

On Random Code-Based Secret Locking Schemes

Mark van Hoeij
Florida State University
Dept. of Mathematics
Tallahassee, FL 32306

Breno de Medeiros
Florida State University
Dept. of Computer Science
Tallahassee, FL 32306

Susanne Wetzel
Stevens Institute of Technology
Dept. of Computer Science
Castle Point on Hudson
Hoboken, NJ 07030

1 Introduction

In [8], Monroe et al. have proposed a novel biometric key encapsulation technique based on secret sharing schemes. Recently, *secret locking* was introduced [7] as the paradigm generalizing the constructions in [8]. In this context, the contributions of this paper are twofold:

Lack of compartmentalization for determinant-based secret locking (DSL). In this paper we show that—contrary to prior belief [7]—the DSL scheme introduced in [8] is not truly compartmented. While this precludes an avenue for proving the security of the scheme, it does not introduce any obvious vulnerabilities or avenues of attack. The result is obtained by showing that the DSL scheme is closely related to a subspace-based construction.

A new family of random code-based (RCSL) constructions. The above discovery about DSL leads to a new schemes, based on the use of random subspace vectors (random codes). This has an interesting side-benefit in that it adds noisy-error tolerance to the distinguishing/non-distinguishing error-tolerance strategy of DSL, addressing an open problem identified in [7]. The new constructions can be tuned to the biometrics of choice, enabling a trade-off between the addition of more non-distinguishing features and the addition of more noisy-error-tolerance to achieve an optimal strategy balancing security (low rate of false-positive in biometrics-based authentication) and redundancy (low false-negative rate).

In the following, we will use the notation and terminology introduced in [8, 7].

2 Determinant-based Secret Locking (DSL) is not truly compartmented

Compartmented access structure. This concept for secret sharing was first introduced by Simmons [9], and has received attention from a number of researchers [2, 4]. In a compartmented secret sharing scheme, each participant P_i is assigned a level $\ell(P_i)$. The same level may be assigned to different users. In order to reconstruct the secret, one share from each level is needed. More formally, the access structure of a compartmented secret sharing scheme is $\Gamma = \{A \in 2^P : A \cap P^j \neq \emptyset\}$, where $P^j = \{P_i \in P : \ell(P_i) = j\}$ is the set of users at level j .

For secret locking we need to consider both compartmented access structures in this traditional sense, as well as relaxations of the notion, where authorized sets may “miss” some compartment levels. Given the level function $\ell(\cdot)$, we define *e-coarsely* compartmented access structures to be those Γ which contain only authorized sets which have empty intersection with at most e different levels:

$$\{A \in 2^P : A \cap P^j \neq \emptyset\} \subseteq \Gamma \subseteq \{A \in 2^P : |\{j; A \cap P^j = \emptyset\}| \leq e\}$$

A compartmented access structure is also trivially a 0-coarsely compartmented access structure.

Determinant-based secret locking [8]. Consider two m by m matrices M_0 and M_1 with entries in the finite field \mathbb{F}_q . The biometric data is translated into a choice function $f : \{1, \dots, m\} \rightarrow \{0, 1\}$. Let M_f denote the m by m matrix whose i 'th row is the i 'th row of matrix $M_{f(i)}$. Thus, the function f indicates for each i whether to select the i 'th row from M_0 or from M_1 . Given the correct function f , the secret in the DSL scheme is the determinant $\det(M_f)$.

Suppose that $\mathcal{DF} \subset \{1, 2, \dots, m\}$ is the set of distinguishing features. Then the matrices M_0 and M_1 are constructed in such a way that if f is the function belonging to the correct biometric data, and g is some other choice function for which $f(i) = g(i)$ for all $i \in \mathcal{DF}$, then $\det(M_f) = \det(M_g)$. In other words, to recover the correct secret it suffices to make the correct choice (the choice between $\text{Row}_i(M_0)$ and $\text{Row}_i(M_1)$) for all $i \in \mathcal{DF}$.

To compute $\det(M_f)$ one must choose precisely one of $\text{Row}_i(M_0)$ or $\text{Row}_i(M_1)$ for each $i \in \mathcal{DF}$. For each $i \notin \mathcal{DF}$ we must also choose $\text{Row}_i(M_0)$ or $\text{Row}_i(M_1)$ but now the choice does not matter. So at first sight it appears that for each i we must use either $\text{Row}_i(M_0)$ or $\text{Row}_i(M_1)$ but not both, which would mean that DSL is a compartmented scheme (see Theorem 1 in [8]).

However, in the following we will show that DSL is not truly compartmented as breaking DSL is equivalent to breaking another (hard) problem given below that is not compartmented. Note, that this does not imply that breaking DSL is easy, only that DSL should not be considered compartmented.

Let us assume that an attacker correctly guesses at least one non-distinguishing feature. Suppose that $i = 1$ is non-distinguishing. Then, selecting either

Row₁(M_0) or Row₁(M_1) would lead to the correct secret $\det(M_f)$, provided that all distinguishing features are chosen correctly. Since Row₁(M_0) and Row₁(M_1) lead to the same determinant, if we choose the difference $v := \text{Row}_1(M_0) - \text{Row}_1(M_1)$ as the first row of M_f , then the $\det(M_f)$ becomes 0 assuming the remaining choices are correct. Thus, we have reduced the original problem to the problem of determining the function $f : \{2, \dots, m\} \rightarrow \{0, 1\}$ such that v and $M_{f(i)}$, $i = 2, \dots, m$ are linearly dependent. If we find such a function, we may compute the span V of v and $M_{f(i)}$, $i = 2, \dots, m$. Conversely, if we have V then determining the function $f : \{2, \dots, m\} \rightarrow \{0, 1\}$ becomes easy as it suffices to select the rows from M_0 or M_1 that are in V . Thus, the problem “compute V ” is equivalent to DSL because given the solution f of DSL we can compute V , and given V we can compute f .

Lemma 1 *Let $F = \mathbb{F}_q$ be a finite field, and M_0, M_1 be two $(m - 1) \times m$ -dimensional matrices with entries in F , resulting from the construction of a DSL instance. Then, the security of this DSL instance is equivalent to the following problem: Find an $m - 1$ -dimensional subspace V of the row space of $\begin{pmatrix} M_0 \\ M_1 \end{pmatrix}$ which contains $2m - 2 - d$ rows, and where d , the number of distinguishing features, satisfies $0 < d < m - 1$. In addition, it is known that if the i -th row of M_0 is in V , then with very high probability, the i -th row of M_1 is NOT in V , and vice-versa.*

To compute V , one must determine $m - 1$ elements of V since V has dimension $< m$. Of the $2m - 2$ remaining vectors Row _{i} (M_0) and Row _{i} (M_1), $i = 2, \dots, m$ there are at least $2m - 2 - d$ vectors in V . In order to compute V one must determine a sufficient number of independent vectors in V . For determining the vectors, it may not be necessary to restrict the choice to only one of the two vectors Row _{i} (M_0) or Row _{i} (M_1) for each i . Instead, one may choose both Row _{i} (M_0) and Row _{i} (M_1) in cases where row i corresponds to a non-distinguishing feature. That is, for such i , both vectors may help us to get closer to finding V . So the problem “compute V ” is not a (strongly) compartmented problem and therefore DSL is not compartmented either. It is important to note that this does not imply that finding of V is easy.

Theorem 1 *The DSL scheme is only d -coarsely compartmented.*

In DSL, V has co-dimension 1. However, if one were to choose some subspace V of higher co-dimension, then this would allow for error-correction. This directly leads to the RCSL scheme which generalizes DSL.

3 Random Code Secret Locking (RCSL)

The determinant-based scheme accommodates errors in the rows corresponding to non-distinguishing features, but cannot easily account for more than very few *noisy* errors (i.e., reading errors for distinguishing features). Since one can

expect to have a small number of noisy errors, currently this deficiency of the determinant-based scheme is overcome using decoding by exhaustive search. However, the cost of such a strategy quickly becomes unwieldy as the number of noisy errors is allowed to increase, as can be extrapolated from the evidence provided in [7].

Assume there are d distinguishing features, and that a biometric reading of the legitimate user is likely to have at most e incorrect, noisy readings among these d features. The new random code secret sharing scheme exhibits the following error-correction capabilities: If the actual number of errors is $\lambda \leq e$, then the errors can be quickly corrected. However, if the actual number of errors is $\lambda > e$ then the errors can be corrected by introducing exhaustive search with exponential cost depending on $\lambda - e$ (instead of λ as in the case of the determinant scheme).

The RCSL scheme works as follows: choose a random subspace V of \mathbf{F}_q^m of dimension $m - e$. V can be interpreted as defining a random linear code. It should not be disclosed, as the key K is an element of \mathbf{F}_q that can be determined quickly from a matrix representation of V in *reduced echelon* form. Such a matrix (and hence the key) can be easily and uniquely reconstructed from any spanning set for V .

An m by 2 table T is first populated with random code words, that is, random elements of V . If q is large enough, with high probability, any subset of $m - e$ randomly selected vectors from T will span V . Then, for each row of T that corresponds to a distinguishing feature, one of the two vectors—the one corresponding to the incorrect reading for the distinguishing feature—will be replaced by a random vector in $\mathbf{F}_q^m \setminus V$.

Suppose that during an authentication attempt one vector from each row in T is selected. Furthermore, let \mathcal{Q} denote the span of the selected vectors. If the selection is correct, then $\mathcal{Q} \subseteq V$. Moreover, if q is large enough, then $\mathcal{Q} = V$ with high probability. If, however, the selection includes λ incorrect choices among the d distinguishing features, then the dimension of \mathcal{Q} will likely be the minimum of $m - e + \lambda$ and m (because if $\lambda \leq e$, each error will result in an extra, independent vector being added to the computed span). In particular, $\mathcal{Q} \neq V$ when $\lambda > 0$ in which case the key cannot be computed from \mathcal{Q} . However, these λ errors can be corrected quickly whenever $\lambda \leq e$, as follows. For each row of the table T , check whether switching the original vector selection results in a decrease in the dimension of \mathcal{Q} . If this is the case, keep the newly selected vector and adjust \mathcal{Q} .

If $\lambda > e$, then it is unlikely that switching the vector selection in any single row will decrease the dimension of \mathcal{Q} , and so the above method no longer works. In this case, correcting errors is done with a combinatorial search. For example, if $\lambda = e + 2$, then try all $\binom{m}{3}$ combinations of three rows, and for each combination, try if simultaneously switching vector selections at those three rows makes the selected vectors linearly dependent. If so, then it is probable that 3 errors have been corrected, after which the remaining $\lambda - 3$ errors are handled as in the previous paragraph.

We note that the method does not work as well for small q : Some spurious

linear relations are to be expected, and as a result the number of errors that can easily be corrected will become slightly less than e .

Moreover, only in the particular case $q = 2$, there is an alternative combinatorial search that is more effective than the “general q ” strategy above, and which implies that one needs to use more dimensions to achieve security. As the alternative combinatorial search strategy is still exponential, the case $q = 2$ will be investigated for its eventual connections to hard problems in coding theory.

References

- [1] X. Boyen. Reusable cryptographic fuzzy extractors. In *Proc. of the 11th ACM Conf. on Comp. and Comm. Secur.*, 2004.
- [2] E. F. Brickell. Some ideal secret sharing schemes. *J. of Comb. Math. and Comb. Computing* 9: pp. 105–112, 1989.
- [3] Y. Dodis, L. Reyzin, and A. Smith. Fuzzy extractors and cryptography, or How to use your fingerprints. In *Proc. of Adv. in Cryptology – EURO-CRYPT’04*, 2004.
- [4] H. Ghodosi, J. Pieprzyk, and R. Safavi-Naini. Secret sharing in multilevel and compartmented groups. *Proc. of the 3rd Australasian Conf. on Info. Secur. and Privacy (ACISP’98)*. LNCS 1438, pp. 367–378, Springer-Verlag, 1998.
- [5] A. Juels and M. Sudan. A fuzzy vault scheme. *Proc. of the 2002 IEEE International Symp. on Inform. Theory*, p. 480, 2002.
- [6] A. Juels and M. Wattenberg. A fuzzy commitment scheme. In *Proc. of the 6th ACM Conf. on Comp. and Comm. Secur.*, pp. 28–36, 1999.
- [7] S. Kamara, B. Medeiros, and S. Wetzel. Secret locking: Exploring new approaches to biometric key encapsulation. To appear in *Proc. of the 2nd Intern. Conf. on e-Business and Telecommunications (ICETE 2005)*, Reading, UK, October 2005.
- [8] F. Monrose, M. K. Reiter, and S. Wetzel. Password hardening based on keystroke dynamics. *Internl. J. of Info. Secur.* 1(2):69–83, 2002.
- [9] G. Simmons. How to (really) share a secret. *Adv. in Cryptology – Proc. of CRYPTO’88*, (S. Goldwasser, ed.), LNCS 403, pp. 390–448, Springer-Verlag, 1990.