Stochastic Data Transformation Boxes for Information Security Applications

Ahmad Albatsha and Michael A. Ivanov(✉)

National Research Nuclear University "MEPhI" (Moscow Engineering Physics Institute), Kashirskoe highway 31, 115409 Moscow, Russian Federation
maivanov@mephi.ru

Abstract. Stochastic methods are commonly referred to as methods which are directly or indirectly based on using a pseudo-random number generator (PRNG). In some cases, stochastic methods are the only possible mechanism of protecting information from an active adversary. In this paper we examine a construction of *R*-boxes, which are a generalization of *S*-boxes, classical structural elements of cryptographic primitives of hashing, block and stream encryption. *R*-boxes are in fact stochastic adders, i.e. adders with an unpredictable operating result, which depends on the key table *H*. A distinguishing feature of *R*-boxes is their efficient software and hardware implementation.

Keywords: Stochastic transformation · R-box · Random Feedback Shift Register (RFSR) · Non-linear M-sequence

1 Introduction

An analysis of information security threats and development of computer technologies allows to arrive at a firm conclusion that the role of stochastic methods of information security is constantly increasing. Stochastic methods are commonly referred to as methods which are directly or indirectly based on using pseudo-random number generators (PRNG). As an example of a universal stochastic method of information security we can mention the method of introducing unpredictability in the operation of means and objects of security. By using PRNG all tasks of information security can be solved successfully. Thus, in some cases stochastic methods are the only possible mechanism of protecting information from an active adversary. A particular case of stochastic methods are cryptographic methods of information security.

The term "stochastic" in relation to information security applications was, apparently, first used by S.A. Osmolovsky in constructing codes which detect and correct mistakes arising when transferring data through communication channels (Osmolovsky 1991, 2003). The stochastic codes suggested by him offer unique properties, two of which are worth mentioning. They are: the ability to provide a predefined probability of correct information reception and the possibility to solve, beside the task of error detection and correction during data transmission, two other important tasks of information security – providing confidentiality and integrity of the information transferred.

A.V. Samsonovich and V.V. Klimov (eds.), *Biologically Inspired Cognitive Architectures (BICA) for Young Scientists*, Advances in Intelligent Systems and Computing 636,
DOI 10.1007/978-3-319-63940-6_31

2 Stochastic Transformation Blocks. R-Boxes

The reference (Asoskov et al. 2003) suggests a block of stochastic transformation (*R*-box), which can be effectively applied when solving various tasks related to information security. The construction of one of the possibly simplest variants of stochastic transformation block *R*, which was first proposed for solving the task of error correcting coding in operation (Osmolovsky 1991), and its graphical representation are shown in Fig. 1. The key information of *n*-bit *R*-box is filling the table $H = \{H(m)\}$, $m = 0, \ldots, (2^n - 1)$, of dimension $n \times 2^n$, which contains elements $GF(2^n)$, mixed in a random fashion, i.e. $H(m) \in GF(2^n)$. In other words, the table *H* contains consecutive states of *n*-bit PRNG. The result $R_H(A, B)$ of the transformation of the input *n*-bit binary set *A* depends on how the table *H* is filled as well as on the transformation parameter *B*, specifying displacement in the table with respect to the cell holding the value *A*, in the following way $R_H(A, B) = H((m_A + B) \bmod 2^n)$, where m_A is the address of the cell in the table *H*, containing the code *A*, i.e. $H(m_A) = A$. Otherwise speaking, the result of the operation of *R*-box consists in reading the cell content in the table *H*, repeatedly displaced at *B* positions toward major addresses with respect to the cell containing the code *A*. To ensure the independence of transformation time from the source data we introduce into *R*-box the table $H^{-1} = \{H^{-1}(j)\}$ of dimension $n \times 2^n$, where $\forall j = 0, 1, \ldots, (2^n - 1)$ $H^{-1}(j) = m_j$. To put it differently, the cell having the address *j* in the array H^{-1} holds the address of the cell of the array *H* containing the code *j*. Below are the facts which deserve attention:

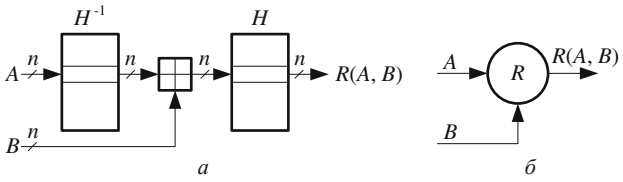

Fig. 1. The behavior of *R*- box (*a*) and its graphical representation (*b*). ⊞ – modulo 2^n adder.

- when $H^{-1} = \{0, 1, \ldots, (2^n - 1)\}$ and $B = 0$, we get a classical *S*-box (substitution box) with the substitution table *H*;
- when writing its own address in each cell of the arrays *H* and H^{-1} we get a classical 2^n adder, which means that the *R*-box can be rightfully called a *stochastic adder*, i.e. adder with an unpredictable operating result, which depends on how the key table *H* is filled.

R-box has an easy software implementation. Below follows an example of implementing an 8-bit box of stochastic transformation in Assembler (a system of commands x86, standard Intel notation) (Fig. 2).

R-boxes can be applied for implementing stream encryption. In this case, a Plaintext is fed to input *A*, a Keystream is fed to input *B*, and a Ciphertext is removed from output $R_H(A,B)$. We should bear in mind that it is essential to apply the transformation R^{-1} (reverse *R*) at the receiver.

```
;=====================================================================
;====      RBox      -      the      procedure      of      stochastic
transformation.==============
;=====================================================================
;==== When called: AL - input byte, AH - transformation parameter,
===
;==== DS - segment address of array H^-1&H, ==========================
;==== BX - relative address of array H^-1&H(fig.2), ==================
;==== CX - dimension of arrays Addr and H (HSize). ==================
;==== Upon return: AL - output byte. ================================
;=====================================================================
RBox        PROC
            push  bx
            xlat                ; Reading from list H^-1
            add   al, ah        ; AL - output byte address in array
            add   bx, cx        ; BX - relative address of array
            xlat                ; Reading from list
            pop   bx
            ret
RBox        ENDP
;=====================================================================
```

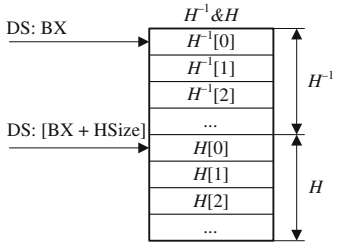

Fig. 2. Array $H^{-1}\&H$

The second area of application of R-boxes is substitution of modulo 2^n adders in modifying known algorithms of stochastic data transformation, for example, stream algorithms PIKE (Fig. 3), RC4, and several others (Asoskov et al. 2003; Stallings 2016; Hammood et al. 2016; McKague 2016; Rivest and Schuldt 2016). In addition, with the help of an R-box it is possible to substitute modulo 2^n adder in two ways in Fig. 1 and thus get two kinds of R^2-boxes.

Let us consider an example of a nonlinear transformation on the basis of RFSR (Random Feedback Shift Register), which is obtained after substituting the modulo 2^n adder with an R-box in the architecture of an additive generator (Asoskov et al. 2003).

Let the number of bits in Q state (the number of storage elements) RFSR be 128: $|M| = |Q| = 128$, $Q = (Q_{16} \ldots Q_1)$, $Q_i \in GF(2^8)$, $i = 1, \ldots, 16$. The nonlinear stochastic transformation on the basis of RFSR, constructed in accordance with Galois module (Fig. 4), is defined by the following expressions:

$$F(Q) = f^{16}(Q) = f^{16}(Q_{16} \| \ldots \| Q_1).$$

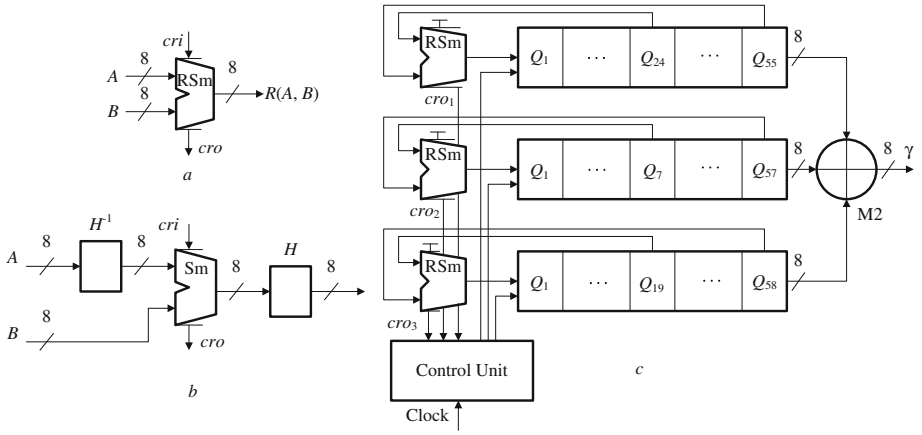

Fig. 3. Modified PRNG of stream cipher PIKE: *a* – the graphical representation of the stochastic adder; *b* – the scheme of the stochastic adder; *c* – the scheme of PRNG. Sm – modulo 256 adder, RSm – stochastic 8-bit adder, M2–8-bit modulo 2 adder, γ – PRNG output, *cri* – Carry In, *cro* – Carry Out.

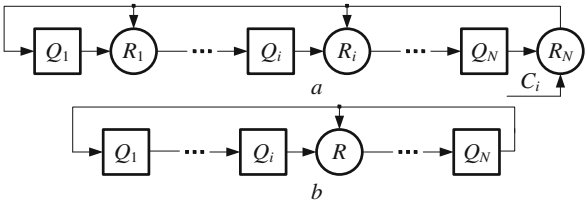

Fig. 4. RFSR: *a* – general scheme; *b* – RFSR with a single *R*-box

The equation of the base transformation f takes the following form:

$$\begin{cases} Q_j = R_j(Q_1, Q_{j+1}), j = 1, \ldots, 15, \\ Q_{16} = R_{16}(Q_1, C_i). \end{cases}$$

where $C = C_1 \ldots C_i \ldots C_{16}$ is the control sequence (which probably depends on the key).

Finally, RFSR has a remarkable property: when choosing the appropriate table H of stochastic transformation it is possible to obtain a nonlinear maximum length sequence (M-sequence) generator, whose properties differ fundamentally from those of linear M-sequences, formed by LFSR. Figure 5 shows an example of the M-sequence generator of the length 63 ($N = 3$, $n = 2$, $\left| Q_i \right| = 2$).

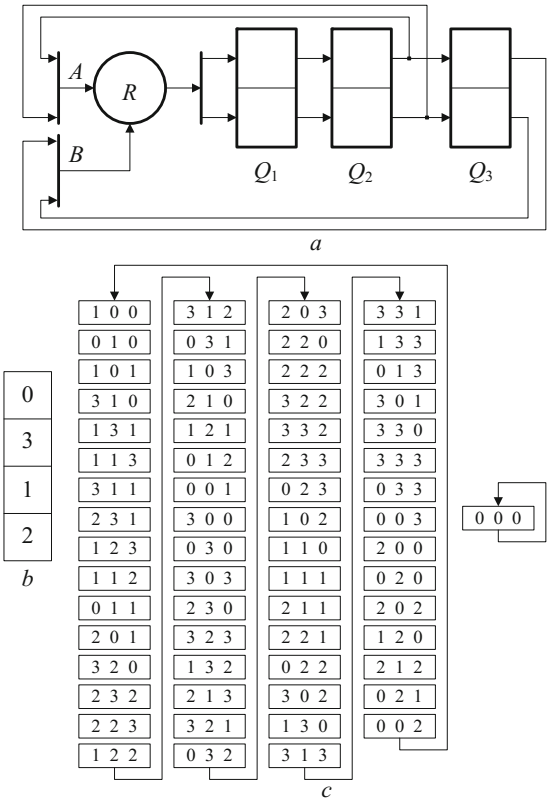

Fig. 5. Example of RFSR: *a* – the scheme of the nonlinear M-sequence generator; *b* – the table of stochastic transformation; *c* – the switching diagram of the generator.

3 Conclusion

We have analyzed the construction of *R*-boxes, which are a generalization of *S*-boxes, classical structural elements of cryptographic primitives of hashing, block and stream encryption. *R*-boxes are in fact stochastic adders, i.e. adders with an unpredictable operating result, which depends on the filling of the key table *H*, which can be formed by using a method similar specified in the stream cipher RC4.

A distinctive feature of *R*-boxes is their effective software and hardware implementation.

R-boxes can be applied in the following areas:

- Constructing blocks of direct-to-reverse stochastic transformation in the implementation of stochastic methods of data transfer (Osmolovsky 1991);
- Implementing stream encryption; in this case a Plaintext is fed to input *A*, a Keystream is fed to input *B*, and a Ciphertext is removed from output $R_H(A, B)$; in this case it is essential to use the transformation R^{-1} (reverse *R*) at the receiver;

– Increasing the cryptographic security of known algorithms by substituting modulo 2^n adders with n-bit R- boxes.

Implementing nonlinear transformations through mixing cipher state (MixState), including those depending on the key information, in the construction of 2D and 3D iterative cryptographic algorithms (Ivanov et al. 2012a; Ivanov and Chugunkov 2012b; Ivanov et al. 2014; GOST R 34.12-2015 2015).

– Constructing shift registers with stochastic feedback (RFSR), which are a generalization of classical linear feedback registers (LFSR), i.e. PRNG, functioning in finite fields, which work well in practice as structural elements in primitives of symmetric cryptography. When choosing the appropriate table H RFSR forms nonlinear M-sequences, which have different properties from those characteristic of linear M-sequences. Innovative solutions of forming sequences of the length 2^m, where m stands for the number of storage elements in PRNG, forming sequences with a tail, forming universally programmed PRNG, forming sequences of any length, less than or equal to 2^m, working in the case of LFSR, also work in a more common case, that of RFSR (Ivanov et al. 2009).

Acknowledgement. The publication is prepared in accordance with the scientific research under the Agreement between the Federal State Autonomous Educational Institution of Higher Education "National Research Nuclear University MEPhI" and the Ministry of Education and Science № 14.578.21.0117 on 27.10.2015. The unique identifier for the applied scientific research (project) is RFMEFI57815X0117.

References

Osmolovsky, S.A.: Stochastic Methods of Data Transmission. Radio i Svyaz, Moscow (1991)
Osmolovsky, S.A.: Stochastic Methods of Information Defense. Radio i Svyaz, Moscow (2003)
Asoskov, A.V., Ivanov, M.A., Mirsky, A.A., et al.: Stream Ciphers. Kudits-Obraz, Moscow (2003)
Stallings, W.: The RC4 stream encryption algorithm, 5 July 2016. people.cs.clemson.edu/~jmarty/courses/Spring-2016/CPSC424/papers/RC4ALGORITHM-Stallings.pdf
Hammood, M.M., Yoshigoe, K., Sagheer, A.M.: RC4-2S: RC4 stream cipher with two state tables, 5 July 2016. ualr.edu/computerscience/files/2014/01/Paper-12.pdf
McKague, M.E.: Design and analysis of RC4-like stream ciphers, 5 July 2016. etd.uwaterloo.ca/etd/memckagu2005.pdf
Rivest, R.L., Schuldt, J.C.N.: Spritz—a spongy RC4-like stream cipher and hash function, 5 July 2016. people.csail.mit.edu/rivest/pubs/RS14.pdf
Ivanov, M.A., Vasilyev, N.P., Chugunkov, I.V., et al.: Three-dimensional pseudo-random number generator for implementing in hybrid computer systems. Vestnik NRNU MEPhI **1**(2), 232–235 (2012a)
Ivanov, M.A., Chugunkov, I.V.: Cryptographic methods of information defense in the computer systems and networks: teaching guide. National Research Nuclear University MEPhI, Moscow (2012b)

Ivanov, M.A., Spiridonov, A.A., Chugunkov, I.V., et al.: Three-dimensional data stochastic transformation algorithms for hybrid supercomputer implementation. In: Proceedings of 17th IEEE Mediterranean Electrotechnical Conference, Beirut, Lebanon, pp. 451–457 (2014)

GOST R 34.12-2015. Information Technology. Cryptographic Information Defense. Block Ciphers. Standartinform, Moscow (2015)

Ivanov, M.A., Chugunkov, I.V., Matsuk, N.A., et al.: Stochastic Methods of Information Defense in Computer Systems and Networks. Kudits-Press, Moscow (2009)