Distributed Encryption Systems

Adrian ATANASIU Victor MITRANA

University of Bucharest, Faculty of Mathematics
Str. Academiei 14, 70109, Bucharest, Romania
email: *adrian@matem.buc.soros.ro*, *mitrana@moisil.math.ro*

Abstract. This paper debates upon the distributed encryption by using formal methods from language theory. After some basic properties of distributed encryption systems were presented, we have pointed out possible applications in cryptography by constructing an authentication system based on hidden channels.

1 Introduction

So far, a few results and methods in formal language theory have been used for the security of conveyance messages. Along these lines, there were investigated some public-key cryptosystems based on language theory (iterated morphisms and substitution, hiding regular languages, etc.) A survey and more details can be found in [7] and the references thereof.

However, our opinion is that the central concept in hiding messages, that of replacing some sequences by another ones, is very common in language theory. To substitute some subwords of a word with other strings in the aim of hiding the original message is one of the well-known techniques in cryptography. A substitution can be viewed as a production of the form $x \to y$ where the words x, y are given; to apply such a substitution to a given text w all non-overlapped occurrences of x are simultaneously replaced by y. Thus, different texts are obtained, according to different decompositions of w with respect to x [4]. The present paper continues a series of papers [3], [4], [1], [2] dealing with cryptosystems based on substitution. Here we study the way of encrypting by distributed systems and we discuss on the possibility to construct an authentication system based on hidden channels [8].

The topic seems to be very generous and attractive; in this paper we exploit just one result presented at the end of the third section by developing an authentication method. Nevertheless, the reader may construct other methods whose implementation requires simple devices as finite automata or generalized sequential machines.

2 Preliminaries

An alphabet is a finite nonempty set; if $V = \{a_1, a_2, \dots, a_n\}$ is an alphabet, then any sequence $w = a_{i_1}a_{i_2}\dots a_{i_k}, 1 \leq i_j \leq n, 1 \leq j \leq k$, is called word (string) over V. The length of the aforementioned word w is denoted by $|w|$ and equals k. The empty word is denoted by $e, |e| = 0$. The set of all words over V is denoted by V^* and $V^+ = V^* - \{e\}$.

For two words x, y we denote by $N_x(y)$ the number of occurrences of x in y, that is

$$N_x(y) = card(\{\alpha | y = \alpha x \beta\}),$$

and extend this notation to a finite nonempty set A

$$N_A(y) = \sum_{x \in A} N_x(y).$$

Note that we count all different occurrences of x, including the overlappings. Let V be an alphabet and $P \subset V^* \times V^*$ be a finite nonempty set of *encryption rules* written in the form $x \rightarrow y,\ x, y \in V^*$. Denote by $dom(P)$ the set $\{x | x \rightarrow y \in P\}$ and $P^{-1} = \{y \rightarrow x | x \rightarrow y \in P\}$.

For $w \in V^*$, the encryption of w by means of P is the set

$$w(P) = \{z_0 y_1 z_1 y_2 \dots z_{n-1} y_n z_n | \text{ for some } n \geq 1\},$$

where

$$\begin{array}{rl} (i) & w = z_0 x_1 z_1 x_2 \dots z_{n-1} x_n z_n, \\ (ii) & x_i \rightarrow y_i \in P,\ 1 \leq i \leq n, \\ (iii) & N_{dom(P)}(z_j) = 0,\ \text{for any } 0 \leq j \leq n. \end{array}$$

Furthermore, for a set $A \subset V^*$

$$\begin{array}{rcl} A(P) & = & \bigcup_{w \in A} w(P), \\ \emptyset(P) & = & \emptyset. \end{array}$$

A *distributed encryption scheme* of degree $n \geq 1$ is a structure

$$\gamma = (V, P_1, P_2, .., P_n),$$

where V is an alphabet and $P_i \subset (V^* \times V^*)$ are finite and nonempty sets of encryption rules, for all $1 \leq i \leq n$. Let π be a permutation of the set $\{1, 2, \ldots, n\}$ and $w \in V^+$ be a string over V. We define

$$CT_\gamma(w, \pi) = w(P_{\pi(1)})(P_{\pi(2)}) \ldots (P_{\pi(n)})$$

the set of all cryptotexts (the cryptoset) of the plain text w, by means of γ, based upon the permutation π. Moreover, S_n is the set of all permutations of $\{1, 2, \ldots, n\}$.

Example 1 Consider $V = \{a, b, c, d\}, P_1 = \{ab \rightarrow cd, b \rightarrow a, aa \rightarrow d\}, P_2 = \{ac \rightarrow c, ab \rightarrow a\}$ and $w = abab$. There are only two permutations of $\{1, 2\}$; the cryptosets based on the two permutations are:

$$\begin{array}{rcl} abab(P_1)(P_2) & = & \{cdcd, aacd, cdaa, aaaa\}(P_2) = \{acd\} \\ abab(P_2)(P_1) & = & \{aa\}(P_1) = \{d\}. \end{array}$$

A *distributed encryption system* of degree n is the structure

$$\Gamma = (V, P_1, P_2, \ldots, P_n, A),$$

where $\gamma = (V, P_1, P_2, \ldots, P_n)$ is a distributed encryption scheme of degree n and $A \subseteq V^*$ is the plain (initial) language. The *cryptolanguage* defined by Γ and the permutation $\pi \in S_n$ is

$$CL_\Gamma(\pi) = \bigcup_{x \in A} CT_\gamma(x, \pi).$$

In what follows, we restrict our investigation to encryption systems whose plain language is either finite or regular.

3 Basic properties of distributed encryption systems

Theorem 1 *Let $\Gamma = (V, P_1, P_2, \ldots, P_n, A)$ be a distributed encryption system with a finite/regular plain language A. For each permutation $\pi \in S_n$ there exist the morphisms h_1, h_2 and the finite/regular language R such that*

$$CL_\Gamma(\pi) = h_2(h_1^{-1}(h_2(h_1^{-1}(\ldots h_1^{-1}(h_2(h_1^{-1}(A) \cap R)) \cap R)) \cap \ldots \cap R)).$$

Proof. We consider the new alphabets $V^{(i)},\ 1 \leq i \leq n$, where

$$V^{(i)} = \{a^{(i)} | a \in V\}.$$

Furthermore, $x^{(k)} = a_1^{(k)} a_2^{(k)} \ldots a_q^{(k)}$ provided that $x = a_1 a_2 \ldots a_q,\ a_i \in V,\ 1 \leq i \leq q$.

Assume that

$$P_j = \{x_{j,t} \rightarrow y_{j,t} | 1 \leq t \leq r_j\}.$$

Let $c_{i,1}, c_{i,2} \dots, c_{i,r_i}$ new symbols, for all $1 \leq i \leq n$. Define the morphisms

$$h_1 : \bigcup_{i=1}^{n} V^{(i)} \cup \bigcup_{i=1}^{n} \{c_{i,1}, c_{i,2} \dots, c_{i,r_i}\} \longrightarrow V \cup \bigcup_{i=1}^{n-1} V^{(i)},$$

defined as follows

$$\begin{array}{rcl} h_1(a^{(1)}) & = & a,\ a \in V, \\ h_1(a^{(i)}) & = & a^{(i-1)},\ a \in V,\ 2 \leq i \leq n, \\ h_1(c_{\pi(1),j}) & = & x_{\pi(1),j},\ 1 \leq j \leq r_{\pi(1)}, \\ h_1(c_{\pi(i),j}) & = & x^{(i-1)}_{\pi(i),j},\ 2 \leq i \leq n,\ 1 \leq j \leq r_{\pi(i)}, \end{array}$$

and

$$h_2 : \bigcup_{i=1}^{n} V^{(i)} \cup \bigcup_{i=1}^{n} \{c_{i,1}, c_{i,2} \dots, c_{i,r_i}\} \longrightarrow V \cup \bigcup_{i=1}^{n-1} V^{(i)},$$

with

$$\begin{array}{rcl} h_2(a^{(i)}) & = & a^{(i)},\ a \in V,\ 1 \leq i \leq n-1, \\ h_2(a^{(n)}) & = & a,\ a \in V, \\ h_2(c_{\pi(i),j}) & = & y^{(i)}_{\pi(i),j},\ 1 \leq i \leq n-1,\ 1 \leq j \leq r_{\pi(i)}, \\ h_2(c_{\pi(n),j}) & = & y_{\pi(n),j},\ 1 \leq j \leq r_{\pi(n)}. \end{array}$$

If A is a finite set, put

$$\begin{array}{rcl} p & = & max\{|x| \ : \ x \in A\} \\ q & = & max\{|x| \ : \ x \in \bigcup_{i=1}^{n} dom(P_i^{-1})\}. \end{array}$$

The language R required by the theorem is

$$R = \begin{cases} \bigcup_{i=1}^{n}(((V^{(i)})^* \setminus (V^{(i)})^* dom^{(i)}(P_{\pi(i)})(V^{(i)})^*)\{c_{\pi(i),1}, c_{\pi(i),2}, \dots, c_{\pi(i),r_{\pi(i)}}\})^*, \\ \text{ if } A \text{ is a regular language} \\ \\ \bigcup_{i=1}^{n}((((V^{(i)})^* \setminus (V^{(i)})^* dom^{(i)}(P_{\pi(i)})(V^{(i)})^*)\{c_{\pi(i),1}, c_{\pi(i),2}, \dots, c_{\pi(i),r_{\pi(i)}}\})^* \cap \\ \cap (V^{(i)})^{pq^i}),\ \text{if } A \text{ is a finite set,} \end{cases}$$

where $V^{(k)}$ delivers the set of all strings of length at most k. It is easy to notice that R is finite/regular provided that A is finite/regular. Let w be an arbitrary string in A. All strings in $h_1^{-1}(w)$ are obtained from w by replacing some of its substrings $x_{\pi(1),j}$

with $c_{\pi(1),j}$ and all the remained symbols with their associated symbols "coloured" by the "colour 1". The intersection with the language R controls this process, namely it removes all strings which contain "coloured" substrings $x^{(1)}_{\pi(1),j}$, for some $1 \leq j \leq r_{\pi(1)}$. Thus, $h_2(h_1^{-1}(w) \cap R)$ contains exactly the "coloured" copies of the strings in $w(P_{\pi(1)})$. An inductive reasoning completes the proof. □

Due to the closure properties of the family of regular sets and to the previous theorem one may state:

Corollary 1 *If* Γ *is a distributed encryption system of degree* n *whose plain language is regular,* $CL_\Gamma(\pi)$ *is a regular set, for all permutations* $\pi \in S_n$.

Returning to the aim of introducing the distributed encryption systems, we focus our attention to an important feature that may be required for encryption systems, that of *reversibility.*

A distributed encryption system $\Gamma = (V, P_1, P_2, \ldots, P_n, A)$ is *weakly reversible* if exists a permutation $\pi \in S_n$ such that

$$A \subseteq A(P_{\pi(1)})(P_{\pi(2)}) \ldots (P_{\pi(n)})(P^{-1}_{\pi(n)}) \ldots (P^{-1}_{\pi(1)}).$$

The same encryption system is *strongly reversible* if exists a permutation $\pi \in S_n$ such that

$$A = A(P_{\pi(1)})(P_{\pi(2)}) \ldots (P_{\pi(n)})(P^{-1}_{\pi(n)}) \ldots (P^{-1}_{\pi(1)}).$$

A complete characterizations of reversible distributed encryption systems seems very difficult to be reached. However, both properties are decidable for every distributed encryption system, whatever plain language it has.

Theorem 2 *One can algorithmically decide whether or not a given distributed encryption system is weakly or strongly reversible.*

Proof. The problem is trivialy decidable for systems having a finite plain language. In the case of systems with regular plain languages, the statement follows immediately from the decidability status of the inclusion and equivalence problems for regular languages. Note that the crypto-language defined by a distributed encryption system with a regular plain languages can effectively be constructed. □

Theorem 3 *The distributed encryption system* $\Gamma = (V, P_1, \ldots, P_n, A)$ *is weakly reversible if exists a permutation* $\pi \in S_n$ *such that for each* $0 \leq i \leq n-1$

$$A_i \subseteq A_i(P_{\pi(i+1)})(P^{-1}_{\pi(i+1)}),$$

where $A_0 = A$ *and* $A_i = A(P_{\pi(1)}) \ldots (P_{\pi(i)})$.

Proof. Without loss of generality one may assume that the permutation π is the identical one. We proceed by induction on n. For $n = 1$ the assertion is immediately true from the definition of weakly reversability. Assume the assertion true for all distributed encryption systems of degree k and let $\Gamma = (V, P_1, P_2, \ldots P_{k+1}, A)$ be an encryption system of degree $k + 1$. By the induction hypothesis one may infer that

$$A \subseteq A_k(P_k^{-1}) \ldots (P_1^{-1}).$$

As $A_k \subseteq A_k(P_{k+1})(P_{k+1}^{-1})$, we may write

$$A \subseteq A_k(P_k^{-1}) \ldots (P_1^{-1}) \subseteq A_k(P_{k+1})(P_{k+1}^{-1})(P_k^{-1}) \ldots (P_1^{-1}) \subseteq$$
$$\subseteq A(P_1)(P_2) \ldots (P_{k+1})(P_{k+1}^{-1})(P_k^{-1}) \ldots (P_1^{-1}),$$

which ends the proof. □

A similar sufficient condition holds for strongly reversible distributed encryption systems as well. Unfortunately, this condition, despite that it is just sufficient, does not appear to be essentialy simpler than the definition. On the other hand, if we restrict the investigation to encryption systems of degree one with one encryption rule only (parallel substitution [4]), the reversibility of the system implies a very special plain language ([5]). By increasing either the degree of the system or the number of encryption rules, in order to preserve the reversability property, we have to limit strictly the form of the initial language.

These considerations led us to consider a different way of decryption the crypto-language. To this end, we consider that the legal receiver knows the distributed encryption scheme of the sender and has to figure out the maximal language (with respect to inclusion) that leads to the received language by applying the encryption scheme to it. Note that this problem is completely different of the reversibility problem. The problem will be treated in more detail in the next section.

Obviously, the first question that naturally arises is: Can the receiver algorithmically determine the maximal language? Clearly, the answer is affirmativ if the crypto-language is finite. The case of regular crypto-languages is settled by the next theorem

Theorem 4 *Let $\gamma = (V, P_1, P_2, \ldots, P_n)$ be a distributed encryption scheme, R be a regular set and π be a permutation in S_n. The maximal language A such that $CL_\Gamma(\pi) = R$, $\Gamma = (\gamma, A)$, can algorithmically be constructed.*

Proof. For the set of encryption rules P over the alphabet V and the string $w \in V^*$ denote by

$$\begin{aligned} w < P > \;=\; & \{w_0y_1w_1y_2 \ldots y_kw_k | w = w_0x_1w_1x_2 \ldots x_kw_k, \text{ for some } k \geq 1, \\ & x_i \rightarrow y_i \in P, 1 \leq i \leq n, \text{ and } N_{dom(P^{-1})}(w_j) = 0, 0 \leq j \leq k\}. \end{aligned}$$

The algorithm we are going to present is based on the following two facts:

1. If $y \in x < P >$, then $x \in y(P^{-1})$.
2. If $x \in y(P)$, then $y \in x < P^{-1} >$.

Again, for sake of simplicity we assume that the permutation π is the identical permutation.

Algorithm 1

Input: The encryption scheme γ and the regular set R;

Output: The maximal plain language A.

```
begin
$A := R$;
for  $i := n$ downto  1 do
   begin
   $E := A < P_i^{-1} >$;
   $F := E(P_i)$;
   if  $F = A$ then  $A := E$
      else
      begin
      $Q := F \setminus A$;
      $M := Q < P_i^{-1} >$;
      $F' := E \setminus M$;
      if  $A = F'(P_i)$ then  $A = F'$
         else  "THE PROBLEM HAS NO SOLUTION"; stop
      end;
   end;
end.
```

Owing to the two facts mentioned at the beginning of this proof it follows that at each step i the algorithm computes, if exists, the maximal set L such that $R = CL_{\Gamma'}(\varepsilon')$, where

(i) $\Gamma' = (V, P'_1, P'_2, \ldots, P'_{n-i+1}, L)$,

(ii) $P'_j = P_{i+j-1},\ 1 \leq j \leq n-i+1$,

(iii) ε' is the identical permutation of $\{1, 2, \ldots, n-i+1\}$.

Indeed, at each step i, $F'(P_i) \subseteq A$ holds, and F' is maximal with this property.

The above remarks prove the correctness of the algorithm provided that all its instructions can be effectively performed. For a regular language R and a set of encryption rules P both sets $R < P >$ and $R(P)$ are regular sets which can be effectively constructed. $R(P)$ can be constructed following the proof of Theorem 1 whereas $R < P >$ is the image of a gsm mapping applied to R. The reader can easily construct such a gsm. Moreover, the difference of two effective regular languages is still an effectiv regular language. For more details, we refer to [6]. □

4 Authentication based on distributed encryption systems

Theorem 4 can be used as starting point for building an authentication system as we are going to do in the following. Let $\gamma = (V, P_1, P_2, \ldots, P_n)$ be a distributed encryption scheme and R be a regular language. As we have seen in Theorem 4, one may figure out a maximal set A such that $CT_\gamma(A, \pi) = R$, for some $\pi \in S_n$. Now, we proceed as follows:

- We choose an alphabet V' with $V \subset V'$.
- We enrich the sets P_i up to P_i', for all $1 \leq i \leq n$.
- Let $k \geq 0$ be an arbitrary integer. We define the permutation $\pi' \in S_{n+k}$ that extends, in some sense, the permutation π. Thus, if $\pi'(i) = \pi(j)$, $\pi'(l) = \pi(s)$ and $j < s$, then $i < l$ holds.
- We construct the new sets of encryption rules $P_{n+1}', \ldots, P_{n+k}'$ satisfying the properties

  – if $\pi'(i) \leq n$, then $A_i := A_{i-1}(P_{\pi'(i)}')$;
  – if $\pi'(i) > n$, then $A_i := A_{i-1}(P_{\pi'(i)}') = A_{i-1}$;

  for all $1 \leq i \leq n+k$ and $A_0 := A$.
- Let R' be an infinite regular language that includes R (usually, R is finite). Let A' be the maximal set computed by the algorithm from Theorem 4 starting with $\gamma' = (V', P_1', P_2', \ldots, P_{n+k}')$, R' and π'. Obviously, $A \subset A'$.

The distributed encryption system $\Gamma' = (V', P_1', P_2', \ldots, P_{n+k}', A')$ and the permutation π' are public whilst $\Gamma = (V, P_1, P_2, \ldots, P_n, A)$ and π are secret. Everybody may encrypt a message in A' and send the cryptotext to a receiver. The receiver checks whether the received text is in R (this can be algorithmically done because R is regular). In accordance with the answer, the receiver either rejects the message or accepts and decodes it within the system Γ.

Example 2 Let $\gamma = (V, P_1, P_2)$, where

$$V = \{a, b, c\}, \quad P_1 = \{a \to cb, b \to a\}, \quad P_2 = \{aba \to ab, bab \to ba\}.$$

If $\pi = (2, 1)$ and $R = \{cbaa, acba, acbcb\}$, then the maximal language obtained by Algorithm 1, starting from R and γ is

$$A = \{abab, baba, babb\}.$$

Following the above considerations, we take:

$$\begin{array}{rcl} V' & = & \{a, b, c, d\},\ k = 1,\ \pi' = (2, 3, 1), \\ P'_1 & = & P_1 \cup \{bc \to bc\}, \\ P'_2 & = & P_2 \cup \{c \to bcb, ab \to bcb, ba \to adb, d \to da\}, \\ P'_3 & = & \{abb \to aba, aba \to abb, bab \to baa, baa \to bab, adb \to acb\}. \end{array}$$

The choice of a certain permutation does not introduce any ambiguity in the system due to the following remarks. A set of words A is uniquely encrypted by the distributed encryption scheme $\gamma = (V, P_1, P_2, \ldots, P_n)$ if exists a permutation $\pi \in S_n$ such that the next two conditions hold:

$$\begin{array}{ll} 1. & CT_\gamma(A, \pi) \neq \emptyset, \\ 2. & CT_\gamma(A, \sigma) = \emptyset,\ \text{for all } \sigma \in S_n \setminus \{\pi\}. \end{array}$$

Theorem 5 *Let $\gamma = (V, P_1, P_2, \ldots, P_n)$ be a distributed encryption scheme, A is a set of words over V such that $CT_\gamma(A, \varepsilon) \neq \emptyset$. Then, there is a distributed encryption scheme γ' such that A is uniquely encrypted by γ'.*

Proof. One defines the new alphabets $V_i = \{a_i | a \in V\},\ 1 \leq i \leq n$, where $V_i \cap V = V_i \cap V_j = \emptyset,\ i \neq j$. Then, one defines the encryption scheme

$$\gamma' = (V', P'_1, P'_2, \ldots, P'_{2n+1}),$$

where

$$\begin{array}{rcl} V' & = & V \cup V_1 \cup V_2 \cup \ldots \cup V_n, \\ P'_1 & = & \{a \to a_1 | a \in V\},\ P'_{2n+1} = \{a_n \to a | a \in V\}, \\ P'_{2k} & = & \{h_k(x) \to h_k(y) | x \to y\},\ 1 \leq k \leq n, \\ P'_{2k+1} & = & \{a_k \to a_{k+1} | a \in V\},\ 1 \leq k \leq n-1. \end{array}$$

The morphisms h_k, $1 \leq k \leq n$, are defined from V^* into V_k^* by $h_k(a) = a_k$, for all $a \in V$. It is easy to notice that each set P'_{2k} is applicable only between P'_{2k-1} and P'_{2k+1}; consequently A is uniquely encrypted by γ. □

References

[1] A. Atanasiu, Substitution with words and languages: Application to Cryptography. In *Mathematical aspects of natural and formal languages* (Gh. Păun ed.), World Scientific, 43, 1994, 1–12.

[2] A. Atanasiu, About encryption using formal methods: Substitution with words and languages, *Ann. Univ. Buc.*, XLLII–XLIII, 1994, 68–75.

[3] A. Atanasiu, V. Mitrana - Substitution on words and languages. In *Developments in Language Theory. At the Crossroads of Mathematics, Computer Science and Biology* (A. Salomaa, G. Rozenberg, eds.), World Scientific Publishing, 1994, 51–60.

[4] A. Atanasiu, V. Mitrana, Parallel substitution on words and languages. *Proc. of the* 9^{th} *ROSYCS'93, Iasi* (V. Felea, G. Ciobanu, eds.), 1993, 24–30.

[5] S. La Tore, M. Napoli, D. Parente, Parallel word substitution, *Fundamenta Informaticae* 27, 1(1996), 27–36.

[6] M. A. Harrison - *Introduction to Formal Language Theory*, Addison-Wesley, Reading Mass., 1978.

[7] A. Salomaa, *Public-key Cryptography*, Springer-Verlag, Heidelberg 1990.

[8] J. Seberry, J. Pieprzyk, *Cryptography: An Introduction to Computer Security.* Advanced in Computer Sciences series, Prentice Hall, 1989.