions $inv(a^1, a^2, a^3)$ remain unmodified after decoding each of the $2^n - 1$ components of A^{coll} to obtain a^{coll}. Also, according to Lemma 10, all decoded bits at positions minor$(a^i; a^j, a^k)$ for $(i,j,k) \in \{(1,2,3),(2,1,3),(3,1,2)\}$ keep the majority value v (the one of a^j and a^k) with probability $p(v)$.

What is achieved with the above composition is that, regardless of the p-majority strategy used by colluders to generate A^{coll}, the word a^{coll} resulting from decoding A^{coll} is a word generated by a $p(v)$-majority strategy collusion between a^1, a^2, a^3, where the value $p(v)$ is controlled by the merchant by choosing appropriate values for parameters d and t (see Table VII, Table VIII, Table IX and Fig. 5). It can be seen from Table IV that Controlling $p(v)$ is necessary to the keep low the probability ϵ of successful collusion. If A^i has some bits with value 'Unreadable', those bits are randomly set to "0" or "1."

VII. Numerical Results and Conclusions

Once parameters d and t have been fixed, the number of buyers can be increased by increasing n. For $d = 5$ and $t = 5$, Table X shows the size of the code (number of buyers), the codeword length of our proposal, the probability of a successful collusion ϵ and the codeword length of Boneh-Shaw's proposal for the same n and ϵ.

It can be seen that Boneh-Shaw's construction results in much longer codewords than our proposal. Further, as n increases, their codeword length increases faster than ours.

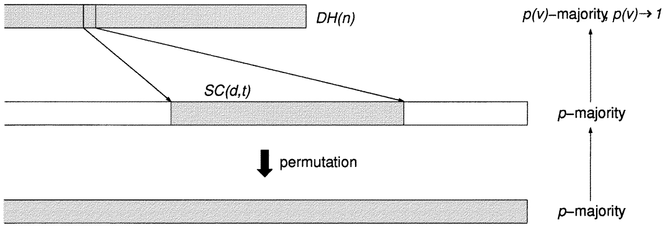


Fig. 5. Construction of three-secure codes.

TABLE VII
Possible Collusion of Three Codewords of a $SC(2,3)$ Code With $b^1 \neq b^2$ Both Encoding A "1" and b^3 Encoding A "0"

	Zone-A	Zone-B			Zone-C		
b^1	11	11	00	00	00	00	00
b^2	11	00	11	00	00	00	00
b^3	00	00	00	00	11	00	00

TABLE VIII
Possible Collusion of Three Codewords of a $SC(2,3)$ Code With $b^1 = b^2$ Encoding A "1" and b^3 Encoding A "0"

	Zone-A	Zone-B			Zone-C		
b^1	11	11	00	00	00	00	00
b^2	11	11	00	00	00	00	00
b^3	00	00	00	00	11	00	00

TABLE IX
Lowest Probability $p(v)$ of Decoding as the Majority Bit v in a Collusion of Three Buyers, for Several Parameter Choices (d, t)

d	t	$\min p(v)$
3	4	0.68
5	5	0.8
7	9	0.89
31	100	0.99

TABLE X
Codeword Length Comparison Between our Proposal and Boneh-Shaw's for Several Numbers of Users and Security Levels

n	buyers	ϵ	Our length	B-S length
7	128	$0.14 \cdot 10^{-6}$	6985	$2,788,320$
8	256	$0.15 \cdot 10^{-13}$	14025	$8,393,220$
9	512	$0.19 \cdot 10^{-27}$	28105	$28,340,928$

TABLE XI
CODEWORD LENGTH COMPARISON BETWEEN OUR PROPOSAL AND BONEH-SHAW'S ASSUMING $\epsilon = 10^{-10}$

buyers	Our length	Boneh-Shaw's length
512	28,105	5,148,000
1,024	56,265	5,269,992
...	...	...
32,768	1,802,185	5,883,888
65,536	3,604,425	6,006,780
131,072	7,208,905	6,129,816

In our proposal, once d and t have been fixed, the value ϵ decreases exponentially as n increases, which yields security levels higher than needed. Thus, a better comparison is to use a fixed ϵ and assume that, for our security requirements, $\forall\, \epsilon' < \epsilon$ one has $\epsilon' \approx 0$. We take a value $\epsilon = 10^{-10}$ and use it as security level for Boneh-Shaw's construction. Results are presented in Table XI.

For a fixed $\epsilon = 10^{-10}$, we can observe that our proposal yields shorter codeword lengths up to $n = 16$ (number of buyers is 65 536). For values of $n > 16$ Boneh-Shaw's proposal offers a shorter codeword length. The explanation is that our codeword length increases as $O(N)$, while Boneh-Shaw's increases as $O(\log N)$ with a large constant factor; this large constant factor prevents Boneh-Shaw's scheme from comparing favorably unless N is very large.

APPENDIX
PROOF OF LEMMAS

Proof: (Lemma 1): Let a^1, a^2, a^3 be three codewords of a $DH(n)$ code. Define $I = \mathrm{inv}(\mathrm{a}^1, \mathrm{a}^2)$ and $\overline{I}$ the positions not in I. Since $d(a^i, a^j)_{i \neq j} = 2^{n-1}$, then $|I| = 2^{n-1} - 1$.

Let $x = |\mathrm{inv}(\mathrm{a}^1, \mathrm{a}^2, \mathrm{a}^3)|$ (obviously, $\mathrm{inv}(\mathrm{a}^1, \mathrm{a}^2, \mathrm{a}^3) \subset \mathrm{I}$) and let y be the total number of positions $i \in \overline{I}$ where $a_i^2 = a_i^3$ (these are the positions that form minor($a^1; a^2, a^3$)). As $d(a^2, a^3) = 2^{n-1}$, then $x + y = 2^{n-1} - 1$.

There are $2^{n-1} - 1 - x$ positions $i \in I$ where $a_i^3 \neq a_i^1$ (these are the positions that form minor($a^3; a^1, a^2$)) and y positions $i \in \overline{I}$ where $a_i^3 \neq a_i^1$. As $d(a^3, a^1) = 2^{n-1}$ we have that $2^{n-1} - 1 - x + y = 2^{n-1}$.

By solving

$$\begin{cases} x + y = 2^{n-1} - 1 \\ 2^{n-1} - 1 - x + y = 2^{n-1} \end{cases}$$

we get $x = 2^{n-2} - 1$ and $y = 2^{n-2}$. Finally, we conclude that $|\mathrm{inv}(\mathrm{a}^1, \mathrm{a}^2, \mathrm{a}^3)| = \mathrm{x} = 2^{\mathrm{n}-2} - 1$, $|\mathrm{minor}(a^1; a^2, a^3)| = y = 2^{n-2}$, $|\mathrm{minor}(a^2; a^1, a^3)| = 2^{n-1} - y = 2^{n-2}$, $|\mathrm{minor}(a^3; a^1, a^2)| = 2^{n-1} - 1 - x = 2^{n-2}$. ■

Proof: (Lemma 2): First of all, we prove the existence and properties of a^z. As a $DH(n)$ code is a linear code, any linear combination of codewords results in another codeword. Then, we compute $a^z = a^1 \oplus a^2 \oplus a^3$, where $\oplus$ denotes the component-wise modulo 2 addition.

We prove that $a_i^z = a_i^1 = a_i^2 = a_i^3, \forall\, i \in inv(a^1, a^2, a^3)$. This is true because, if $a_i^1 = a_i^2 = a_i^3 = 1$, then $a_i^1 \oplus a_i^2 \oplus a_i^3 = 1$, and if $a_i^1 = a_i^2 = a_i^3 = 0$, then $a_i^1 \oplus a_i^2 \oplus a_i^3 = 0$.

Then, we prove $a_i^z = a_i^1, \forall\, i \in \mathrm{minor}(a^1; a^2, a^3)$. This is true because $a_i^z = a_i^1 \oplus a_i^2 \oplus a_i^3$ and, as $a_i^2 = a_i^3$, then $a_i^z = a_i^1$.

Using the same procedure, we can prove $a_i^z = a_i^2, \forall\, i \in \mathrm{minor}(a^2; a^1, a^3)$ and $a_i^z = a_i^3, \forall\, i \in \mathrm{minor}(a^3; a^1, a^2)$.

Next, we prove the second part of the Lemma. Let $a^j \in DH(n) \backslash \{a^1, a^2, a^3, a^z\}$.

Call x the number of positions in $inv(a^1, a^2, a^3)$ where $a_i^j = a_i^1$. Then the number of positions in $inv(a^1, a^2, a^3)$ where $a_i^j \neq a_i^1$ is $2^{n-2} - 1 - x$ (see Lemma 1).

Call y the number of positions in $minor(a^1; a^2, a^3)$ where $a_i^j = a_i^1$. Then the number of positions in $minor(a^1; a^2, a^3)$ where $a_i^j \neq a_i^1$ is $2^{n-2} - y$.

Call z the number of positions in $minor(a^2; a^1, a^3)$ where $a_i^j = a_i^1$. Then the number of positions in $minor(a^2; a^1, a^3)$ where $a_i^j \neq a_i^1$ is $2^{n-2} - z$.

Call t the number of positions in $minor(a^3; a^1, a^2)$ where $a_i^j = a_i^1$. Then the number of positions in $minor(a^3; a^1, a^2)$ where $a_i^j \neq a_i^1$ is $2^{n-2} - t$.

From the expressions at the bottom of the page we build the following equation system:

$$\begin{cases} x + y + z + t = 2^{n-1} - 1 \\ -x - y + z + t = 1 \\ -x + y - z + t = 1 \\ -x + y + z - t = 1. \end{cases}$$

By solving it, we get $x = 2^{n-3} - 1$ and $y = z = t = 2^{n-3}$. Finally, we conclude

$$\begin{aligned} d_{inv(a^1,a^2,a^3)}(a^j, a^1) &= 2^{n-2} - 1 - x = 2^{n-3} \\ d_{\mathrm{minor}(\mathrm{a}^1;\mathrm{a}^2,\mathrm{a}^3)}(a^j, a^1) &= 2^{n-2} - y = 2^{n-3} \\ d_{\mathrm{minor}(\mathrm{a}^2;\mathrm{a}^1,\mathrm{a}^3)}(a^j, a^1) &= 2^{n-2} - z = 2^{n-3} \\ d_{\mathrm{minor}(\mathrm{a}^3;\mathrm{a}^1,\mathrm{a}^2)}(a^j, a^1) &= 2^{n-2} - t = 2^{n-3}. \end{aligned}$$

In an analogous way, we can prove that the distances between a^j and a^2, a^3 are 2^{n-3} as well. ■

Proof: (Lemma 3) : Without loss of generality, take $i = 1$. We have that, for bit positions in $\mathrm{inv}(\mathrm{a}^1, \mathrm{a}^2, \mathrm{a}^3)$, there is no difference between a^1 and a^{coll} since bits in those positions are undetectable. Also, each of the 2^{n-2} bits in minor($a^1; a^2, a^3$) differs between a^1 and a^{coll} with probability p; therefore, the probability of there being t differing bits in those positions is given by a binomial probability function $b(t; 2^{n-2}, p)$. Also, each of the $2 \cdot 2^{n-2}$ bits in minor($a^2; a^1, a^3$) and minor($a^3; a^1, a^2$) differs between a^1 and a^{coll} with probability $(1-p)$; therefore, the probability of there being $k-t$ differing bits in those positions is given by a binomial probability function $b(k-t; 2^{n-1}, 1-p)$.

As $d(a^j, a^1) = d_{inv(a^1,a^2,a^3)}(a^j, a^1) + d_{\mathrm{minor}(\mathrm{a}^1;\mathrm{a}^2,\mathrm{a}^3)}(a^j, a^1) + d_{\mathrm{minor}(\mathrm{a}^2;\mathrm{a}^1,\mathrm{a}^3)}(a^j, a^1) + d_{\mathrm{minor}(\mathrm{a}^3;\mathrm{a}^1,\mathrm{a}^2)}(a^j, a^1) = 2^{n-1}$, then

$$(2^{n-2} - 1 - x) + (2^{n-2} - y) + (2^{n-2} - z) + (2^{n-2} - t) = 2^{n-1}.$$

As $d(a^j, a^2) = d_{inv(a^1,a^2,a^3)}(a^j, a^2) + d_{\mathrm{minor}(\mathrm{a}^1;\mathrm{a}^2,\mathrm{a}^3)}(a^j, a^2) + d_{\mathrm{minor}(\mathrm{a}^2;\mathrm{a}^1,\mathrm{a}^3)}(a^j, a^2) + d_{\mathrm{minor}(\mathrm{a}^3,\mathrm{a}^1,\mathrm{a}^2)}(a^j, a^2) = 2^{n-1}$, then

$$(2^{n-2} - 1 - x) + y + z + (2^{n-2} - t) = 2^{n-1}.$$

As $d(a^j, a^3) = d_{inv(a^1,a^2,a^3)}(a^j, a^3) + d_{\mathrm{minor}(\mathrm{a}^1;\mathrm{a}^2,\mathrm{a}^3)}(a^j, a^3) + d_{\mathrm{minor}(\mathrm{a}^2;\mathrm{a}^1,\mathrm{a}^3)}(a^j, a^3) + d_{\mathrm{minor}(\mathrm{a}^3,\mathrm{a}^1,\mathrm{a}^2)}(a^j, a^3) = 2^{n-1}$, then

$$(2^{n-2} - 1 - x) + y + (2^{n-2} - z) + t = 2^{n-1}.$$

As $d(a^j, a^z) = d_{inv(a^1,a^2,a^3)}(a^j, a^z) + d_{\mathrm{minor}(\mathrm{a}^1;\mathrm{a}^2,\mathrm{a}^3)}(a^j, a^z) + d_{\mathrm{minor}(\mathrm{a}^2;\mathrm{a}^1,\mathrm{a}^3)}(a^j, a^z) + d_{\mathrm{minor}(\mathrm{a}^3,\mathrm{a}^1,\mathrm{a}^2)}(a^j, a^z) = 2^{n-1}$, then

$$(2^{n-2} - 1 - x) + (2^{n-2} - y) + z + t = 2^{n-1}.$$

In this way, the expression in the Lemma corresponds to the probability of there being a total of $t + (k - t) = k$ differing bits between a^1 and a^{coll}. ■

Proof: (Lemma 4): The expression in the Lemma corresponds to the probability of one, two or three codewords in $\{a^1, a^2, a^3\}$ being at distance k from a^{coll} and the remaining codewords being at a greater distance. ■

Proof: (Lemma 5): The expression in the Lemma corresponds to the probability of one, two or three codewords in $\{a^1, a^2, a^3\}$ being at distance k from a^{coll} and the remaining codewords being at a shorter distance. ■

Proof: (Lemma 6): Lemma 2 says that bits of a^z are identical to bits of a^i in the positions in $minor(a^i; a^j, a^k)$ for $(i, j, k) \in \{(1, 2, 3), (2, 1, 3), (3, 1, 2)\}$. Therefore, the probability of there being k different bits in those $3 \cdot 2^{n-2}$ positions is given by a binomial probability function $b(k; 3 \cdot 2^{n-2}, p)$. ■

Proof: (Lemma 7): According to Lemma 2, a^{coll} and a have 2^{n-3} differing bits in positions in $inv(a^1, a^2, a^3)$. In each $minor(a^i; a^j, a^k)$, for $(i, j, k) \in \{(1, 2, 3), (2, 1, 3), (3, 1, 2)\}$, a^{coll} has all 2^{n-3} bits each of which is different with p and 2^{n-3} bits each of which is different with probability $(1 - p)$. Therefore, we have $3 \cdot 2^{n-3}$ bits with probability p of being different, and thus the probability that t of such bits are different is $b(t; 3 \cdot 2^{n-3}, p)$. On the other hand, we have $3 \cdot 2^{n-3}$ bits with probability $1 - p$ of being different, and thus the probability that $k - t$ of such bits are different is $b(k - t; 3 \cdot 2^{n-3}, 1 - p)$. In this way, the expression in the Lemma computes the probability of there being $t + (k - t) = k$ different bits between a and a^{coll}. ■

Proof: (Lemma 8): The expression in the Lemma computes the probability that at least one out of the $2^n - 3$ codewords in $DH(n) \backslash \{a^1, a^2, a^3\}$ is at distance k of a^{coll}, with the remaining codewords at a longer distance. ■

Proof: (Lemma 11): It can be seen that, if $v = $ "1", bits in "Zone-A" and in "Zone-C" stay undetectable and thus decoding will use Rule 1 and return a value "1."

If $v = $ "0," bits in "Zone-A" and in "Zone-B" stay undetectable. Thus, decoding will use Rule 2 and return a value "0." ■

Proof: (Lemma 12): In a collusion between three codewords $b^1, b^2, b^3 \in SC(d, t)$ with two of them (b^1 and b^2) encoding a value v (without loss of generality, assume $v = 1$ and $\overline{v} = 0$), we have $b^1 = b^2$ with probability $1/t$ and $b^1 \neq b^2$ with probability $1 - 1/t$.

a) In the case $b^1 \neq b^2$, we compute $p_{dif}(v) = 1 - p_{dif}(\overline{v})$ (we suppose d to have an odd value), where $p_{dif}(\overline{v})$ corresponds to the probability of decoding $\overline{v}$ after a collusion based on a p-majority strategy. $p_{dif}(\overline{v})$ is actually the probability of decoding using Rules 2 or 6.

— Rule 2 will be applied if all bits in "Zone-A" and "Zone-B" are "0." Since we are assuming a p-majority strategy, all bits in "Zone-A" will be "0" with probability $b(d; d, 1 - p) = (1 - p)^d$, because the majority bit in these positions is "1." Since $b1 \neq b2$, there will be two detectable blocks in "Zone-B" where the majority bit is "0." Bits in "Zone-B" will be all "0" with probability $b(2d; 2d, p) = p^{2d}$. So the probability of applying Rule 2 is $(1 - p)^d p^{2d}$.

— Since only one out of the three colluding codewords has value "0," it is not possible to have more than one block of "Zone-C" with bit values different from "0." So Rule 4 cannot be applied.

— The next possibility for decoding as "0" is to apply Rule 6. This happens if there are more "0" bits than "1" bits in "Zone-A" and no other rule between 1 and 5 has been applied before. In order to render Rule 3 not applicable, we need one of the two detectable blocks of "Zone-B" to be all zeros. Let us assume it is the leftmost one. This happens with probability $b(d; d, p) = p^d$.

Then we need more than half of the d bits of "Zone-A" with value "0" (or less than one half with value "1"), which happens with probability $\sum_{k=0}^{\lfloor d-1/2 \rfloor} b(k; d, p)$. We also need that one of the two detectable blocks of "Zone-B" is all zeros (which happens with probability p^d) and the other with at least one "1" bit (which causes Rule 2 not to be applied and happens with probability $1 - b(d; d, p) = 1 - p^d$). As this can happen twice, one with each of the blocks of 'Zone-B' forced to have all bits to "0," the total probability is $2 \cdot p^d (1 - p^d) \sum_{k=0}^{\lfloor d-1/2 \rfloor} b(k; d, p)$. The same rule is also applied if both blocks of "Zone-B" have all bits to "0" (with probability p^{2d}) and the number of "1" bits in "Zone-A" is greater than 0 (to make Rule 2 not applicable) and less than than one half of the block length d. The total probability is $p^{2d} \sum_{k=1}^{\lfloor d-1/2 \rfloor} b(k; d, p)$. ■

b) In the case $b^1 = b^2$, the probability of decoding value "1" corresponds to the probability of decoding after applying Rule 1 or Rule 5 (note that Rule 3 is not applicable).

— Rule 1 will be applied if all bits in "Zone-A" are "1" and all bits in "Zone-C" are "0." In both cases, we need all bits to take the majority value, which happens with probability $b(2d; 2d, p) = p^{2d}$.

— The other possibility is to apply Rule 5 conditioned to not having applied Rule 1 before. There are two possible scenarios.

In the first scenario, we need at least one bit of "Zone-C" and more than one half of the bits of "Zone-A" with value "1." This happens with probability $(1 - p^d) \sum_{k=\lfloor d+2/2 \rfloor}^{d} b(k; d, p)$.

In the other scenario, we need all bits of "Zone-C" to be "0" and the number of ones in "Zone-A" to be more than a half of the zone but less than d (otherwise Rule 1 would have been applied before). This happens with probability $p^d \sum_{k=\lfloor d+2/2 \rfloor}^{d-1} b(k; d, p)$. ■

REFERENCES

[1] D. Boneh and J. Shaw, "Collusion-secure fingerprinting for digital data," in *Proc. Advances Cryptology*, vol. LNCS-963, 1995, pp. 452–465.

[2] ——, "Collusion-secure fingerprinting for digital data," *IEEE Trans. Inform. Theory*, vol. IT-44, pp. 1897–1905, 1998.

[3] [Online] Available: http://www.lemuria.org/DeCSS

[4] J. Domingo-Ferrer and J. Herrera-Joancomartí, "Short collusion-secure fingerprints based on dual binary hamming codes," *Electron. Lett.*, vol. 36, no. 20, pp. 1697–1699, 2000.

[5] S. Katzenbeisser and F. A. P. Petitcolas, *Information Hiding: Techniques for Steganography and Digital Watermarking*. Norwood, MA: Artech House, 2000.

[6] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes*. Amsterdam, The Netherlands: North-Holland, 1977.

[7] N. R. Wagner, "Fingerprinting," in *Proc. IEEE Symp. Security Privacy*, Oakland, CA, 1983, pp. 18–22.