# PROBABILITY OF RETRIEVING INFORMATION IN A MULTI-ENCRYPTED ENVIROMENT

***Henryk Piech, Piotr Borowik***

*Institute of Computer and Information Science*
*Czestochowa University of Technology Poland*
*h.piech@adm.pcz.czest.pl, piotrborowikII@gmail.com*

**Abstract.** The goal of our research consists in analyzing the level of security of multi- encrypted information. We exploit a formal model for reasoning about security computer systems, i.e. perfect cryptography and the Dolev-Yao adversary model. Taking into account weakness of cryptosystem, we use probability parameters for elucidate the scale of thread connected with possibility of messages decryptions. Some cryptographic protocols and attacks on them suggest that the order of encryptions does not affect the probability decryption. We try to demonstrate that, in the case when we regard different messages encrypted by the given set of keys, the order of coding can play an essential role.

## Introduction

Dolev and Yao [1] introduced intuitive formalization of cryptographic operations. Many definitions have been proposed on the basis of approaches ranging from modal logics to algebras [2-8]. Much cryptographic analysis of security protocols leads to hypothesis that their algorithms are perfect. They need the decryption keys to extract plain text from ciphertext. Ciphertexts is generate with appropriate key and message. Regarding these assumptions and given number of protocol sessions the insecurity problem is decidable [9-12]. However, it remains an open questions, whether this result remains valid when intruder model is extended by, for example, low level cryptographic primitives [14, 15]. The unification algorithms [8] are prepared for handling properties of Diffie-Hellman cryptographic systems. These results do not solve more general insecurity problems. In this paper, we show that the insecurity problem that use public-key encryptions operators admits combinatorial methods relay on finding repeated keys in different sequences in operation encryption process. The intruder dealing is treated as a process, referring to the probabilistic polynomial time description form. It permits to randomly guess data, obtain results of statistical analysis of exchanged information, exploit keys weakness, use well-known attacks to the used algorithms and exploit partial information to reduce the range of searches. For the used model the probability of illegally cryptanalyzing information from ciphertext may be not negligible. So, we abandon the perfect cryptography assumption and we investigate encryption structure that may be violated [16, 17]. When computing the probability of retrieving data, the intruder knowledge increases as it succeeds in obtaining new information. These considerations are discordant with the usual assumptions made by formal models, which do not define security in terms of probability of successful attacks. As a consequence, in practice, formal proofs are not enough to guarantee system security. We propose definitions and estimations for probability parameters for different kind of encryptions.

## 1. Grammar base for chosen encryption notions

We consider a different type of encrypted information processes sending:

$$
\begin{array}{ll}
1) & A_1 \rightarrow A_2: \{M\}_{K1} \\
 & A_2 \rightarrow A_3: \{\{M\}_{K1}\}_{K2} \\
 & \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\
 & A_{n-1} \rightarrow A_n: \{\{M\}_{K1}\}\ldots\}_{Kn} \\
2) & A \rightarrow B: (\{\{\{\{M\}_{KB}\}_{KB}\}\ldots\}_{KB}) \\
3) & A \rightarrow B: (\{\{\{\{M\}_{K1}\}_{K2}\}\ldots\}_{Kn}) \qquad (1)
\end{array}
$$

All of them as well as different others can be intercepted by intruders. So we treated them similarly as multi-encrypted secret information.

In our investigation and examples the simplified notation will be used $A(B) \rightarrow I$: $(\{M\}_K)$ will be noticed simply by $(\{M\}_K)$, where $A$,$B$ are honest users and $I$ is intruder or $A(B) \rightarrow I$: $(\{\{\{\{M\}_{K1}\}_{K2}\}\ldots\}_{Kn})$ will be noticed simply by $(\{\{\{\{M\}_{K1}\}_{K2}\}\ldots\}_{Kn})$.

The first notation means that the user $A$ or $B$ sends encrypted message $(\{M\}_K)$ and this information is intercepted by intruder $I$. The consequence of this fact can be described: $I \rhd (\{M\}_K)$, which means that the intruder obtains information $(\{M\}_K)$ (information not addressed to him).

The second notation means that the user $A$ or $B$ sends a multi-encrypted message $(\{\{\{\{M\}_{K1}\}_{K2}\}\ldots\}_{Kn})$ and this information is intercepted by intruder $I$: $I \rhd (\{\{\{\{M\}_{K1}\}_{K2}\}\ldots\}_{Kn})$.

Hence, we can describe facts about intercepting information by the intruder simply as $(\{M\}_K)$ or $(\{\{\{\{M\}_{K1}\}_{K2}\}\ldots\}_{Kn})$. Expressions are defined by the grammar:

$M$,$N$:: = expressions,
$K$ key ($K \in$ *Keys* (nonempty set of key symbols),
$m$ fixed length string: plain - text massage,
$(M,N)$ pair,
$\{M\}_K$ encryption of $M$ under $K$.

Let’s recall the often used relation $M \mapsto N$ which says that $N$ can be derived from $M$. This relation has the following features :

$$\begin{array}{lcl} M \mapsto M, & & \\ M \mapsto N_1 \wedge M \mapsto N_2 & \Rightarrow & M \mapsto (N_1,N_2), \\ M \mapsto (N_1,N_2) & \Rightarrow & M \mapsto N_1 \wedge M \mapsto N_2, \\ M \mapsto N \wedge M \mapsto K & \Rightarrow & M \mapsto \{N\}_{\mathrm{K}}, \\ M \mapsto \{N\}_{\mathrm{K}} \wedge M \mapsto K & \Rightarrow & M \mapsto N. \end{array} \qquad (2)$$

The expression $N = (\{M\}_{KA}, \{K_A\}_{KB})$ consists of two coded texts. The $K_A$ key can be decrypted with a difference probability which depends on the length of $K_B$. Retrieving $M$ from $N$: $N \mapsto M$ in polynomial time we can assume that the probability of secrecy braking $p$ is equal 1.

$$N = (\{M\}_{KA}, \{K_A\}_{KB}) \approx N = (\{M\}_{KA}, K_A)$$

Let’s introduce the probability in process of stepwise secrecy braking associated with keys cracking. We assume that initial knowledge of obtaining useful information is obtained with probability equals 1: $p(\{M\}_{KA}, G) = 1$.

The user can obtain $M$ from $\{M\}_{KA}$ if and only if $K_A$ can be derived from

$$\mathrm{G}\ (\mathrm{G} \mapsto \mathrm{K_A}).$$

Multi-encrypted $M$:

$$(\{\{\{\{M\}_{K1}\}_{K2}\}\ldots\}_{Kn})$$

will be decrypted with probability $p_1 * p_2 * \ldots * p_n$, where $p_{i>1}$ - probability of decrypting $(\{\{\{\{M\}_{K1}\}_{K2}\}\ldots\}_{Ki-1})$ .

$M$ encrypted by multi-encrypted keys

$$(\{M\}_{K1},\ \{\{\{K_1\}_{K2}\}\ldots\}_{Kn})$$

will be decrypted with probability $q_1 * q_2 * \ldots * q_n$, where $q_{i>1}$ - probability of decrypting $K_{i-1}$.

We can present grammar of multi-encryption of message and multi-encryption of key:

$(\{\{\{\{M\}_{K1}\}_{K2}\}\ldots\}_{Kn})$ ::= multi-encrypted message,<br>
$n$ degree of encryption nesting $p_i$ probability of decryption $(\{\{\{\{M\}_{K1}\}_{K2}\}\ldots\}_{Ki-1})$,<br>
where $K_{i+1}, K_{i+2}, \ldots, K_n$ had been already decrypted,<br>
$pr_i = p_n * p_{n-1} * \ldots * p_i$ probability of decryption $(\{\{\{\{M\}_{K1}\}_{K2}\}\ldots\}_{Ki-1})$,<br>
where<br>
none of $K_{i+1}, K_{i+2}, \ldots, K_n$ had been already decrypted.

and

$(\{\{\{K\}_{K1}\}\ldots\}_{Kn})$ ::= multi-encrypted key

$n$ degree of encryption nesting $q_i$ probability of decryption $K_{i-1}$, where $K_{i+1},K_{i+2},\ldots,K_n$ had been already decrypted,

$qr_i = q_n{*}q_{n-1}{*}\ldots{*}q_i$ probability of decryption $K_{i-1}$, where none of $K_{i+1},K_{i+2},\ldots,K_n$ had been already decrypted.

Realizing different operations of communication in the network it is possible to obtain the same secrets from different sources. Communication operations are often associated with coding procedures leading to the nested encrypted secret:

$$\begin{gathered}
(\{\{\{\{M1\}_{K1,1}\}_{K2,1}\}\ldots\}_{Kn,1}),\\
(\{\{\{\{M\}_{K1,2}\}_{K2,2}\}\ldots\}_{Kn,2}),\\
\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\\
(\{\{\{\{M\}_{K1,r}\}_{K2,r}\}\ldots\}_{Kn,r}),
\end{gathered}$$

and

$$\begin{gathered}
(\{M\}_{K1,r+1},\ \{\{\{K_{1,r+1}\}_{K2,r+1}\}\ldots\}_{Kn,r+1}),\\
(\{M\}_{K1,r+2},\ \{\{\{K_{1,r+2}\}_{K2,r+2}\}\ldots\}_{Kn,r+2}),\\
\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\\
(\{M\}_{K1,r+s},\ \{\{\{K_{1,r+s}\}_{K2,r+s}\}\ldots\}_{Kn,r+s}),
\end{gathered} \tag{3}$$

where $r$, $s$ - numbers of operations with multi encrypted secrets and keys, respectively.

For every set of operations we can estimate probability of decrypting secret $M$: $pq_1,pq_2,\ldots\ldots,pq_u$, where $u$ - number of set of operations.

From all operations variants we chose the formula below, which is associated with maximum $pq_i$ probability:

$$io(M)=\{v \mid pq=\max\ \{pq_1,pq_2,\ldots,pq_v,\ldots,pq_u\}.$$

Obviously, it is possible that $K_{i,j} = K_{l,m}$. Hence, there is possible earlier encryption of information, which will be used in different operations $i$ in future, which can decrease the probability level $pq_v$.

Let's introduce parameter of message and key encryption probability with the following grammar:

$pq_i$ ::= message encryption probability (message obtained from $i$-th operation),

$pq_i^{(j)}$::= $K_j$ encryption probability (from $i$-th operation),

$i$ - numbers of chronologically sequenced operations,

$j$ - codes of keys.

$$(\{\{\{\{M\}_{Kj,i}\}_{Kj+1,i}\}\ldots\}_{Ks,i}) \mapsto (\{\{\{\{M\}_{Kj,i}\}_{Kj+1,i}\}\ldots\}_{Ks-1,i})\ \big|\, pr_s = p_n{*}p_{n-1}{*}\ldots{*}p_s$$

for *i*-th operation

$$(\{\{\{\{M\}_{Kj,l}\}_{Kj+1,l}\}\ldots\}_{Ks,l}) \mapsto (\{\{\{\{M\}_{Kj,l}\}_{Kj+1,l}\}\ldots\}_{Ks-1,l})\ |pr_s = 1$$

for $l \neq i$-th operations

where $A \mapsto B\ |p - B$ derived from $A$ with probability $p$.

Let's show several examples:

Example 1. In rows chronologically sequenced operations:

$$\begin{gathered}(\{\{\{\{M\}_{K1}\}_{K2}\}) \\ (\{\{\{\{M\}_{K5}\}_{K3}\}_{K4}) \\ (\{M\}_{K1},\ \{K_1\}_{K5}) \\ (\{M\}_{K3},\ \{K_1,K_2\}_{K5})\end{gathered} \tag{4}$$

$$pq(M1) = \max\ \{p_2{*}p_1,\quad p_4{*}p_3{*}p_5,\quad q_5{*}p_1,\ \ p_4{*}p_3\}.$$

Example 2. The permutation of keys in multi-encryption of message

$(\{\{\{\{M2\}_{K1}\}_{K2}\}_{K3}\}_{K4})$

$$\begin{gathered}(\{\{\{\{M2\}_{K2}\}_{K4}\}_{K1}\}_{K3}) \\ (\{\{\{\{M2\}_{K3}\}_{K4}\}_{K2}\}_{K1}) \\ (\{\{\{\{M2\}_{K4}\}_{K3}\}_{K1}\}_{K2})\end{gathered} \tag{5}$$

$$pq(M2) = \max\{p_4{*}p_3{*}p_2{*}p_1,\ \ p_3{*}p_1{*}\,p_3{*}p_2,\ \ p_1{*}p_2{*}p_4{*}p_3,\ \ p_2{*}p_1{*}p_3{*}p_4\} = p_1{*}p_2{*}\,p_3{*}p_4.$$

Example 3. The permutation of keys in multi-encryption of key

$$\begin{gathered}(\{M3\}_{K1},\ \{K_1\}_{K3}\}_{K2}\}_{K4}) \\ (\{M3\}_{K3},\ \{K_3,K_2\}_{K1}\}_{K2}\}_{K4}) \\ (\{M3\}_{K2},\ \{K_2\}_{K4}\}_{K1}\}_{K3}) \\ (\{M3\}_{K4},\ \{K_4\}_{K3}\}_{K2}\}_{K1})\end{gathered} \tag{6}$$

$$pq(M3) = \max\{q_4{*}q_2{*}q_3{*}p_1,\quad q_4{*}q_2{*}\,q_1{*}p_3,\quad q_3{*}q_1{*}q_4{*}p_2,\quad q_1{*}q_2{*}q_3{*}p_4,\quad q_4{*}q_2{*}p,\ \ q_4{*}p_2,\ \ p_4,\ \ q_4{*}q_2{*}p_1,\ \ q_3{*}p_1,\ \ q_3{*}q_1{*}q_4,\ \ p_1,\ \ q_1{*}q_2{*}p_3,\ \ q_1{*}p_2\}$$

## 2. Threats follow from set of multi-encrypted operations

Let's assume, that the length of all $K_i$ is the same and probabilities of their encryption also are the same and equals $p$. On the stipulation with it we have $pq\ (M1) = p^2$, $pq\ (M2) = p^4$, $pq\ (M3) = p$.

To accelerate decryption processes for multi encryption messages we can find the same keys sequences in specific encryption matrix. This matrix we create on base of sequence, chronology for every operation. The form of such matrix contains keys data in rows in particular operation (Table 1).

Table 1

**Encryption messages matrix *EMK*, where: *ind* $K_{(i,j)}$ - index of *i*-th key in encryption chronology $(\{\{\{\{*\}_{K1}\}_{K2}\}_{\dots}\}_{Kn})$ and in operation *j*, *lm* - the number of operations**

| | $t1$ | $t2$ | .... | $ti$ | ...... | $tn$ |
|---|---|---|---|---|---|---|
| *op.* 1 | *ind* $K_{(1,1)}$ | *ind* $K_{(2,1)}$ | .... | *ind* $K_{(i,1)}$ | ... | *ind* $K_{(n,1)}$ |
| *op.* 2 | *ind* $K_{(1,2)}$ | *ind* $K_{(2,2)}$ | .... | *ind* $K_{(i,2)}$ | .... | *ind* $K_{(n,2)}$ |
| ..... | | ..... | ..... | .... | ..... | ..... |
| *op. lm* | *ind* $K_{(1,lm)}$ | *ind* $K_{(2,lm)}$ | ..... | *ind* $K_{(n-1,lm)}$ | ..... | *ind* $K_{(n,lm)}$ |

For next example encryption messages matrix will have following from (Table 2):

Table 2

**Encryption messages matrix - example**

| | $t1$ | $t2$ | $t2$ | $t4$ |
|---|---|---|---|---|
| *op.* 1 | 1 | 2 | 3 | 4 |
| *op.* 2 | 2 | 4 | 1 | - |
| *op.* 3 | 3 | 4 | - | - |
| *op.* 4 | 4 | 3 | 1 | 2 |

Obviously, we can use information about broken key $K_4$ and $K_3$ to decryption message from operation *3*.

Similarly, we use the data from encrypted keys. In this kind of encryption we can build two tables: the first for keys direct encrypted messages *EMM*, and the second only for encrypted keys *EMK* (Tables 3, 4).

Table 3

**Keys direct encrypted messages**

| | $t1$ |
|---|---|
| *op.* 1 | *ind* $K'_{(1,1)}$ |
| *op.* 2 | *ind* $K'_{(1,2)}$ |
| ..... | |
| *op. lm* | *ind* $K'_{(1,lm)}$ |

Table 4

**Encryption keys matrix**

| | $t1$ | $t2$ | .... | $ti$ | ...... | $tn$ |
|---|---|---|---|---|---|---|
| *op.* 1 | *ind* $K_{(1,1)}$ | *ind* $K_{(2,1)}$ | .... | *ind* $K_{(i,1)}$ | ... | *ind* $K_{(n,1)}$ |
| *op.* 2 | *ind* $K_{(1,2)}$ | *ind* $K_{(2,2)}$ | .... | *ind* $K_{(i,2)}$ | .... | *ind* $K_{(n,2)}$ |
| ..... | | ..... | ..... | .... | ..... | ..... |
| *op. lm* | *ind* $K_{(1,lm)}$ | *ind* $K_{(2,lm)}$ | ..... | *ind* $K_{(n-1,lm)}$ | ..... | *ind* $K_{(n,lm)}$ |

Table 5

**Keys direct encrypted messages - example**

| | *t*1 |
|---|---|
| *op.* 1 | 2 |
| *op.* 2 | 2 |
| *op.* 3 | 3 |
| *op.* 4 | 4 |

Having such prepared data, we can propose analysis methods of possibility messages decryption, which delivers the shortest paths to the dangerous situation of:

– confidentiality braking consists in reaching the first open figure of message,
– confidentiality braking consists in reaching all open figures of messages. Our first proposed method referrers to multi encrypted messages and bases on:
– selection one of encryption sequence represented by chronologically ordered key indexes - current pattern (one on row in Table 1),
– searching the same patterns sequences in processes of encryption in remained encryption sequences,
– comparison of common sequences locations and estimate the scale of differentiation *SD*(*i*) (it will be also measure of decryption probability), where *i*-number of rows which play role of pattern location.

$$SD(i) = \{ \sum_{u=1}^{ls(i)} \sum_{j=1}^{lm} \sum_{v=lsc(j)(-1)}^{lcs(j)-ls(i)+1} (loc\_K_{(v,j)} > loc\_K_{(u,i)}) : K_{(u,i)} = K_{(v,j)} \}, \tag{7}$$

where:
*i*,*j* - numbers of operation,
*u*,*v* - numbers of keys on encryption sequence,
*ls* (*i*,*j*) - length of pattern sequence of encryptions in *i*-th rows,
$loc\_K_{(u,i)}$ - location of *u*-th key in *i*-th row,
($loc\_K_{(v,i)} > loc\_K_{(u,i)}$) - is binary evaluated {0,1},
(−1) - means negative step of index *v* changing.

So, *SD*(*i*) shows us the maximum decryption probability, increasing (maximum acceleration of decryption process), results from using decrypted keys from one of the other operations (rows in above presented matrices).

Graphically, we can present this algorithm as in Figure 1. Additionally, for every operation (row *i*), we can find the most cooperated operation (row *j*).

$$j_{max}(i) = \{j:\ \mathbf{max}_{j=1,2,\ldots,lm,\,j\neq i}\Big(\sum_{u=1}^{ls(i)}\ \sum_{v=ls(j)(-1)}^{ls(j)-ls(i)+1} (loc\_K_{(v,j)} > loc\_K_{(u,i)}):\ K_{(u,i)} =$$
$$= K_{(v,j)}\}\,\{j \in SD(i)\}\ . \qquad (8)$$

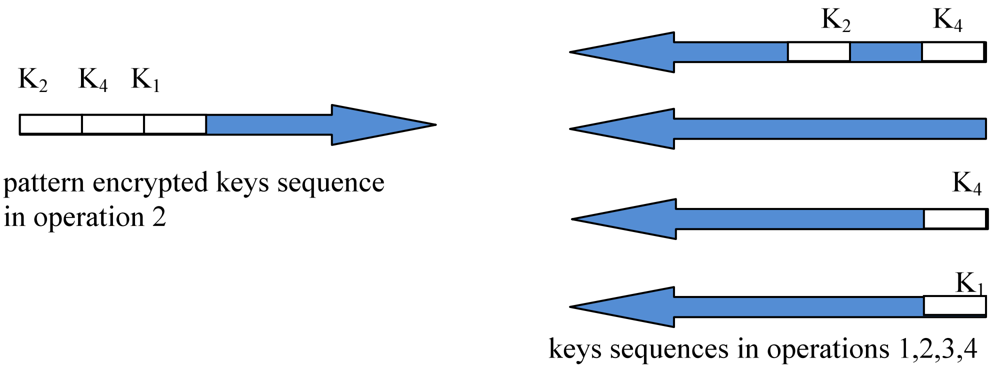
K2
K4
K2 K4 K1
pattern encrypted keys sequence
in operation 2
K4
K1
keys sequences in operations 1,2,3,4

Fig. 1. Algorithm graph presentation of finding earlier decrypted keys in all operations in case of multi-encrypted messages $SD(2) = 4$

Results of finding maximal cooperation are presented as example in Figure 2.

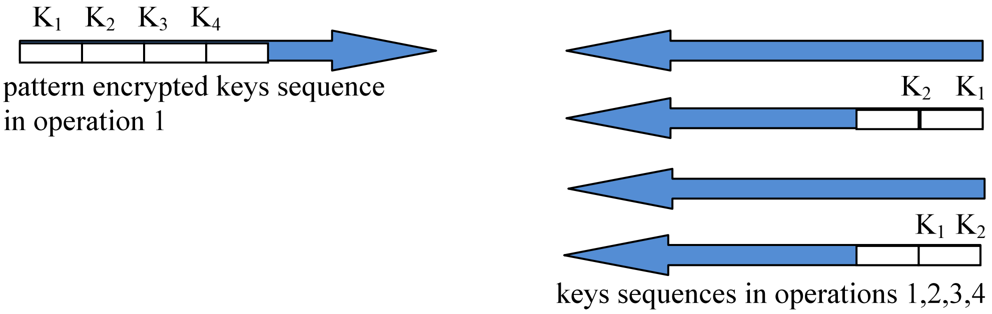
K1 K2 K3 K4
pattern encrypted keys sequence
in operation 1
K2 K1
K1 K2
keys sequences in operations 1,2,3,4

Fig. 2. Mutual cooperation between operations 1 and 2 (compare with Fig. 1). Here, $(j)_{\max}(1) = 2$ and $(j)_{\max}(1) = 4$

Intuitively, we could await that, according to (10), that large cooperation between $i$-th and $j$-th operations ($i$ $|max>$ $j$) leads to small cooperation between $j$-th and $i$-th operations ($i$ $|min>$ $j$) (it could follows from constrain $loc\_K_{(v,j)} > loc\_K_{(u,i)}$), but it isn't true (see Fig. 1). However, sometimes it is possible to build a sequence of operations based on maximum cooperation:

$$(i_1\ |max>\ i_2),$$
$$(i_2\ |max>\ i_3),$$
$$\ldots\ldots\ldots\ldots\ldots$$
$$(i_{n-1}\ |max>\ i_n).$$

where $i_s$ $|max>$ $i_{s+1}$ - in operation $i_s$ are exploited decrypted keys, obtained from operation $i_{s+1}$.

The direction of cooperation can point out a set of operations and can indicate mutual appointments (Tab. 6 - last column).

Table 6

**Proposition of cooperation among operations in decrypting process**

| | *t*1 | *t*2 | *t*2 | *t*4 | \|*max*> |
|---|---|---|---|---|---|
| *op.* 1 | 1 | 2 | 3 | 4 | 2;4 |
| *op.* 2 | 2 | 4 | 1 | - | 1 |
| *op.* 3 | 3 | 4 | - | - | 1 |
| *op.* 4 | 4 | 3 | 1 | 2 | 1 |

Coming back to probabilities we can say that probability of decrypting all messages increases in $\prod_{i=1}^{lm} p^{SD(i)}$ times. When a given key is decrypted we exclude adequate probability factor from the multiplication formula.

At last, the set of cooperation with *i*-th operation encryptions *SC*(*i*) is built:

$$SC(i) = \{\cup j\colon \sum_{u=1}^{ls(i)} \sum_{v=ls(j)(-1)}^{ls(j)-ls(i)+1} (loc\_K_{(v,j)} > loc\_K_{(u,i)}) \mid_{K(u,i) = K(v,i)} > 0\}, \qquad (9)$$

where $(*)|_{K(u,i) = K(v,i)}$ - value of expression under condition $K_{(u,i)} = K_{(v,j)}$.

So, for all rows in our example we have the following set of cooperation in decryption process:

$$SC(1) = \{2,4\},$$
$$SC(1) = \{1,3,4\},$$
$$SC(1) = \{1,3,4\},$$
$$SC(1) = \{1\}.$$

Total set of cooperation among all operations encryptions is equal:

$$SC = \bigcup_{i=1}^{lm} SC(i)$$

The second proposed method referrers to multi encrypted keys and bases on:

- selection all keys of direct encrypted messages $K'_{(1,j)}$ (Tables 3, 5),
- searching the same keys in sequences of encryption in all operations,
- estimation of the scale of decrypting all messages acceleration *KD*(*i*),

$$KD(i) = \{\mathbf{min}_{\boldsymbol{j=1,2,...lm}}\ \boldsymbol{k} \sum_{\boldsymbol{v=1}}^{\boldsymbol{lk(i)}} lk(i) - loc\_K_{(v,j)}\!: K_{(v,j)} = K'_{(1,i)}\} \qquad (10)$$

where:
*lm* - the number of operations,
*lk*(*i*) - length of encrypted keys in *i*-th operation.

So, *KD*(*i*) shows the minimal decryption process (number of stages - connected with sequenced key), leading a brake in the direct encrypted message key in *i*-th operation. Let's present the example based on Tables 2, 6. In this example we graphically mark these keys which should be broken to open all messages information. The first case refer to multi-encrypted messages and the second to multi-encrypted keys.

1. Multi-encrypted messages

Table 7

**The marking of keys indexes for needed for decryption all messages information**

| | *t*1 | *t*2 | *t*2 | *t*4 |
|---|---|---|---|---|
| *op.* 1 | 1 | 2 | 3 | 4 |
| *op.* 2 | 2 | 4 | 1 | - |
| *op.* 3 | 3 | 4 | - | - |
| *op.* 4 | 4 | 3 | 1 | 2 |

2. Multi-encrypted keys

Table 8

**The marking of keys indexes for needed for decryption all messages information**

| | *t*1 |
|---|---|
| *op.* 1 | 2 |
| *op.* 2 | 2 |
| *op.* 3 | 3 |
| *op.* 4 | 4 |

| | *t*1 | *t*2 | *t*2 | *t*4 |
|---|---|---|---|---|
| *op.* 1 | 1 | 2 | 3 | 4 |
| *op.* 2 | 2 | 4 | 1 | - |
| *op.* 3 | 3 | 4 | - | - |
| *op.* 4 | 4 | 3 | 1 | 2 |

In real operation sequences we can find both multi-encrypted messages and multi encrypted keys conventions. This kind of interleave mixed conventions have an internal or external character:

a) $(\{\{\{\{M2\}_{K1}\}_{K2}\}_{K3}\}, \{K_1,K_2\}_{K5})$

b) $(\{\{\{\{M\}_{K2}\}_{K1}\})$

$(\{M\}_{K1}, \ \{K_1\}_{K5})$

$(\{M\}_{K3}, \ \{K_3,K_2\}_{K5})$

$(\{\{\{\{M\}_{K5}\}_{K3}\}_{K4})$ (11)

## 3. Encryption with given threshold of security

The level of security is expressed by probability of decryption. Generally, it is possible to define probability of decrypted full information (all messages) sent and received by group of honest and dishonest users. In our approach we estimate this parameter as follows:

$$p = psc \prod_{i=1}^{lm} \prod_{v \notin SC} p_{(i,v)}, \tag{12}$$

where:

$psc = \prod_{u \in SC} q_u$ - probability of acceleration of decryption all messages,

$psci(i) = \prod_{u \in SC(i)} q_u$- probability of acceleration of decryption $i$-th message.

Increasing the number of stages of encryption we obviously increase the level of security and on determined stage we obtain: $p \leq thresh$, where *thresh* - given level of security.

## Conclusions

We have shown that by using the system of keys we can regulate the level of security. The threat level is the grater the larger is set of common (the same) keys used in different stages and in different of encrypting operations. To effectively increase security, in the better variant, we obviously use unexploited, keys up till now.

## References

[1] Dolev D., Yao A., On the security of public-key protocols, IEEE Transactions on Information Theory 1983, 29, 198-208.

[2] Degano P., Zunino R., A Note on the Perfect Encryption Assumption in a Process Calculus, Foundations of Software Science and Computation Structures (FOSSACS'04), Springer Verlag, 2004.

[3] DeMillo R.A, Lynch N.A., Merritt N.A., Cryptographic Protocols, Proc. of the 14th Annual ACM Symposium on Theory of Computing, ACM Press 1982, 383-400.

[4] Durante A., Focardi R., Gorrieri R., A Compiler for Analysing Cryptographic Protocols Using Non-Interference, ACM Transactions on Software Engineering and Methodology (TOSEM) 2000, 9(4), 489-530.

[5] Gray III J.W., Toward a mathematical foundation for information flow security, Journal of Computer Security 1992, 1, 255-294.

[6] Kemmerer R.A, Analyzing encryption protocols using formal verification techniques, IEEE Journal on Selected Areas in Communications 1989, 7(4), 448-457.

[7] Paulson L.C., The inductive approach to verifying cryptographic protocols, Journal of Computer Security 1998, 6(1-2), 85-128.

[8] Schneier B., Applied Cryptography, second ed., John Wiley & Sons, New York 1996.

[9] Amadio R., Lugiez D., Vanackere V., On the symbolic reduction of processes with and privacy, IEEE CS Press 1996, 174-187.

[10] Chevalier Y., Vigneron L., Towards Efficient Automated Verification of Security Protocols, Proceedings of the Verification Workshop (VERIFY’01) (in connection with IJCAR’01), 2001.

[11] Sandhu R.S., Cryptographic implementation of a tree hierarchy for access control, Information Processing Letters 1988, 27, 95-98.

[12] Rusinowitch M., Turuani M., Protocol Insecurity with Finite Number of Sessions is Npcomplete, Proc. of CSFW-14, 174-190, 2001.

[13] Pereira O., Quisquater J.-J., A Security Analysis of the Cliques Protocols Suites, Proc. of CSFW-14, 73-81, 2001.

[14] Ryan P., Schneider S., An attack on a recursive authentication protocol: A cautionary, tale, Information Processing Letters 1998, 65, 7-10.

[15] Meadows C., Narendran P., A unification algorithm for the group Diffie-Hellman protocol, Proc. of WITS, 2002.

[16] Yeh J.H., Chow R., Newman R., A key assignment for enforcing access control policy exceptions, Proceedings on International Symposium, 1998.

[17] Zhang K., Threshold proxy signature schemes, Information Security Workshop 1997, 191-197.