Some attacks of an encryption system based on the word problem in a monoid

Nacer Ghadbane et Douadi Mihoubi
nasserghedbane@yahoo.com, mihoubi_douadi@yahoo.fr
Laboratory of Pure and Applied Mathematics , Department
of Mathematics, University of M'sila, Algeria

Abstract: In this work, we are interested in **ATS-monoid** protocol (proposed by **P. J. Abisha, D. G. Thomas G. and K. Subramanian**, the idea of this protocol is to transform a system of **Thue** $S_1 = (\Sigma, R)$ for which the word problem is undecidable a system of **Thue** $S_2 = (\Delta, R_\theta)$ or $\theta \subseteq \Delta \times \Delta$ for which the word problem is decidable in linear time. Specifically, it gives attacks against ATS monoid in spésifiques case and thenme examples of these cases.

Keywords: Free monoid, Thue system, Morphism monoids, The closure of a binary relation, The word problem in a monoid, Public Key Cryptography.

MSC(2010): 94A60, 68Q42, 20M05.

1. Preliminaries

A monoid is a set M together with an associative product $x, y \longmapsto xy$ and a unit 1. If $X \subset M$, we write X^* for the submonoid of M generated by X, that is the set of finite products $x_1x_2...x_n$ with $x_1, x_2, ..., x_n \in X$, including the empty product 1. It is the smallest submonoid of M containing X.

An alphabet is a finite nonempty set. The elements of an alphabet Σ are called letters or symbols. Aword over an alphabet Σ is a finite string consisting of zero or more letters of Σ, whereby the same letter may occur several times. The string consisting of zero letters is called the empty word, written ϵ. Thus, $\epsilon, 0, 1, 011, 1111$ are words over the alphabet $\{0, 1\}$. The set of all words over an alphabet Σ is denoted by Σ^*. the set Σ^* is infinite for any Σ. Algebraically, Σ^* is the free monoid generated by Σ. If u and v are words over an alphabet Σ, then so is their catenation uv. Catenation is an associative operation, and the empty word is an identity with respect to catenation: $u\epsilon = \epsilon u = u$ holds for all words u. For a word u and a natural number i, the notation u^i means the word obtained by catenating i copies of the word u. By definition, u^0 is the empty word ϵ. The length of a word u, in symbols $|u|$, is the number of letters in u when each letter is counted as many times as it occurs. Again by definition, $|\epsilon| = 0$. The length function possesses some of the formal properties of logarithm:

$$|uv| = |u| + |v|, |u^i| = i|u|,$$

for any words u and v and integers $i \geq 0$. For example $|011| = 3$ and $|1111| = 4$.

Let $f : S \longrightarrow U$ be a mapping of sets.

- We say that f is **one-to-one** if for every $a, b \in S$ where $f(a) = f(b)$, we have $a = b$.
- We say that f is **onto** if for every $y \in U$, there exists $a \in S$ such that $f(a) = y$.

A mapping $h : \Sigma^* \longrightarrow \Delta^*$, where Σ and Δ are alphabets, satisfying the condition

$$h(uv) = h(u)h(v), \text{ for all words } u \text{ and } v,$$

is called a morphism, define a morphism h, it suffices to list all the words $h(\sigma)$, where a ranges over all the (finitely many) letters of Σ. If M is a monoid, then any mapping $f : \Sigma \longrightarrow M$ extends to a unique morphism $\widetilde{f} : \Sigma^* \longrightarrow M$. For instance, if M is the additive monoid $\mathbb{N}$, and f is defined by $f(\sigma) = 1$ for each $\sigma \in \Sigma$, then $\widetilde{f}(u)$ is the length $|u|$ of the word u.

Let $h : \Sigma^* \longrightarrow \Delta^*$ be a morphism of monoids. if h is **one-to-one** and **onto,** then h is an **isomorphism** and the monoids Σ^* and Δ^* are **isomorphic**. we denote $Hom(\Sigma^*, \Delta^*)$ the set of morphisms from Σ^* to Δ^* and $Isom(\Sigma^*, \Delta^*)$ the set of isomorphisms from Σ^* to Δ^*. We say that $h \in Hom(\Sigma^*, \Delta^*)$ is non trivial if there exists $\sigma \in \Sigma$ such that $h(\sigma) \neq \epsilon$.

A binary reation on Σ^* is a subset $R \subseteq \Sigma^* \times \Sigma^*$. If $(x, y) \in R$, we say that x is related to y by R, denoted xRy. The inverse relation of R is the binary reation $R^{-1} \subseteq \Sigma^* \times \Sigma^*$ defined by $yR^{-1}x \Longleftrightarrow (x, y) \in R$.

The relation $I_{\Sigma^*} = \{(x, x), x \in \Sigma^*\}$ is called the identity relation. The relation $(\Sigma^*)^2$ is called the complete relation.

Let $R \subseteq \Sigma^* \times \Sigma^*$ and $S \subseteq \Sigma^* \times \Sigma^*$ binary relations. The composition of R and S is a binary relation $S \circ R \subseteq \Sigma^* \times \Sigma^*$ defined by

$$x (S \circ R) z \Longleftrightarrow \exists y \in \Sigma^* \text{ such that } xRy \text{ and } ySz.$$

A binary relation R on a set Σ^* is said to be

- reflexive if xRx for all x in Σ^*;
- symmetric if xRy implies yRx;
- transitive if xRy and yRz imply xRz.

The relation R is called an equivalence relation if it is reflexive, symmetric, and transitive. And in this case, if xRy, we say that x and y are equivalent.

Let R be a relation on a set Σ^*. The reflexive closure of R is the smallest reflexive relation $r(R)$ on Σ^* that contains R; that is,

- $R \subseteq r(R)$
- if R' is a reflexive relation on Σ^* and $R \subseteq R'$, then $r(R) \subseteq R'$.

The symmetric closure of R is the smallest symmetric relation $s(R)$ on Σ^* that contains R; that is,

- $R \subseteq s(R)$
- if R' is a symmetric relation on Σ^* and $R \subseteq R'$, then $s(R) \subseteq R'$.

The transitive closure of R is the smallest transitive relation $t(R)$ on Σ^* that contains R; that is,

- $R \subseteq t(R)$
- if R' is a transitive relation on Σ^* and $R \subseteq R'$, then $t(R) \subseteq R'$.

Let R be a relation on a set Σ^*. Then

- $r(R) = R \cup I_{\Sigma^*}$,
- $s(R) = R \cup R^{-1}$
- $t(R) = \bigcup_{k=1}^{k=+\infty} R^k$.

A congruence on a monoid M is an equivalence relation $\equiv$ on M compatible with the operation of M, i.e, for all $m, m' \in M, u, v \in M$

$m \equiv m' \Longrightarrow umv \equiv um'v$

A **Thue** system is a pair (Σ, R) where Σ is an alphabet and R is a non-empty finite binary on Σ^*, we write $urv \rightarrow_R ur'v$ whenever $u, v \in \Sigma^*$ and $(r, r') \in R$. We write $u \rightarrow_R^* v$ if there words $u_0, u_1, ..., u_n \in \Sigma^*$ such that,

$$\begin{array}{c} u_0 = u, \\ u_i \longrightarrow_R u_{i+1}, \forall 0 \leq i \leq n-1 \\ \text{and } u_n = v. \end{array}$$

If $n = o$, we get $u = v$, and if $n = 1$, we get $u \rightarrow_R v$. $\rightarrow_R^*$ is the reflexive transitive closure of $\rightarrow_R$.

The congruence generated by R is defined as follows:

- $urv \longleftrightarrow_R ur'v$ whenever $u, v \in \Sigma^*$, and rRr' or $r'Rr$;
- $u \longleftrightarrow_R^* v$ whenever $u = u_0 \longleftrightarrow_R u_1 \longleftrightarrow_R ... \longleftrightarrow_R u_n = v$.

$\longleftrightarrow_R^*$ is the reflexive symmetric transitive closure of $\rightarrow_R$. Let $\pi_R : \Sigma^* \longrightarrow \Sigma^*/\longleftrightarrow_R^*$ be the canonical surjective monoid morphism that maps a word $w \in \Sigma^*$ to its equivalence class with respect to $\longleftrightarrow_R^*$. A monoid M is finitely generated if it is ithenmorphic to a monoid of the form $\Sigma^*/\longleftrightarrow_R^*$. In this case, we also say that M is finitely generated by Σ. If in addition to Σ also R is finite, then M is a finitely presented monoid. The word problem of $M \simeq \Sigma^*/\longleftrightarrow_R^*$ with respect to R is the set $\{(u, v) \in \Sigma^* \times \Sigma^* : \pi_R(u) = \pi_R(v)\}$ it is undecidable in general [8, 13]. In some cases, the word problem can be much easier.

Indeed, for $\theta \subseteq \Sigma \times \Sigma$, we say that:

$$u, v \in \Sigma^* \text{ are equivalence with respect to } \theta \text{, if and only if, } u \longleftrightarrow_{R_\theta}^* v,$$

where $\longleftrightarrow_{R_\theta}^*$ is the reflexive symmetric transitive closure of $\longrightarrow_{R_\theta}$, with $R_\theta = \{(ab, ba) : (a, b) \in \theta\}$.

In the **Thue** system $S = (\Sigma, R_\theta)$, **R. V. Book** and **H. N. Liu** showed [16] that the word problem is decidable in linear time. This is mainly based on the following theorem **R. Cori** and **D. Perrin**[3].

Let $u, v \in \Sigma^*, \theta \subseteq \Sigma \times \Sigma$ and a sub alphabet $\Delta \subseteq \Sigma$. we define, $P_\Delta : \Sigma^* \longrightarrow \Delta^*$ by:

$$\begin{cases} P_\Delta(\sigma) = \sigma, & \text{if } \sigma \in \Delta, \text{ and} \\ P_\Delta(\sigma) = \epsilon, & \text{if } \sigma \notin \Delta. \end{cases}$$

Then:

$$u \longleftrightarrow_{R_\theta}^* v \Longleftrightarrow \begin{cases} P_{\{\sigma\}}(u) = P_{\{\sigma\}}(v), \text{ for everything } \sigma \text{ of } \Sigma \text{ and} \\ P_{\{\sigma,\mu\}}(u) = P_{\{\sigma,\mu\}}(v), \text{ for everything } (\sigma, \mu) \notin \theta \end{cases}$$

Public-Key cryptography, also called asymmetric cryptography, was invented by **Diffie** And **Hellman** more than forty years ago. In Public-Key cryptography, a user U has a pair of related keys (pK, sK): the key pK is public and should be available to everyone, while the key sK must be kept secret by U. The fact that sK is kept secret by a single entity creates an asymmetry, hence the name asymmetric cryptography.

A one-way function f is a function that maps a domain into range sush that every function value has a unique inverse, with the condition that the calculation of the function is easy whereas the calculation of the inverse is infeasible:

$$\begin{array}{rc} y = f(x) & \text{easy} \\ x = f^{-1}(y) & \text{infeasible} \end{array}$$

Trapdoor one-way functions are a family of invertible functions f_k such that $y = f_k(x)$ is easy if k and x known, and $x = f_k^{-1}(y)$ is infeasible if y is known but k is not known. The devlopment of a partical Public-Key scheme depends on the discovery of a suitable trapdoor one-way function.

2. The ATS-monoid protocol

P. J. Abisha, **D. G. Thomas** and **K. G. Subramanian**, use the theorem of **R. Cori** and **D. Perrin**. To build the ATS-monoid protocol,the idea is transform a system of **Thue** $S_1 = (\Sigma, R)$ for which the word problem is undecidable in a **Thue** system $S_2 = (\Delta, R_\theta)$ with $\theta \subseteq \Delta \times \Delta$ and $R_\theta = \{(ab, ba) : (a, b) \in \theta\}$ for which the word problem is decidable in linear time.

Public-Key (pK)**:** A **Thue** system $S_1 = (\Sigma, R)$ and two words w_0, w_1 of Σ^*. (Σ, R, w_0, w_1) constitute a public-key.

Secret-key (sK)**:** A **Thue** system $S_2 = (\Delta, R_\theta)$ where Δ alphabet of size smaller than Σ, a morphism h from Σ^* to Δ^*, such that for all $(r, s) \in R$:

$$\begin{cases} (h(r), h(s)) \in \{(ab, ba), (ba, ab)\}, \text{ for a pair } (a, b) \in \theta, \text{ or} \\ h(r) = h(s). \end{cases}$$

Therefore:

$$\text{for all } u, v \in \Sigma^*, u \longleftrightarrow_R^* v \Longrightarrow h(u) \longleftrightarrow_{R_\theta}^* h(v).$$

thus if $h(u)$ and $h(v)$ are not equivalent with respect to $\longleftrightarrow_{R_\theta}^*$, then u and v are not equivalent with respect to $\longleftrightarrow_R^*$.

And, we also we have two words x_0, x_1 of Δ^* such that $x_0 \longleftrightarrow_{R_\theta}^* h(w_0), x_1 \longleftrightarrow_{R_\theta}^* h(w_1)$ with $h(w_0)$ and $h(w_1)$ are not equivalent with respect to $\longleftrightarrow_{R_\theta}^*$. $(\Delta, R_\theta, h \in Hom(\Sigma^*, \Delta^*))$ constitute a secret-key.

Encryption: for encrypt a bit $b \in \{0, 1\}$, **Bob** chooses a word c of Σ^* in the equivalence class of w_b with respect to $\longleftrightarrow_R^*$, i. e, $c \in [w_b]_{\longleftrightarrow_R^*}$ where $[w_b]_{\longleftrightarrow_R^*}$ denotes the equivalence class of w_b with respect to $\longleftrightarrow_R^*$ and then sent to **Alice**.

Decryption: Upon receipt of a word c of Σ^*, **Alice** calculated $h(c) \in \Delta^*$, since $c \longleftrightarrow_R^* w_b$ and according to the result for all $u, v \in \Sigma^*, u \longleftrightarrow_R^* v \Longrightarrow h(u) \longleftrightarrow_{R_\theta}^* h(v)$ we have $h(c) \longleftrightarrow_{R_\theta}^* h(w_b)$, for example if $h(c) \longleftrightarrow_{R_\theta}^* x_0$ the message is decrypted 0.

Example :
Public-Key (pK)**:**
$\Sigma = \{\sigma_1, \sigma_2, \sigma_3, \sigma_4\}$,
$R = \{(\sigma_2\sigma_3, \sigma_3\sigma_2), (\sigma_2\sigma_4, \sigma_4\sigma_2), (\sigma_1\sigma_3, \sigma_3\sigma_1)\}$,
$w_0 = \sigma_1\sigma_2\sigma_4\sigma_3\sigma_1\sigma_2\sigma_3\sigma_4$,
$w_1 = \sigma_2\sigma_4\sigma_3\sigma_4\sigma_2\sigma_1$.
Secret-key (sK):
$\Delta = \{a, b, c\}, \theta = \{(a, b), (a, c)\}$ and $h : \Sigma^* \longrightarrow \Delta^*$ is defined by :

$$h(\sigma_1) = \epsilon, h(\sigma_2) = a, h(\sigma_3) = b, h(\sigma_4) = c.$$

We have $R_\theta = \{(ab, ba), (ac, ca)\}$, $h(w_0) = x_0 = acbabc$ and $h(w_1) = x_1 = acbca$.
Now we verify the following conditions :
1. $h(w_0)$ et $h(w_0)$ are not equivalent with respect to $\longleftrightarrow^*_{R_\theta}$.
2. for all $(r, s) \in R$:

$$\begin{cases} (h(r), h(s)) \in \{(ab, ba), (ba, ab)\}, \text{ for a pair } (a, b) \in \theta, \text{ or} \\ h(r) = h(s). \end{cases}.$$

For condition 1. Just use the theorem of **R. Cori** and **D. Perrin**, we have $P_{\{b\}}(h(w_0)) = P_{\{b\}}(acbabc) = bb$ and $P_{\{b\}}(h(w_1)) = P_{\{b\}}(acbca) = b$, then $h(w_0)$ and $h(w_1)$ are not equivalent with respect to $\longleftrightarrow^*_{R_\theta}$.
For condition 2. we have $R = \{(\sigma_2\sigma_3, \sigma_3\sigma_2), (\sigma_2\sigma_4, \sigma_4\sigma_2), (\sigma_1\sigma_3, \sigma_3\sigma_1)\}$ then $(h(\sigma_2\sigma_3), h(\sigma_3\sigma_2)) = (ab, ba) \in R_\theta$, $(h(\sigma_2\sigma_4), h(\sigma_4\sigma_2)) = (ac, ca) \in R_\theta$, $(h(\sigma_1\sigma_3), h(\sigma_3\sigma_1)) = (b, b)$ (we have $h(\sigma_1\sigma_3) = h(\sigma_3\sigma_1)$.
Therefore:

$$\text{for all } u, v \in \Sigma^*, u \longleftrightarrow^*_R v \Longrightarrow h(u) \longleftrightarrow^*_{R_\theta} h(v).$$

Encryption: for example, for encrypt the 0, **Bob** chooses a word c of $\{\sigma_1, \sigma_2, \sigma_3, \sigma_4\}^*$ in the equivalence class of w_0 with respect to $\longleftrightarrow^*_R$, i. e, $c \in [w_0]_{\longleftrightarrow^*_R}$ where $[w_0]_{\longleftrightarrow^*_R}$ denotes the equivalence class of w_0 with respect to $\longleftrightarrow^*_R$, and then sent to **Alice**.
we have $w_0 = \sigma_1\sigma_2\sigma_4\sigma_3\sigma_1\sigma_2\sigma_3\sigma_4 \longleftrightarrow^*_R \sigma_1\sigma_4\sigma_2\sigma_3\sigma_1\sigma_2\sigma_3\sigma_4 \longleftrightarrow^*_R \sigma_1\sigma_4\sigma_3\sigma_2\sigma_1\sigma_2\sigma_3\sigma_4$.
We choose $c = \sigma_1\sigma_4\sigma_3\sigma_2\sigma_1\sigma_2\sigma_3\sigma_4$.
Decryption: Upon receipt of a word c of $\{\sigma_1, \sigma_2, \sigma_3, \sigma_4\}^*$,
Alice calculated $h(c) = h(\sigma_1\sigma_4\sigma_3\sigma_2\sigma_1\sigma_2\sigma_3\sigma_4) = cbaabc \in \{a, b, c\}^*$, Now using the theorem of **R. Cori** and **D. Perrin**, such that $h(c) \longleftrightarrow^*_{R_\theta} h(w_0)$. we have $P_{\{a\}}(h(c)) = P_{\{a\}}(h(w_0)) = aa$, $P_{\{b\}}(h(c)) = P_{\{b\}}(h(w_0)) = bb$, $P_{\{c\}}(h(c)) = P_{\{c\}}(h(w_0)) = cc$.
then for all σ of $\{a, b, c\}$, $P_{\{\sigma\}}(h(c)) = P_{\{\sigma\}}(h(w_0))$. In addition it is verified that $P_{\{\sigma,\mu\}}(h(c)) = P_{\{\sigma,\mu\}}(h(w_0))$, for all $(\sigma, \mu) \notin \theta$, we have the complementary of θ is $C_{\Delta\times\Delta}\theta = \{(a, a), (b, a), (b, b), (b, c), (c, a), (c, b), (c, c)\}$,
then $P_{\{b,c\}}(h(c)) = P_{\{b,c\}}(h(w_0)) = cbbc$. Finally $h(c) \longleftrightarrow^*_{R_\theta} h(w_0) = x_0$ and the word is decrypted 0.

3. Security of ATS-monoid protocol

An attack against **ATS-monoid** does not allow to find exactly the **Secret-key**. We will get rather a key that is equivalent to it in the following direction:

We say that $(\Delta', R_{\theta'}, h' \in H(\Sigma^*, \Delta'^*))$ is an equivalent key to the **Secret-key** $(\Delta, R_\theta, h \in Hom(\Sigma^*, \Delta^*$ if any message encrypted with the **Public-Key** (Σ, R, w_0, w_1) can be decrypted with $(\Delta', R_{\theta'}, h' \in Hom(\Sigma^*, \Delta'^*))$. This is the case for example if $(\Delta', R_{\theta'}, h' \in Hom(\Sigma^*, \Delta'^*))$ checks the following three conditions:

1. h' is non trivial and $|\Delta'| \leq |\Sigma|$.
2. $\forall (r,s) \in R,\ \begin{array}{c}(h'(r), h'(s)) \in \{(ab, ba), (ba, ab)\}, \text{ for a pair } (a,b) \in \theta', \text{ or} \\ h'(r) = h'(s).\end{array}$
3. $h'(w_0)$ et $h'(w_0)$ are not equivalent with respect to $\longleftrightarrow^*_{R_{\theta'}}$.

Now we recall some keys that are equivalent to the **Secret-key** $(\Delta, R_\theta, h \in Hom(\Sigma^*, \Delta^*))$.

1. if $h(\Sigma) = \{h(\sigma), \sigma \in \Sigma\}$ and $\theta' = \theta \cap h(\Sigma) \times h(\Sigma)$. then: $(h(\Sigma), R_{\theta'}, h \in Hom(\Sigma^*, \Delta^*))$ is an equivalent key to the **Secret-key** $(\Delta, R_\theta, h \in Hom(\Sigma^*, \Delta^*))$.
2. if $|\Delta'| = |\Delta|$, $i \in Iso(\Delta^*, \Delta'^*)$ and $i(\theta) = \{(i(a), i(b)), (a,b) \in \theta\}$. then $(\Delta', R_{i(\theta)}, i \circ h \in Hom(\Sigma^*, \Delta$ is an equivalent key to the **Secret-key** $(\Delta, R_\theta, h \in Hom(\Sigma^*, \Delta^*))$.

Now describe a general attack against the **ATS-monoid** protocol. In the first time we notice that a key $(\Delta', R_{\theta'}, h' \in Hom(\Sigma^*, \Delta'^*))$ equivalent to the **Secret-key** $(\Delta, R_\theta, h \in Hom(\Sigma^*, \Delta^*))$ is independent of alphabet Δ,the only thing that matters is the size of Δ. On the other hand, we observe that the relation $R_{\theta'}$ is easily deduced from the knowledge of $h' \in Hom(\Sigma^*, \Delta'^*)$. Then for a **Public-Key** (Σ, R, w_0, w_1) there is a algorithm noted by **Algo-ATS-monoid** which returns an equivalent key to the **Secret-key** $(\Delta, R_\theta, h \in Hom(\Sigma^*, \Delta^*))$ to complexity $|R| \sum_{i=1}^{i=k} (i+1)^{|\Sigma|}$, with $k = |\Delta|$.

Alg**orithm – ATS – monoid**
Data : (Σ, R, w_0, w_1), **Public – Key** (pK) *of* **ATS – monoid** *proto*col.
Re**sult** : $(\Delta_i, R_{\theta_i}, h_i \in Hom(\Sigma^*, \Delta_i^*))$, *equivalent key to the* **Secret – key**.
While $i, 1 \leq i \leq |\Sigma|$ **Do**
  Δ_i *is any alphabet of i lettres*
  While $h_i \in Hom(\Sigma^*, \Delta_i^*)$ **Do**
   $\theta_i \longleftarrow \emptyset$
   While $(r,s) \in R$ **Do**
    Calculate $h_i(r)$ *and* $h_i(s)$
     If $h_i(r) \neq h_i(s)$ **Then**
      If $h_i(r) = ab$ *and* $h_i(s) = ba$, *for* $a, b \in \Delta_i$ **Then**
        If $(a,b) \notin \theta_i$ *and* $(b,a) \notin \theta_i$ *then* $\theta_i \longleftarrow \theta_i \cup \{(a,b)\}$
      If *no Choose another morphism, i.e.* **Return** *to the second loop* **While**
     End If
   End while
   If $h_i(w_0)$ *and* $h_i(w_1)$ *are not equivalent modulo* $\longleftrightarrow^*_{R_{\theta_i}}$ **Then**
   Return $(\Delta_i, R_{\theta_i}, h_i \in H(\Sigma^*, \Delta_i^*))$
  End While
End while

4. Some attacks against ATS-monoid

In this section we give some attacks against **ATS-monoid** that is to say in each case we return an equivalent key to the **secret-key** of this protocol.

Corollary 4.1

Let (Σ, R, w_0, w_1) be a **Public-Key** of **ATS-monoid** protocol**.**

If $\forall (r,s) \in R, |r| = |s|$, then $(\Delta_1 = \{a\}, R_\theta = \emptyset, h_1 \in Hom(\Sigma^*, \Delta_1^*))$ where for all $\sigma \in \Sigma, h_1(\sigma) = a$, is an equivalent key to the **Secret-key**.

Proof

The key $(\Delta_1 = \{a\}, R_\theta = \emptyset, h_1 \in Hom(\Sigma^*, \Delta_1^*))$ where for all $\sigma \in \Sigma, h_1(\sigma) = a$, checked the following three conditions:

1. the morphism h_1 is not trivial because for all $\sigma \in \Sigma, h_1(\sigma) = a \neq \epsilon$.

2. $\forall (r,s) \in R, h_1(r) = h_1(s) = (a)^{|r|} = (a)^{|s|}$.

3. if $R_\theta = \emptyset$, then $\longleftrightarrow^*_{R_\theta} = I_{\Sigma^*}$ consequently $h_1(w_0)$ and $h_1(w_1)$ are not equivalent modulo $\longleftrightarrow^*_{R_\theta}$ since $h_1(w_0) \neq h_1(w_1)$. then $(\Delta_1 = \{a\}, R_\theta = \emptyset, h_1 \in Hom(\Sigma^*, \Delta_1^*))$ is an equivalent key to the **Secret-key**.

Corollary 4.2

Let (Σ, R, w_0, w_1) be a **Public-Key** of **ATS-monoid** protocol**.**

S'il existe $(r,s) \in R, |r| \neq |s|$, then $(\Delta_1 = \{a\}, R_\theta = \emptyset, h_1 \in Hom(\Sigma^*, \Delta_1^*))$ where $h_1(\Sigma) = \{a, \epsilon\}$ is an equivalent key to the **Secret-key**.

Example 4.3

Public-Key:

$\Sigma = \{\sigma_1, \sigma_2, \sigma_3, \sigma_4, \sigma_5\}$,

$R = \{(\sigma_1\sigma_3, \sigma_3\sigma_1), (\sigma_1\sigma_4, \sigma_4\sigma_1), (\sigma_2\sigma_3, \sigma_3\sigma_2), (\sigma_2\sigma_4, \sigma_4\sigma_2), (\sigma_5\sigma_3\sigma_1, \sigma_3\sigma_5)\}$,

$w_0 = \sigma_4\sigma_2\sigma_4\sigma_3\sigma_4\sigma_2\sigma_3\sigma_4, w_1 = \sigma_2\sigma_4\sigma_3\sigma_4\sigma_2\sigma_1$.

The key $(\Delta_1 = \{a\}, R_\theta = \emptyset, h_1 \in Hom(\Sigma^*, \Delta_1^*))$ or $h_1(\sigma_1) = h_1(\sigma_3) = \epsilon, h_1(\sigma_2) = h_1(\sigma_4) = h_1(\sigma_5) = a$ is verified the following conditions:

1. the morphism h_1 is non trivial.

2. $\forall (r,s) \in R, h_1(r) = h_1(s)$.

3. we have $h_1(w_0) = a^6$ et $h_1(w_1) = a^4$ and like $\longleftrightarrow^*_{R_\theta} = I_{\Sigma^*}$, then $h_1(w_0)$ and $h_1(w_1)$ are not equivalent with respect to $\longleftrightarrow^*_{R_\theta}$.

. then $(\Delta_1 = \{a\}, R_\theta = \emptyset, h_1 \in Hom(\Sigma^*, \Delta_1^*))$ is an equivalent key to the **Secret-key**.

Corollary 4.4

Let (Σ, R, w_0, w_1) be a **Public-Key** of **ATS-monoid** protocol**.**

if there exists σ_k of the alphabet Σ such that for all $(r,s) \in R, |r|_{\sigma_k} = |s|_{\sigma_k} = 0$, then

$(\Delta_1 = \{a\}, R_\theta = \emptyset, h_1 \in Hom(\Sigma^*, \Delta_1^*))$ or for all $\sigma \in \Sigma$ with $\sigma \neq \sigma_k, h_1(\sigma) = \epsilon$ and $h_1(\sigma_k) = a$, is an equivalent key to the **Secret-key.**

Proof

The key $(\Delta_1 = \{a\}, R_\theta = \emptyset, h_1 \in Hom(\Sigma^*, \Delta_1^*))$ is checked three conditions:

1. the morphism h_1 is non trivial. because $h_1(\sigma_k) = a \neq \epsilon$.

2. $\forall (r,s) \in R, h_1(r) = h_1(s) = \epsilon$.

3. if $R_\theta = \emptyset$, then $\longleftrightarrow^*_{R_\theta} = I_{\Sigma^*}$, so it must verify that $h_1(w_0) \neq h_1(w_1)$.

REFERENCES

1. A. Ali, P et N. Hadj. S, "La Cryptographie et ses Principaux Systèmes de Références," RIST, n^o 4, (2002).
2. M. Benois, "Application de l'étude de certaines congruences à un problème de décidabilité," Séminaire Dubreil , n^o 7, (1972).
3. R. Cori et D. Perrin, "Automates et Commutations Partielles," RAIRO-Informatique théorique, tome19, n^o 1, p.21-32, (1985).
4. W. Diffie, M. E. Hellman, "New Direction in Cryptography," IEEE Trans, on Inform Theory, 22(6), P. 644-665, (1976).
5. M. Eytan, G. TH, "Guilbaud. Présentation de quelques monoïdes finis. Mathématiques et sciences humaines," tome 7, p. 3-10, (1964).
6. R. Floyd, R. Beigel, "Traduction de D. Krob. Le langage des machines," International Thomthenn France, Paris, (1995).
7. Y. Lafont, "Réécritue et problème du mot," Gazette des Mathématiciens, Laboratoire de Mathématiques Discrètes de Luminy, Marseille, France, (2009).
8. A. Markov, "On the impossibility of certain algorithme in the theory of asthenciative systems,", Doklady Akademi Nauk SSSR,55, 58:587-590, 353-356, (1947).
9. Y. Metiver, "Calcul de longueurs de chaines de réécriture dans un monoïde libre," U.E.R. de Mathématiques et informatique, Université de Bordeaux 1, France (1983).
10. M. Nivat, "Sur le noyau d'un morphisme du monoïde libre," Séminaire Schutzenberger, tom 1, n^o 4, p, 1-6, (1970).
11. L. Perret, "Etude d'outils algébriques et combinatoires pour la cryptographie à clef publique," thèse de doctorat, Universit´e de Marne–la–Vallée, (2005).
12. H. Phan, P. Guillot," Preuves de sécurité des schémas cryptographiques," université Paris 8, (2013).
13. E. Post, "Recursive unthenlvability of a problem of Thue,", Journal of Symbolic Logic, 12(1):1-11, (1947).
14. S. Qiao, W. Han, Y. Li and L. Jiao, "Construction of Extended Multivariante Public Key Cryptosystems," International Journal of Network Security, Vol. 18, No.1, pp. 60-67, (2016).
15. H. Rosen, "Cryptography Theory and Practice," Third Edition, Chapman and Hall/CRC, (2006).
16. R. V. Book, H. N. Liu, "Rewriting Systems and Word Problems in a Free Partially Commutative Monoid," Information Processing Letters n^o 26, p. 29-32, (1987).