

# Privacy for Mobile Code

## Position Paper

**Sergio Loureiro<sup>1</sup>, Refik Molva**

Institut Eurecom, Sophia Antipolis - France

{loureiro, molva}@eurecom.fr

**Abstract:** This position paper discusses the problem of evaluating a function on an untrusted host, while maintaining the confidentiality of the function. A new non-interactive protocol designed to evaluate a function on an untrusted host is presented. The protocol prevents the disclosure of the function under cryptographic assumptions.

**Keywords:** Mobile code protection, privacy of computations, malicious hosts.

## 1 Introduction

With the advent of new computing paradigms like mobile code and ubiquitous computing, the privacy and integrity of software programs become a major concern beyond classical data security considerations. Running a program in a potentially hostile environment may raise various security requirements, as follows:

- a company might need to prevent the disclosure of certain sensitive algorithms implemented in its software products despite extensive code analysis and reverse engineering by potential intruders including its customers;
- a mobile software agent acting on behalf of a person might need to assure the integrity of some critical operation performed on an untrusted remote host;
- a data collection agent might need to assure both the confidentiality and the integrity of the results computed at various competing sites.

In this position paper, we suggest a cryptographic mechanism for evaluating a function on an untrusted environment while assuring the privacy of the function. The goal of function privacy is twofold:

- algorithm confidentiality, i. e., hiding the design of the algorithm;
- integrity of execution, i. e., if an attacker cannot derive the algorithm, then he cannot figure out the best way of tampering it to his benefit.

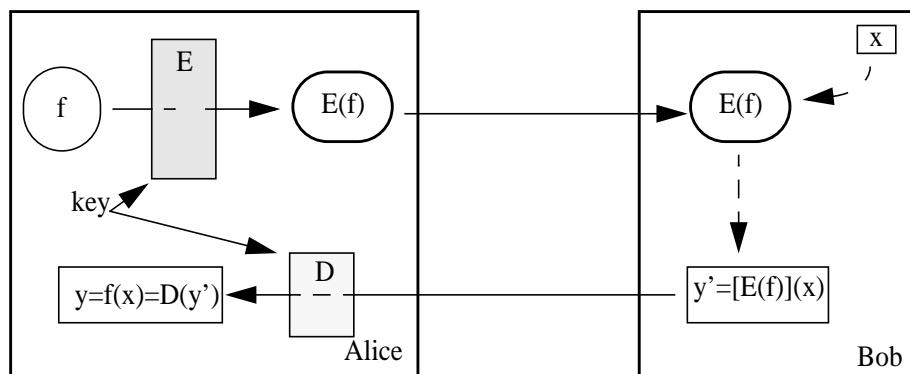
The position paper is organized as follows: in section two, the existing approaches to function evaluation with confidentiality are referred and a definition of autonomous protocol is given. Section three defines the intractability assumption of coding theory used as a guarantee of security of our protocol. In section four, a simple protocol to achieve privacy for mobile code is given and its security is evaluated. Section five is dedicated to future work and conclusions.

---

1. Supported by the Portuguese FCT grant: PRAXIS XXI/BD/13875/97

## 2 Related Work

The problem addressed here was referred in the seminal paper by Abadi, Feigenbaum and Kilian [AFK89], which focuses on hiding data from an oracle, or in other words, computing with encrypted data. Based on this idea, Abadi and Feigenbaum [AF90] developed a protocol to secure circuit evaluation, which allows a player to evaluate his data on another player's boolean circuit, thereby preserving the confidentiality of his data, under the Quadratic Residue Assumption (QRA), and also hiding the circuit from the owner of the data. Even though originally intended for data confidentiality, this protocol can also be used for encrypting functions. The major drawback of the protocol is the communication complexity between the two players.



**FIGURE 1. Autonomous Protocol**

Recently, Sander and Tschudin [ST98b], [ST98a] defined a function hiding scheme based on an autonomous (non-interactive) protocol as depicted in Figure 1. This protocol is autonomous in so far as the interactions between the owner of the function (Alice) and the remote party that evaluates the function (Bob), consist only of the transmission of the function by Alice to Bob and the transmission of the result back to Alice by Bob. Unlike the protocol by Abadi and Feigenbaum [AF90], an autonomous protocol does not involve the exchange of information between players during function evaluation.

In an autonomous protocol, a function  $f$  owned by Alice is evaluated by Bob on the input data  $x$  (provided by Bob), while preventing the disclosure of  $f$  to Bob. The privacy of  $f$  is assured by the transformation  $E$  that satisfies the following properties:

- it is infeasible under the intractability of a computational problem to derive  $f$  from  $E(f)$  without the knowledge of a secret trapdoor;
- the cleartext result  $f(x)$  can be derived from the encrypted result  $[E(f)](x)$  in polynomial time using a secret trapdoor (function  $D$ ).

Sander and Tschudin [ST98b] illustrated the autonomous protocol concept with a method that allows to encrypt polynomials, based on the Goldwasser Micali [GM84] encryption scheme. When the functions to be evaluated can be expressed in terms of polynomials, function hiding is achieved, under the QRA.

The possibility of using the so-called composition techniques is also referred in [ST98b], but no security evaluation is provided. The composition techniques consist in multiplying function  $f$  by a random invertible function.

The goal of the protocols for function evaluation with privacy is conceptually different from the protocols used for Private Information Retrieval (see for example [CMS99] for one of the latest results on the field

and a survey of previous work), where the goal is to hide an index  $i$ , while retrieving the bit  $a_i$  from a public database in the form of a string  $A=a_1a_2\dots a_i\dots a_n$ , therefore preserving the privacy of the query.

In [Hoh98] the author presented a technique, called black box, designed to render code interpretation more complex. The work focuses on the Java programming language and performs obfuscation of the bytecode in order to render reverse engineering more complex, especially when automatic disassemblers are used. For example, variables are split into different arrays and their names changed. However, the security of such empiric techniques is difficult to quantify.

We excluded from this section all the protocols using several players or data replication between different non-communicating databases. We will suggest an original autonomous protocol based on an intractability assumption of coding theory. Therefore, a brief overview of the computational complexity assumption beyond certain coding problems is given.

### 3 Coding Theory

The idea presented in this position paper consists of encrypting a function represented on a matrix format, with a transformation similar to the one used to construct the public key on Public Key Cryptosystems based on coding theory. Cryptosystems based on coding theory rely on the difficulty of decoding or finding a minimum weight codeword in a large linear code with no visible structure. These general problems of coding theory were proven to be NP complete [EBvT78] and were used on the public key cryptosystems proposed by McEliece [McE78], Niederreiter [Nie86] and Gabidulin [GPT91]. Some identification schemes that exploit these problems have also been proposed in [Ste93] and [Ver95].

Despite the general problem of finding a minimum weight codeword in a large linear code with no visible structure being NP-complete, the best known attacks exploit the properties of linear codes to find a trapdoor, i. e., to recover the structure of the original code or to find an equivalent code. This attack is usually called a Brickell-like attack [Bri84].

The security of the cryptosystem is highly dependent on the class of codes used. The initial proposal from Niederreiter used concatenated codes, which were proven to be insecure [Sen94]. Reed-Solomon codes were also proven to be insecure [SS92]. McEliece proposed Goppa codes that proved to be secure. Nevertheless, Goppa codes generated by a Goppa polynomial which has binary coefficients are also insecure [Loi98]. A description of the cryptanalysis of the McEliece scheme is beyond the scope of this position paper and can be found on [LM99].

The properties that a code should have in order to be an eligible candidate for these cryptosystems, which result from the lessons learned from successful attacks against this kind of cryptosystems, are the following [CC98]:

- The class of codes must be large enough to avoid any enumeration;
- An efficient decoding algorithm should exist for this class;
- The generator or parity-check matrix of a transformation of the code must not give any information about its structure.

If the codes obey these rules then the security of the cryptosystems is equivalent to the problem of decoding any linear code without any visible structure. The class of Goppa codes meets all the properties referred. There are a big number of different Goppa codes, efficient decoding algorithms exist and it does not exist an efficient algorithm to retrieve the characteristic parameters of the code from a permuted generation matrix [CS98]. Therefore, we use the problem of decoding as an intractable assumption.

## 4 Function Evaluation with Privacy - FEP

Figure 2 depicts the operations performed by the two players of the autonomous protocol using the proposed function privacy scheme as described below. The description of our scheme is done for binary codes, like on the original McEliece scheme, but can be extended to q-ary linear codes which were also proven to be secure [JM96]. Nevertheless, the binary matrix format is suitable for representing boolean functions or circuits.

Let  $G$  be a generating matrix for an  $[n,k,d]$  Goppa code  $C$ . Let  $P$  be an  $n \times n$  random permutation matrix and  $E$  an  $l \times n$  random matrix where at least  $(n-t)$  columns consist of the null vector.  $G$ ,  $P$ , and  $E$  are kept secret by Alice. Let  $F$  be a  $l \times k$  matrix over  $\mathbb{Z}_2$  representing function  $f$ . Alice computes the encrypted function  $F'$  by  $F' = FGP + E$  and sends  $F'$  to Bob. Bob evaluates  $F'$  on his data  $x \in (\mathbb{Z}_2)^t$  expressed by the multiplication  $y' = xF'$  and sends back the result  $y'$  to Alice.

Alice decrypts the result  $y_1 = y'P^{-1}$  and uses  $C$ 's secret decoding algorithm [MS77] to retrieve the clear-text result  $y = xF$  from  $y_1 = xFG + xEP^{-1}$ . The vector  $xEP^{-1}$  is a correctable error vector since its Hamming weight  $w(xEP^{-1})$  is inferior at  $t$ .

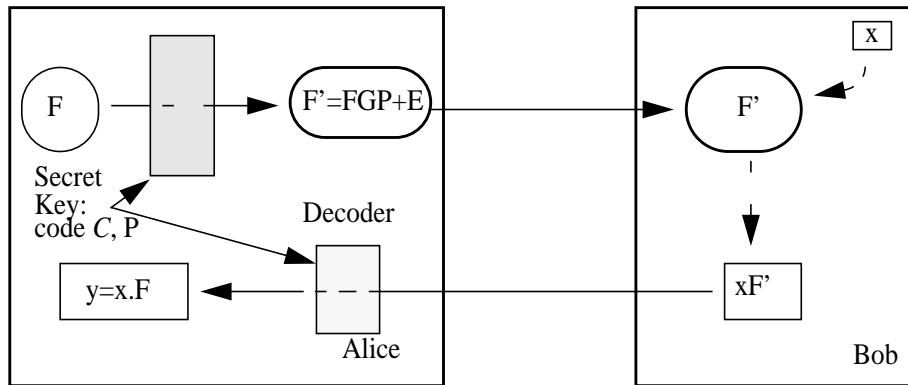


FIGURE 2. Autonomous Protocol based on coding theory.

### 4.1 Example

In this example, we show how to use the protocol described for remote evaluation of a ridiculously small boolean circuit with 2 inputs and 4 outputs. Each output can be expressed by an equation:

$$y_i = f_{3i} \cdot x_1 \cdot x_2 + f_{2i} \cdot x_2 + f_{1i} \cdot x_1 + f_{0i}, 0 \leq i \leq 4$$

Therefore, the circuit can be represented by an  $4 \times 4$  matrix, and the operations are performed over  $\mathbb{Z}_2$ . The circuit evaluation can be done by the following vector by matrix multiplication:

$$\begin{bmatrix} 1 & x_1 & x_2 & x_1 x_2 \end{bmatrix} \cdot \begin{bmatrix} f_{00} & f_{01} & f_{02} & f_{03} \\ f_{10} & f_{11} & f_{12} & f_{13} \\ f_{20} & f_{21} & f_{22} & f_{23} \\ f_{30} & f_{31} & f_{32} & f_{33} \end{bmatrix} = \begin{bmatrix} y_0 & y_1 & y_2 & y_3 \end{bmatrix}$$

Generally, in a boolean circuit with  $l$  inputs and  $k$  outputs, each equation has  $m = 2^l$  terms corresponding to all possible combinations between inputs. Thus  $m$  is the size of the input vector  $x$  and the matrix  $F$  representing the circuit has size  $m \times k$ . This matrix will be transformed on a matrix  $F'$  of size  $m \times n$ .

The example highlights a disadvantage of our protocol: the expansion of the matrix expressing the function. For the code [1024, 524, 101] initially proposed by McEliece, the size of the circuit will be almost duplicated. Nevertheless, this disadvantage also happens to a higher degree with the other autonomous protocol previously referred [ST98b].

## 4.2 Security Evaluation

The function privacy property relies on the hardness of retrieving the private function  $F$  from the encrypted function  $F'$ . The matrix  $F$  does not change the codewords of the code, that is, it does not influence the security properties of the code described on section three. On the other hand, each row of  $F'$  is a codeword of an unstructured code, so under our intractability assumption, it is infeasible for Bob to retrieve each row of  $F$  individually.

The error matrix  $E$  used in our scheme enhances the security of the function hiding, in particular against matrix factorization attacks. The use of matrix  $E$  as a randomizer is an important security advantage to our scheme over the composition techniques based on the multiplication by random matrices.

If the function  $f$  is invertible, Alice can always find the input data  $x$  from  $f(x)$  and  $f$ . The confidentiality of the input data  $x$  with respect to a third party intruder, during transmission, can be assured if Bob adds a correctible random error vector to the result of the computation. Then, the total number of errors, that is the sum of the errors in the matrix  $E$  and the ones in the vector  $xF'$ , cannot exceed the error correcting capability of the code. If the result  $f(x)$  is transmitted back to Bob, there is the possibility of an interpolation attack, which hardness relies on the complexity of the function  $f$ .

## 5 Conclusion and Future Work

This position paper presented an original approach to the problem of function evaluation with privacy, using an intractability assumption of coding theory.

The aim of our protocol was to address the issue of secure evaluation of functions in potentially hostile environments. Even though the basic purpose of our scheme is privacy, the privacy of the function can also assure the integrity of its execution. If an attacker cannot disclose the original function, and if the final result is encrypted, he will not be able to tamper the function in his benefit.

Future work will focus on more efficient representations for boolean functions and the extension of our protocol to a broader class of languages.

## 6 References

- [AF90] Martin Abadi and Joan Feigenbaum. Secure circuit evaluation. *Journal of Cryptology*, 2(1):1–12, 1990.
- [AFK89] Martin Abadi, Joan Feigenbaum, and Joe Kilian. On hiding information from an oracle. *Journal of Computer and System Sciences*, 39(1):21–50, August 1989.
- [Bri84] Ernest F. Brickell. Breaking iterated knapsacks. In G. R. Blakley and David Chaum, editors, *Advances in Cryptology: Proceedings of CRYPTO 84*, volume 196 of *Lecture Notes in Computer Science*, pages 342–358. Springer-Verlag, 1985, 19–22 August 1984.
- [CC98] A. Canteaut and F. Chabaud. A new algorithm for finding minimum-weight words in a linear code: application to McEliece’s cryptosystem and to narrow-sense BCH codes of length 511. *IEEE Transactions on Information Theory*, 1998.
- [CMS99] Christian Cachin, Silvio Micali, and Markus Stadler. Computationally private information retrieval

with polylogarithmic communication. In Jacques Stern, editor, *Advances in Cryptology: EUROCRYPT '99*, Lecture Notes in Computer Science. Springer, 1999.

- [CS98] A. Canteaut and N. Sendrier. Cryptanalysis of the original McEliece cryptosystem. In *In Advances in Cryptology - ASIACRYPT'98, Lecture Notes in Computer Science*. Springer-Verlag, 1998.
- [EBvT78] R. McEliece E. Berlekamp and H. van Tilborg. On the inherent intractability of certain coding problems. *IEEE Trans. Information Theory*, IT-24(3):384–386, May 1978.
- [GM84] Shafi Goldwasser and Silvio Micali. Probabilistic encryption. *Journal of Computer and System Sciences*, 28(2):270–299, April 1984.
- [GPT91] E. M. Gabidulin, A. V. Paramonov, and O. V. Tretjakov. Ideals over a non-commutative ring and their application in cryptography. In D. W. Davies, editor, *Advances in Cryptology—EUROCRYPT 91*, volume 547 of *Lecture Notes in Computer Science*, pages 482–489. Springer-Verlag, 8–11 April 1991.
- [Hoh98] Fritz Hohl. An approach to solve the problem of malicious hosts. In *4th ECOOP Workshop on Mobility: Secure Internet Mobile Computations*, 1998.
- [JM96] Heeralal Janwa and Oscar Moreno. McEliece public key cryptosystems using algebraic-geometric codes. *Designs, Codes and Cryptography*, 8(3):293–307, June 1996.
- [LM99] Sergio Loureiro and Refik Molva. Function hiding based on error correcting codes. In C. H. Lee Manuel Blum, editor, *Proceedings of Cryptec'99 - International Workshop on Cryptographic techniques and Electronic Commerce*. City University of Hong-Kong, July 1999.
- [Loi98] P. Loidreau. Some weak keys in McEliece public-key cryptosystem. In *IEEE International Symposium on Information Theory, ISIT'98, Boston, USA*, 1998.
- [McE78] R. McEliece. A public-key cryptosystem based on algebraic coding theory. In *Jet Propulsion Lab. DSN Progress Report*, 1978.
- [MS77] F. MacWilliams and N. Sloane. *The Theory of Error-Correcting Codes*. North-Holland, 1977.
- [Nie86] H. Niederreiter. Knapsack-type cryptosystems and algebraic coding theory. In *Probl. Contr. and Information Theory*, 1986.
- [Sen94] Nicolas Sendrier. On the structure of a randomly permuted concatenated code. In *EUROCODE 94*, pages 169–173, Abbaye de la Bussiere sur Ouche, France, October 1994.
- [SS92] V. Sidelnikov and S. Shestakov. On cryptosystems based on generalized Reed-Solomon codes. *Discret. Mat.*, 4:57–63, 1992.
- [ST98a] Tomas Sander and Christian Tschudin. On software protection via function hiding. In *Proceeding of the Second Workshop on Information Hiding*, Portland, Oregon, USA, April 1998.
- [ST98b] Tomas Sander and Christian Tschudin. Towards mobile cryptography. In *Proceeding of the 1998 IEEE Symposium on Security and Privacy*, Oakland, California, May 1998.
- [Ste93] Jacques Stern. A new identification scheme based on syndrome decoding. In Douglas R. Stinson, editor, *Advances in Cryptology—CRYPTO '93*, volume 773 of *Lecture Notes in Computer Science*, pages 13–21. Springer-Verlag, 22–26 August 1993.
- [Ver95] P. Veron. A fast identification scheme. In *IEEE Symposium on Information Theory*, Canada, 1995.