

# Data recovering for a neural network-based biometric authentication scheme

Dmitry Bogdanov, Vladimir Mironkin

CTCrypt  
May 30, 2018 r.

# Introduction to the problem

Classical approaches for accessing to information resources and services:

- paroles
- tokens
- smart-cards
- RFID-tags

Contradiction with ease of handling: users should remember symbolic sequences or control the safety of the mentioned devices.

# Introduction to problem

The following biometric characteristics can be used:

- retina
- fingerprints
- face
- vascular pattern
- handwritten password

# Introduction to the problem

Necessity to select an algorithm for transformation of biometric parameters into a pattern or a cryptographic key is used for further access.

The properties of such transformation:

- **independent** and **equiprobable** distributions of output bits (key bits)
- mapping the set of “close” samples of biometric parameters into the same output vector

# Introduction to the problem

A classical method of implementing biometric authentication schemes: necessity to comparison of an output vector with a predefined (secret) pattern.

**Disadvantages of this method:** need to store secret pattern (a sort of a secret key).

# Introduction to the problem

The following two frameworks were considered in ??:




- secure sketches: a pair of transformations  $SS(w) = s$  and  $Rec(w_0, s) = w$ ,  $w_0 \approx w$ , where  $w$ ,  $w_0$  — a vector of biometric parameters,  $s$  — a non-secret additional vector;
- fuzzy extractors: a pair of transformations  $Gen(w) = (R, P)$ ,  $\sim Rep(w_0, P) = R$ ,  $w_0 \approx w$ , where  $w$ ,  $\sim w_0$  — a vector of biometric parameters,  $R$  — a secret key,  $P$  — a non-secret additional vector.



1. Dodis Y., Reyzin L., Smith A. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. Lecture Notes in Computer Science, v. 3072, 2004, pp. 523-540.

# Bibliography

There are publications on fuzzy extractors for fingerprints, retina, face.

-  2. *Yang S., Verbauwhede I.* Automatic Secure Fingerprint Verification System Based on Fuzzy Vault Scheme. Proc. IEEE ICASSP 2005, p.609-612.
-  3. *Kanade S., Petrovska-Delacretaz D., Dorizzi B.* Multi-Biometrics Based Cryptographic Key Regeneration Scheme. Proc. of the 3rd IEEE international conference on Biometrics: Theory, applications and systems, p. 333-339, 2009.
-  4. *Lee Y.J., Bae K., Lee S.J., Park K.R., Kim J.* Biometric Key Binding: Fuzzy Vault Based on Iris Images. Proc. of 2nd International Conference on Biometrics, p. 800-808, 2007.

# Introduction to the problem

In work 1 it is shown, that there are exist transformations with “negligible” leakage of a secret information through a non-secret additional vector.

**The core question is:** How much this leakage for specific transformation?

Boyen in paper 5 shows the possibility to obtain the secret key when it's used multiple times for several transformations.



5. *Boyen X.*, Reusable fuzzy extractors.

**Conclusion:** the choice of a transformation does affect the security of biometric systems.



## Introduction to the problem

Specialists of JSC PNIEI and Penza State University propose an approach for constructing biometric authentication system transformations, that use of neural networks.



**The paper** asserts that the usage of neural network mechanisms increases the percentage of corrected errors up to 93% with simultaneous decrease the probability of missing errors.



*6. Bezyaev A., Ivanov A., Efimov O., Kapituloov N.*  
Comparison of the potentialities of classical and neural network mechanisms for detecting and correcting errors occurring in biometric codes during authentication. *Neurocomputers: development, application*, 6, 2009.

# GOST R 52633

We need only general requirements and properties of a training algorithm.

-  7. GOST R 52633.0-2006. Information protection. Information protection technology. Requirements to the means of high-reliability biometric authentication.
-  8. GOST R 52633.5-2011. Information protection. Information protection technology. The neural net biometry-code convertor automatic training.

## Definition

*A neuron is a weighted summation of input parameters  $x_0, \dots, x_{n-1}$ , where  $n \in N$ . An output value of the neuron is calculated by the formula  $y_i = Z(\sum_{i=1}^n w_i x_i)$  where  $w_i$  – a weight coefficient of the neuron,  $Z$  – Heaviside step function.*

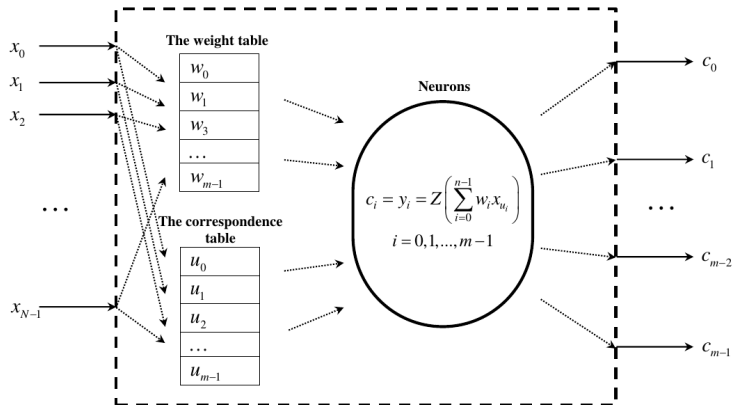
Let's assume the neural network consists of  $m$  neurons, each of them has  $n$  inputs.

Suppose the biometric data is encoded by parameters  $x_0, \dots, x_{N-1}$ .

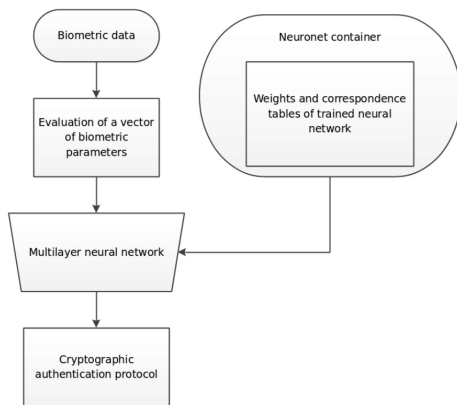
## Note

*The process of transformation of the correspondence and the weight tables at the stage of a neural network training is organized in such way that for a legitimate biometric sample at the output of the  $i$ -th neuron the  $i$ -th bit of the key sequence is calculated by the formula  $c_i = y_i = Z(\sum_{u=1}^n w_u x_{iu})$  and for any other biometric samples the correspondence bit is equiprobable.*

# GOST R 52633



**Pic.:** The generation of the key sequence  $\bar{c}$



**Pic.:** General scheme of a neural network-based biometric authentication system

# Attack on standard GOST R 52633

In paper 9 there was proposed an attack on the biometric authentication scheme, based on knowledge of the contents of neuronet container. This attack allows to recover the key effectively.



*9. Marshalko G. B. On the security of a neural network-based biometric authentication scheme, Math. Asp. of Cryptogr., 5:2 (2014), 87-98.*

# Attack on standard GOST R 52633

In response JSC PNIEI specialists proposed in work 10 the scheme of a neural network container protection using cryptographic algorithms.



10. Technical specification (project). Neural network container protection using cryptographic algorithms.  
[http://dissov.pnzgu.ru/files/dissov.pnzgu.ru/2017/tech/kolchugina/spisok\\_trudov\\_vedushey\\_org.pdf](http://dissov.pnzgu.ru/files/dissov.pnzgu.ru/2017/tech/kolchugina/spisok_trudov_vedushey_org.pdf) [in Russian].



The binary sequence  $\{\gamma_i\}_{i=0}^m$  is used to encrypt the correspondence and the weight tables. Components of this sequence are formed for each neuron separately by the hash function  $h$ :

$$\gamma_i = \left( h_i^{(1)} \parallel h \left( h_i^{(1)} \right) \parallel \dots \parallel h^{k-1} \left( h_i^{(1)} \right) \right)_{0 \sim nb-1}, \quad (1)$$

where  $b$  – length of  $w_i$ , and  $d$  – length of  $u_i$ ,  $k = \left\lceil \frac{n(b+d)}{l} \right\rceil$  and the components  $\gamma_i$  are defined as follows:

$$h_0^{(1)} = h(s \parallel p \parallel 0), \quad (2)$$

$$h_i^{(1)} = h(s \parallel p \parallel i \parallel c_0, \dots, c_{i-1}), i > 0. \quad (3)$$

where  $s$  – a fixed parameter “salt”, and  $p$  – a password.

# Protection of neural network biometric containers

The network encryption is based on a summation of  $\gamma_i$  and a concatenation of the correspondence rows of the tables  $\bar{u}$  and  $\bar{w}$ :

$$E_i = (u_i || w_i) \oplus \gamma_i. \quad (4)$$

## Note

*According to the scheme from work 10 we store the values  $E_i$ ,  $i = \overline{0, m-1}$  in a neural network biometric container instead of the tables  $\bar{u}$  and  $\bar{w}$ .*



11. GOST R 34.11-2012. Information technology. Cryptographic data security. Hash function.

# Protection of neural network biometric containers

The process of authentication for a *legitimate* user is arranged as follows: after receiving a password and a vector of input parameters  $x_0, \dots, x_{N-1}$  from user, the system calculates

$$h_0^{(1)} = h(s \| p \| 0) \quad (5)$$

and

$$\gamma_0 = \left( h_0^{(1)} \| h \left( h_0^{(1)} \right) \| \dots \| h^{k-1} \left( h_0^{(1)} \right) \right) \quad (6)$$

Knowing the  $\gamma_0$  the system calculates  $u_0 \| w_0 = E_0 \oplus \gamma_0$ .

# Protection of neural network biometric containers

Knowing the  $u_0$ ,  $w_0$  and  $x_0, \dots, x_{N-1}$ , the system calculates sequentially  $c_0$ ,  $h_1^{(1)} = h(s\|p\|1\|c_0)$  and  $\gamma_1$ . After that system calculates  $u_1\|w_1 = E_1 \oplus \gamma_1$ .

Weights and correspondence tables are restored according to the rule:

$$u_i\|w_i = E_i \oplus \gamma_i \quad (7)$$

and also calculated the sequence  $c_0, \dots, c_m$ .

# Protection of neural network biometric containers

## Note

*According to the standard a neural network training satisfies the following conditions:*

- ① *Any four successive rows of the table  $\bar{u}$  consist of  $4n$  different elements;*
- ②  *$\forall i, j \in \overline{0, N-1}$  a number of usages of  $x_i$  differs from a number of usages of  $x_j$  by no more than 2 in the table  $\bar{u}$ .*

**Our attack will be based on the use of the first property!!!**

## Recovering of the key bits

For rows of the correspondence table  $\bar{u}$  let's define the following event:

$$A_{i \sim j} = \left\{ \left\{ u_i^0, \dots, u_{j-1}^{n-1} \right\} \cap \left\{ u_j^0, \dots, u_j^{n-1} \right\} = \emptyset \right\},$$

where  $0 \leq i < j \leq m - 1$ ,  $u_b^a$  – element of the row  $u_b$ .

This means that in the row  $u_j$  are no elements that match with the elements from rows  $u_i, u_{i+1}, \dots, u_{j-1}$ .

# Recovering of the key bits

## Theorem

Let  $\bar{c} \in V_m$  is a sequence formed by the neural network with  $4n < N < 2^b$ . Let  $\bar{c}' \in V_j$ ,  $j \in \overline{1, m-1}$  is such sequence that  $c'_i = c_i$ ,  $i = \overline{0, j-2}$  for  $j > 1$ . Then

$$\mathbf{P} \{A_{\max(0, j-3) \sim j}\} = \begin{cases} 1, & c'_{j-1} = c_{j-1}, \\ \frac{(N - \min(3, j)n)^{[n]}}{2^{nb}}, & c'_{j-1} \neq c_{j-1}. \end{cases}$$

where  $u_i$  are calculated according to the standard.

# Recovering of the key bits

## Note

*If we correctly restored the values  $p, c_0, c_1, \dots, c_{j-2}, j > 2$ , then in the substitution of bit  $c'_{j-1} = c_{j-1}$  from the key sequence into expression  $h_j^{(1)} = h(s \| p \| j \| c_0, \dots, c_{j-1})$  and after elimination of  $\gamma_i$  we will get a row  $u_j$  of the correspondence table, in which with probability 1 there will be no elements from previous three rows.*



# Recovering of the key bits

## Note

*If in ?? we put  $c'_{j-1} = c_{j-1} \oplus 1$  then it will turn out some other random  $\gamma'_i$ . After elimination of  $\gamma'_i$  we will get a some random row  $u'_j$ , which can contain elements that match the elements from the previous three rows of the correspondence table.*

## Recovering of the key bits

**What is the probability** that a random row will have all elements different and will not coincide with the elements in three consecutive rows of correspondence table?

### Example

*For the scheme from work 10 with the parameters  $N = 416$ ,  $n = 32$*

$$\mathbf{P}\{A_{0\sim 1}\} \approx 2.7 \cdot 10^{-5}, \quad \mathbf{P}\{A_{0\sim 2}\} \approx 1.5 \cdot 10^{-6},$$
$$\mathbf{P}\{A_{j-3\sim j}\} \approx 5,9 \cdot 10^{-8}, \quad j \geq 3.$$

## Recovering of the key bits

- This probability is much less than 1 and allows us to build an algorithm that successively searches through the password and the bits of the key sequence.
- When we guess the true password and true bits of the key sequence in the resulting correspondence table will be no forbidden links while guessing the false password or false bits of the key sequence in the resulting table will be forbidden links with the probability close to 1.

# An algorithm

Input: an empty set.

Output:  $p, (c_0, c_1, \dots, c_{m-2})$ .

- 1 Set  $C_0 = \dots = C_{m-2} = \emptyset$ .
- 2 Randomly select  $p \in V_r$  and calculate the row  $u_0$  according to the specification.
- 3 Guess the pair of values  $c_0 \in \{0, 1\}$  and calculate the rows  $u_1$  according to the specification. If  $\bigcap_{j=0}^1 u_j = \emptyset$ , supplement  $C_0$  by the corresponding bit  $c_0$ . If  $C_0 = \emptyset$ , go to step 2, otherwise go to step 4.
- 4 Guess the pair of values  $c_1 \in \{0, 1\}$  and calculate the row  $u_2$ . If  $\bigcap_{j=0}^2 u_j = \emptyset$ , supplement  $C_1$  by the corresponding bit  $c_1$ . If  $C_1 = \emptyset$ , go to step 2, otherwise go to step 5.
- 5 Set  $i = 3$ .
- 6 If  $i \geq m$ , the algorithm finishes its work, otherwise we guess the pair of values  $c_{i-1} \in \{0, 1\}$  and calculate the row  $u_i$ . If  $\bigcap_{j=i-3}^i u_j = \emptyset$ , supplement  $C_{i-1}$  by the corresponding bit  $c_{i-1}$ .  
If  $C_{i-1} = \emptyset$ , go to step 2, otherwise set  $i = i + 1$  and go to step 6.

# Recovering of the key bits

The algorithm 1 forms the set of possible key sequences:

$$C = \{(c_0, c_1, \dots, c_{m-1}) \mid c_i \in C_i, i = \overline{0, m-2}, c_{m-1} \in \{0, 1\}\}.$$

## Note

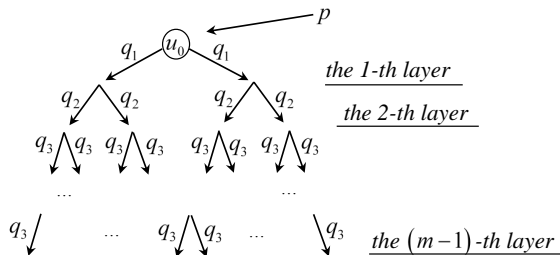
*The results of work 9 allow to define the last bit  $c_{m-1}$  and the vector of input biometric parameters  $x_0, \dots, x_{N-1}$ .*

## Definition

*Let  $q_i = \mathbf{P} \{A_{\max(0, i-3) \sim i}\}$ ,  $i \in \overline{1, m-1}$ ,  $q_i = q_3$  for  $i \geq 3$ .*

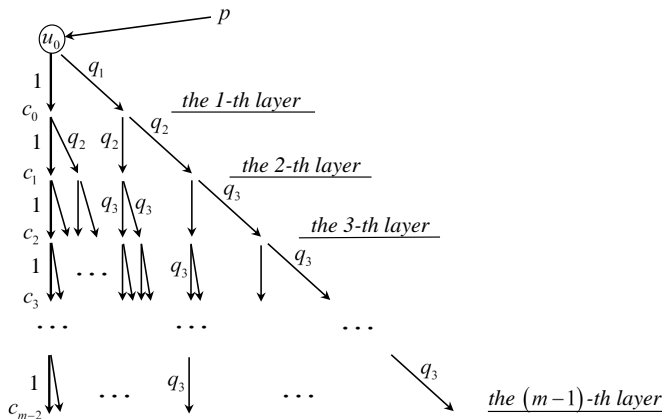
# Recovering of the key bits

We illustrate the operation of the algorithm by the graphs.



**Pic.:** The scheme of the algorithm 1 for a false  $p$

# Recovering of the key bits



**Pic.:** The scheme of the algorithm 1 for a true  $p$

## Recovering of the key bits

For a true  $p$  the desired sequence is in the set  $C$  with **probability 1!!!**

In this case the success of the algorithm consists of a rejection of all other variants of the sequence  $c_0, c_1, \dots, c_{m-2}$ . It means the  $(m-1)$ -th layer of this subtree doesn't have any vertices corresponding to the false sequences  $c_0, c_1, \dots, c_{m-2}$ .



## Recovering of the key bits

For a false  $p$  the success of the algorithm consists of a rejection of all generated sequences  $c_0, c_1, \dots, c_{m-1}$ .

Here is a table of the probability of success of the algorithm, depending on the length of the password:

$r$	<b>4</b>	<b>6</b>	<b>8</b>	<b>10</b>
Probability	$1 - 5.6 \cdot 10^{-8}$	$1 - 5.8 \cdot 10^{-8}$	$1 - 5.9 \cdot 10^{-8}$	$1 - 5.9 \cdot 10^{-8}$
$r$	<b>12</b>	<b>14</b>	<b>16</b>	<b>18</b>
Probability	$1 - 5.9 \cdot 10^{-8}$	$1 - 5.9 \cdot 10^{-8}$	$1 - 5.9 \cdot 10^{-8}$	$1 - 6.0 \cdot 10^{-8}$

# Characteristics of the recovery algorithm

How «quickly» can we reject false values  $p$ ?

## Example

*For the scheme from work 10 with the parameters  $N = 416$ ,  $n = 32$ ,  $m = 256$ , the probability of rejection false value  $p$  in no more than 6 hash function evaluations is equal to  $\mathbf{P}\{T \leq 6\} \approx 1 - 7.9 \cdot 10^{-14}$ .*

## Characteristics of the recovery algorithm

Let a bit length of password is equal  $r$ . The probability that all  $2^r - 1$  of false passwords will be reject with a common complexity  $T_r < 6 \cdot (2^r - 1)$  for the scheme from work 9 is not less than:

$r$	<b>4</b>	<b>6</b>	<b>8</b>	<b>10</b>
$T_r$	$1 - 1 \cdot 10^{-12}$	$1 - 5 \cdot 10^{-12}$	$1 - 1 \cdot 10^{-11}$	$1 - 8 \cdot 10^{-11}$
$r$	<b>12</b>	<b>14</b>	<b>16</b>	<b>18</b>
$T_r$	$1 - 3 \cdot 10^{-10}$	$1 - 1 \cdot 10^{-9}$	$1 - 5 \cdot 10^{-9}$	$1 - 2 \cdot 10^{-8}$

# Characteristics of the recovery algorithm

The probability that while guessing true password the algorithm will receive true key sequence (and only true key sequence) for less than  $2(m - 1)$  hash function evaluations for scheme from work 9 is not less than  $1 - 4 \cdot 10^{-5}$ .

## Note

*This value does not depend on the length of password but only on the key length.*

## Characteristics of the recovery algorithm

The approximate values of the probability that algorithm for the scheme from work 9 with the total complexity less then  $T_r \leq 2m+3(2^{r+1} - 1)$  will receive true password and true key sequence are presented in the table:

$r$	<b>4</b>	<b>6</b>	<b>8</b>	<b>10</b>	<b>12</b>	<b>14</b>	<b>16</b>
Probability	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999

# Conclusion

- Protection scheme of a neural network container **does not provide any protection** to the proposed method.
- The proposed method of the key information recovering **regardless** of the used encryption rules.
- The proposed method of the key information recovering **does not require** a knowledge of the biometric data.
- The proposed method of the key information recovering allows to **restore all parameters** determining the neural network, including the biometric template.
- The obtained results showed the key sequence recovering is equivalent to the password recovering.

Thank you for your attention!