Modeling Security Threats to Cryptographically Protected Data

Alexandra A. Savelieva

Scientific Advisor: Prof. Sergey M. Avdoshin

In this paper, we introduce a mathematical model of threats for analyzing the security of cryptographic systems based on risk management principles. We also provide economic indicators as a basis to build a rationale for investments to cryptographic systems. Some points of designing software tools to support our methodology are covered in the paper. The new approach that incorporates the threat model, automatic cryptographic strength verification tools and economic techniques, is instrumental for providing sound arguments to choose a cryptographic system and for implementing an information security strategy. An overview of alternative approaches is provided along with the results of comparative analysis revealing their drawbacks as compared to the method presented in this paper.

***Index Terms*—risk management, threat modeling, cryptographic system, discounted cash flow.**

I. Introduction

Ross Anderson, Professor in Security Engineering at the University of Cambridge Computer Laboratory and an industry consultant, concludes his well-known paper [2] saying *"the evaluator should not restrict herself to technical tools like cryptanalysis and information flow, but also apply economic tools"*. Our paper aims at providing a formal way of analyzing cryptographic systems security.

The analysis of modern publications on security revealed a lack of methods designed to facilitate the process of cryptographic protection efficiency. Formalized security risk analysis and management methodologies such as CRAMM [10], RiskWatch [24] and GRIF [11] are focused on information system security as a whole and do not consider the peculiarities of evaluating cryptographic systems. There is a mathematical model designed by V.P. Ivanov [16] which applies the principles of the catastrophe theory and queuing theory to computing of a cryptographic system efficiency indicators. Although the approach incorporates economic and technical perspective, its major restriction is that it can only apply to the so-called *restricted-use cryptographic systems* [7] whose security depends on keeping both the encryption and decryption algorithms secret. The author reduces the problem of breaking a cipher to engineering analysis of the program that implements the encryption mechanism. This assumption is inadmissible for modern cryptographic systems, being in conflict with Kerckhoffs's fundamental principle [17] that encryption should not depend on the secrecy of the system - which sooner or later would be compromised - but should solely depend on the secrecy of the key. Finally, various tools for cryptographic protocols analysis [5, 6, 8] focus only on the high-level, conceptual design of a protocol on the supposition that cryptographic algorithms satisfy perfect encryption assumptions, so the strength of ciphers remains out of scope.

II. Problem Statement

The purpose of our work is to develop an approach to analyzing the security of cryptographic systems. In order to achieve the goal, we need to:

1) formulate the steps of cryptographic systems evaluation process;
2) develop a mathematical model of security threats;
3) design software tools to facilitate the process of cryptosystem efficiency assessment by a computer security specialist;
4) select appropriate economic indicators as a basis to build an economic rationale for investments to cryptographic systems and to provide sound arguments for implementing an information security strategy.

Results of the 1st stage of our research were published in proceedings of SYRCoSE'2008 [26]. In particular, items (1) and (3) from the above list were considered. In addition to it, in [26] we described our multiple-category divisions of cryptographic systems, adversaries and attacks designed for developing of a mathematical model of security threats (2). Therefore, in this paper we will focus on new results on (2) and (4) achieved ever since; as for item (1), we will restrict ourselves to providing a brief overview. We have also decided it is appropriate to elaborate more on (3) since some important aspects of designing new tools for cryptanalysis did not receive much attention in the previous paper.

III. Cryptographic Systems Evaluation Process

The process of cryptosystem efficiency assessment can be described as a sequence of steps, each of them directed at answering a specific question [26]:

- Step 1: What cryptosystem is the object of attack?
- Step 2: Who wants to attack the cryptosystem?
- Step 3: Which attack techniques are most likely to be used to break the cryptosystem?
- Step 4: Is the cryptosystem capable of withstanding such attacks?
- Step 5: Does the cryptosystem provide sufficient security in the given context?

The environment typically imposes restrictions on the attack scenarios that the cryptographic systems are exposed to, so Steps 1 to 3 imply modeling threats to a cryptographic system in a given context. Step 4 is about analyzing the cryptographic system resistance to the types of attacks defined at Step 3. Finally, Step 5 involves using various risk analysis techniques and economic tools to evaluate the data obtained during Steps 1-4.

IV. ABC-Model of Security Threats

We can assume that the adversary is most likely to choose the attack with the maximum benefit for a given cost, or choose the least costly attack that gives them a particular benefit [27]. Each cryptosystem has a set of attacks that is applicable to it and a set of attacks that is not. These hypotheses perfectly fit into common risk-management methodologies and result in the following approach to evaluating security threats.

Each cryptanalytic attack has a value of risk associated with it and defined as the product of probability of the hazard and its potential impact:

Risk = Probability · Impact

Impact refers to effect of an attack on a specific type of cryptographic system. *Probability* reflects the likelihood that an adversary will consider a specific type of attack appropriate in terms of available resources and target secret data. Thus, a formal model of the cryptosystem coupled with formal models of the adversaries will yield a set of the most hazardous attacks that the cryptosystem is exposed to. The model of security threats represented as a composition of 3 elements will be referred to as an *ABC-model* ('A' for **a**ttack, 'B' for code**b**reaker and 'C' for **c**ryptosystem). In [26], a description of multiple-category divisions of cryptographic systems, adversaries and attacks that we suggest as a basis for modeling the components of a security threats is provided.

Let $\mathrm{A} \subseteq A_1 \times A_2 \times ... \times A_9$ be a set of parametric models of attack, where A_i ($i = \overline{1,9}$) represents a domain for the i[th] parameter as per our taxonomy [26]. Each model $\vec{a} \in \mathrm{A}$ is a vector $\left(a_1, a_2, \ldots, a_9\right)$, where $a_i \in A_i$.

Similarly, a parametric model for a code-breaker is $\vec{b} \in \mathrm{B}$, where $\mathrm{B} \subseteq B_1 \times B_2 \times ... \times B_6$, B_j ($j = \overline{1,6}$) represents a domain for the j[th] parameter, and parametric model for a cryptographic system is $\vec{c} \in \mathrm{C}$, where $\mathrm{C} \subseteq C_1 \times C_2 \times ... \times C_6$, C_k ($k = \overline{1,6}$) represents a domain for the k[th] parameter as per our taxonomies. It is important to note that sets A_i, B_j, and C_k are finite.

For simplicity, we will further omit the word 'model' when referring to parametric models of attacks, codebreakers and cryptosystems.

Let $\Re : \mathrm{A} \times \mathrm{B} \times \mathrm{C} \to [0;1]$ be a function defining the level of risk associated with an attack $\vec{a} \in \mathrm{A}$ as applied by a code-breaker $\vec{b} \in \mathrm{B}$ for cryptanalysis of a cryptosystem $\vec{c} \in \mathrm{C}$. Let function $\mathrm{I} : \mathrm{C} \times \mathrm{A} \to [0;1]$ define impact (as described above), and function $\mathrm{P} : \mathrm{B} \times \mathrm{A} \to [0;1]$ define probability. Then risk $\Re$ is evaluated as follows:

$$\Re(\vec{a},\vec{b},\vec{c}) = \mathrm{I}(\vec{c},\vec{a}) \cdot \mathrm{P}(\vec{b},\vec{a})$$

The function $\mathrm{I}(\vec{c},\vec{a})$ is recursively defined via a family of functions $\mathrm{I}_{gh} : C_g \times A_h \to \mathbb{R}_+$, $g = \overline{1,6}$, $h = \overline{1,9}$, where $\mathbb{R}_+$ is a set of nonnegative real numbers. I_{gh} defines the level of interference between parameters c_g and a_h:

- $\mathrm{I}_{gh}(c,a) = 0$, if an attack with parameter value $a \in A_h$ is inapplicable to a cryptosystem with parameter value $c \in C_g$;
- $0 < \mathrm{I}_{gh}(c,a) < 1$, if a cryptosystem parameter value $c \in C_g$ reduces the likelihood that an attack with parameter value $a \in A_h$ can achieve a success;
- $\mathrm{I}_{gh}(c,a) = 1$, in case of no correlation between parameters $c \in C_g$ and $a \in A_h$;
- $\mathrm{I}_{gh}(c,a) > 1$, if a cryptosystem parameter value $c \in C_g$ points out a high probability that an attack with parameter value $a \in A_h$ will be instrumental for cryptanalysis.

To demonstrate the dependency, an illustrative example will be useful. If a cipher is implemented in hardware, it increases the probability that *side-channel attacks* [30] based on information gained from the physical implementation of a cryptosystem (including timing, power consumption, and electromagnetic leaks) will be used to for cryptanalysis. The quantitative level of interference is defined based on expert knowledge.

Let $\overline{\mathrm{I}_{gh}} : C_g \times A_h \to [0;1]$ be a normalized function:

$$\overline{\mathrm{I}_{gh}}(c,a) = \frac{\mathrm{I}_{gh}(c,a)}{\sum_{\xi \in C_g} \mathrm{I}_{gh}(\xi,a)}$$

Then the level of damage from an attack $\vec{a} \in \mathrm{A}$ to a cryptosystem $\vec{c} \in \mathrm{C}$ is evaluated as follows:

$$\mathrm{I}(\vec{c},\vec{a}) = \min_{h=\overline{1,9}} \prod_{g=\overline{1,5}} \overline{\mathrm{I}_{gh}}(c_g, a_h)$$

If at least one parameter value contradicts the applicability of $\vec{a} \in \mathrm{A}$ to breaking $\vec{c} \in \mathrm{C}$, the function yields 0: this is achieved through using multiplicative criterion.

Accordingly, $\mathrm{P}(\vec{b},\vec{a})$ is expressed in a similar way via parameters of an attack $\left(a_1, a_2, \ldots, a_9\right)$ and a code-breaker $\left(b_1, b_2, \ldots, b_6\right)$. An example of correlation between parameters is that a brute-force attack (or any other attack which can be parallelized efficiently) is most likely to be used by an adversary who has access to distributed computation resources.

Therefore, the formula defining the level of risk associated with an attack $\vec{a} \in \mathrm{A}$ as applied by a code-breaker $\vec{b} \in \mathrm{B}$ for

cryptanalysis of a cryptosystem $\vec{c} \in \mathrm{C}$ is as follows:

$$\Re(\vec{a},\vec{b},\vec{c}) = \min_{h=\overline{1,9}} \prod_{g=\overline{1,6}} \overline{\mathrm{I}_{gh}}(c_g, a_h) \cdot \min_{h=\overline{1,9}} \prod_{t=\overline{1,6}} \overline{\mathrm{P}_{th}}(b_t, a_h)$$

If the level of risk associated with an attack $\vec{a} \in \mathrm{A}$ in a given context (defined in terms of $\vec{c} \in \mathrm{C}$ and $\vec{b} \in \mathrm{B}$) exceeds a threshold $\theta \in [0;1]$, i.e. $\Re(\vec{a},\vec{b},\vec{c}) > \theta$, then the attack will be considered as a threat that the codebreaker imposes on the cryptosystem. The *admissible risk level* θ is a customizable parameter of the ABC-threat model. When defining θ, the following two criteria are considered:

- the significance of cryptographically protected data;
- the amount of computing and storage recourses available to the specialist.

In the general case:

- a cryptosystem can comprise a number of sub-systems $\vec{c} \in \mathrm{C}'$ $\left(\mathrm{C}' \subseteq \mathrm{C}\right)$, e.g. a symmetric cipher and a key generator, each of them having a different set of applicable attacks;
- a cryptosystem can be a target for several code-breakers $\vec{b} \in \mathrm{B}'\left(\mathrm{B}' \subseteq \mathrm{B}\right)$ who differ in terms of skills, resources etc.

These assumptions yield a set of attacks $\Lambda = \bigcup_{\vec{b} \in B'} \bigcup_{\vec{c} \in C'} \lambda(\vec{b},\vec{c})$, where $\lambda(\vec{b},\vec{c}) = \left\{ \vec{a} \in \mathrm{A} : \Re(\vec{a},\vec{b},\vec{c}) > \theta \right\}$. Thus, the process of analyzing a cryptosystem's security is reduced to evaluation of its capability to resist the attacks in Λ by means of instrumental tools for cryptanalysis discussed in the following section.

When designing the ABC-model, we stemmed from the following admissions:

- the inaccuracy of expressing the interference between a combination of cryptosystem parameters and a combination of attack parameters through interference between individual parameters is negligible;
- the inaccuracy of modeling code-breakers as independent individuals who are not supposed to cooperate is negligible.

The adjustment of the ABC-model to overcome these admissions would require significant complication of the model. The question of the admissions' influence on the model accuracy is subject to further research.

It is important to note that the taxonomy for cryptanalytic attacks is applicable to modeling attacks not only on cryptosystems but also on cryptographic protocols. This is a very important property of the ABC-model: as shown in [28], the interaction between cryptosystems and cryptographic protocols has not been deeply studied and still remains an open area of research.

V. Software Tools for Cryptanalysis

Our work on designing a new tool for cryptanalysis was inspired by the need to constantly re-evaluate the cryptographic algorithms strength. Such toolkits already exist for some classes of cryptanalytic attacks and enable finding out vulnerabilities, not only on new cryptographic systems being proposed, but also on old schemes which for long have been considered secure. In particular, in the last years researchers have devoted much effort to develop techniques to formally analyze cryptographic protocols [5, 6, 8]. Another important research direction is that of designing tools for high-level side-channel attack simulation developed with the aim of automating analysis techniques to help a cryptanalyst identify possible implementation vulnerabilities with minimal effort on their side. In [22], an approach is presented based on the SystemC 2.0 language [1], the de facto standard in complex digital system simulations illustrated by a case study of analyzing various implementations of AES [23] algorithm.

In our research, we focused on developing tools for analyzing public-key cryptographic algorithms strength. Before designing a new tool for cryptanalysis, we investigated available solutions for solving discrete logarithms and integer factorization problems which are the basis for various modern cryptographic systems, such as RSA [25] and ElGamal signature scheme [12]. Our analysis is supported by an extensive survey of mathematical libraries [15]. The set of evaluation criteria that we used is as follows:

- The tools should provide efficient implementation of big integer arithmetic operations;
- The tools should be compatible with Windows platforms given the large amount of cryptographic products running under Microsoft operating systems;
- The tools should have a base upon which to write implementations of integer factorization algorithms and index-calculus algorithms for discrete logarithm computation, including algorithms for creating factor bases and linear algebra techniques for solving sparse systems of equations;
- The tools should have extensible architecture so that new methods could be easily added to the implementation with the advent of new cryptanalytic techniques;
- The tools should be completely automatic and should carry out their job even when run by users having a limited amount of expertise in the field.

The advantages of programs like Maple [13] or Mathematica [29] are unlimited precision and easy-to-program algorithms. However, they are extremely inefficient for computations in number theory.

Java also has multiprecision capabilities and is highly portable. However, it is very slow in terms of number-theoretical operations. High performance can be achieved through using of low-level programming languages. Although C and C++ built-in numeric data types have limited precision, there are a lot of multiprecision libraries with many of them available as free software (GNU GPL), e.g. LIP, LiDIA, CLN,

NTL, PARI, GMP, MpNT etc.

One of the first multiprecision libraries was LIP (Large Integer Package) [21] written by Arjen K. Lenstra and later maintained by Paul Leyland. Despite being highly portable, the ANSI C library is not appropriate for our purpose as it is not efficient. In addition to it, the library provides no base for developing number-theoretic algorithms.

A Class Library for Numbers (CLN) [9] written by Bruno Haibleand and currently maintained by Richard Kreckel is a C++ library that implements elementary arithmetical, logical and transcendental functions. It has a rich set of classes for integers, rational numbers, floating-point numbers, complex numbers, modular integers etc. The drawbacks of this universality are the lack of emphasis on speed, and hence no optimization for the specific tasks of big integer operations.

GMP (GNU Multiple Precision arithmetic library) [14] was developed by Törbjord Granlund and the GNU free software group. Although this C library for arbitrary precision arithmetic is faster than most multiprecision libraries due to its highly optimized ASM implementations for the most common inner loops and for a lot of CPUs, it has the drawbacks of being incompatible with Windows and lack of primitives to support integer factorization and DLP methods.

LiDIA [20] is a C++ library for computational number theory developed at the Technical University of Darmstadt by Thomas Papanikolau. LiDIA includes highly optimized implementations for multiprecision data types and can use different integer packages (like Berkley MP, GMP, CLN, libI, LIP etc.). LiDIA's drawback is that the library is not portable to Windows platform.

NTL (a Library for doing Number Theory) [19] written and maintained mainly by Victor Shoup is a high-performance C++ library. As shown in [15], NTL outperforms other libraries in terms of big integer operations, however it needs to be extended to become instrumental for our purposes as it has no implementation of either integer factorization or DLP algorithms.

Another disadvantage that all the libraries have in common is the high level of programming skills that a cryptanalyst needs to use them.

Since no alternative solution matches all five criteria at once, the rationale for developing new software tools was clear (see Table I). We designed software tools CRYPTO [3, 4] having in mind the efficiency criteria stated above. Multiprecision C++ library DESIGNER that is the core of CRYPTO is an extension of NTL. To provide the user with an easy access to integer factorization and DLP functions, an application ANALYST was implemented in C#. For illustration of the efficiency of CRYPTO, the timing for 55-bit DLP computation on a 3.2 GHz Intel Pentium/1Gb memory PC is 8 hours against 10 minutes that our implementation takes to compute a discrete logarithm in 80-bit field.

TABLE I
A MULTIPLE-CATEGORY COMPARISON OF AVAILABLE SOLUTIONS FOR SOLVING DISCRETE LOGARITHMS AND INTEGER FACTORIZATION PROBLEMS

Evaluation Criteria \ Alternative	Maple	LIP	CLN	LiDIA	GMP	NTL	CRYPTO
High-performance multiprecision operations	−	−	−	+	+	+	+
Compatibility with Windows platform	+	+	+	−	−	+	+
Primitives for modern cryptanalytic algorithms implementation	−	−	−	+	+	−	+
Extensible architecture	+	+	+	+	+	+	+
Usability	+	−	−	−	−	−	+

VI. ECONOMIC PERSPECTIVE

We suggest that the *discounted cash flow (or DCF) approach* [18] should be used to provide economic rationale for investments to cryptographic systems. In finance, the DCF is a method of valuing a project, company, or asset using the concepts of the time value of money. All future cash flows are estimated and discounted to give their present values. The discount rate used is generally the appropriate cost of capital and may incorporate judgments of the uncertainty (riskiness) of the future cash flows.

The cash flow R_t related to a cryptographic system can be described using the following formula:

$$R_t = -Cost_t + Profit_t \cdot (1 - \mathrm{P}_t) - Loss_t \cdot \mathrm{P}_t ,$$

where $Cost_t$ is the cost of a implementation, deployment and support of the cryptographic system;

$Profit_t$ is the value of information assets being protected;

$Loss_t$ refers to the hazard in case of unauthorized access to the asset by an adversary;

P_t is the probability of an adversary to break the cryptographic system;

t is the time (e.g. in years) before the future cash flow occurs.

VII. CONCLUSION

The paper proposes a formalized methodology for analyzing the efficiency of a cryptosystem. Model-based analysis described in this paper is a part of the five-step process designed to focus on the specific aspects of cryptographic systems security. The methodology is supported by software tools designed to evaluate the cryptographic system capability to resist various types of attacks. We expect that economic perspective introduced in this paper will be of value to security specialists for justifying IT budget and communicating their proposals to the co-workers with financial background.

The direction of our future work is the development a built-in expert knowledge base to aid in-house cryptographic systems expertise. This involves evaluating the dependency between the parameters of a cryptosystem model and the applicable attacks on the one hand, and the parameters of an attacker model and the types of attacks that they are likely to use, on the other hand.

REFERENCES

[1] American National Standards Institute and Institute of Electrical and Electronic Engineers. IEEE Standard SystemC Language Reference Manual. Std 1666 - 2005, New York, 2006

[2] Anderson R. Why information security is hard - an economic perspective // Proceedings of the 17th Annual Computer Security Applications Conference (ACSAC '01), 10-14 Dec 2001, New Orleans, Louisiana, USA, 2001.

[3] Avdoshin S.M., Savelieva A.A. Tools for asymmetric ciphers analysis: Industrial registration certificate No. 10193 dated 18.03.2008 (in Russian).

[4] Avdoshin S.M., Savelieva A.A. Tools for asymmetric ciphers analysis:. Certificate of official registration in the register of software No. 2008612526 dated 10.04.2008, Federal Service For Intellectual Property, Patents And Trademarks. (in Russian).

[5] Bodei C., Buchholtz M., Degano P., Nielson F., Riis Nielson H. Automatic validation of protocol narration. In Proceedings of the 16th IEEE Computer Security Foundations Workshop (CSFW 2003), IEEE Computer Society Press, Washington, 2003. Pp. 126 - 140.

[6] Boreale M., De Incola R., Pugliese R. Proof techniques for cryptographic processes. SIAM J. Comput., 31(3), 2002. Pp. 947-986.

[7] Brassard J. Modern Cryptology. Springer-Verlag, Berlin - Heidelberg, 1988. - 107 p.

[8] Cheminod M., Cibrario Bertolotti I., Durante L., Sisto R., Valenzano A. Tools for cryptographic protocols analysis: A technical and experimental comparison // Computer Standards & Interfaces, 2008.

[9] CLN // Available at: http://www.ginac.de/CLN/ 06.02.2007

[10] CRAMM V Official website // Siemens Enterprise Communications Limited 2006. Available at: www.cramm.com

[11] Digital Security: GRIF //Available: http://www.dsec.ru/products/grif/

[12] ElGamal T. A Public-Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms // IEEE Transactions on Information Theory, v. IT-31, n. 4, 1985. P. 469-472

[13] Garvan F. The Maple Book. Chapman & Hall/CRC, 2001. - 496 p.

[14] GMP // Available at: http://gmplib.org/ 06.02.2007

[15] Hriţcu C., Goriac I., Gordân R. M., Erbiceanu E. MpNT: Designing a Multiprecision Number Theory Library. Faculty of Computer Science, "Alexandru Ioan Cuza" University, Iasi, 2003.

[16] Ivanov V.P. Mathematical evaluation of information protection from unauthorized access // "Specialnaya tekhnika". 2004, N 1. –Pp. 58-64. (in Russian).

[17] Kerckhoffs A. La cryptographie militaire // Journal des sciences militaires, vol. IX. P. 5-38, Jan. 1883, (P. 161-191, Feb. 1883).

[18] Kruschwitz L., Loeffler A. Discounted Cash Flow: A Theory of the Valuation of Firms (The Wiley Finance Series). Wiley, 2005. 178 p.

[19] NTL: Library for doing Number Theory. Available at: http://www.shoup.net/ntl/ 06.02.2008

[20] LiDIA // Available at: http://www.cdc.informatik.tu-darmstadt.de/TI/LiDIA/ 06.02.2007

[21] LIP // Available at: http://www.win.tue.nl/~klenstra/ 06.02.2007

[22] Menichelli F., Menicocci R., Olivieri M., Trifiletti A. High-Level Side-Channel Attack Modeling and Simulation for Security-Critical Systems on Chips // IEEE Transactions on Dependable and Secure Computing,Volume: 5, Issue: 3, July-Sept. 2008, Pp. 164-176.

[23] RIJNDAEL description. Submission to NIST by Joan Daemen, Vincent Rijmen // Available at http://csrc.nist.gov/encryption/aes/round1/docs.htm

[24] RiskWatch Official website // RiskWatch, Inc. Available at: http://www.riskwatch.com/

[25] Rivest R.L., Shamir A., Adleman L.M. A Method for Obtaining Digital Signatures and Public Key Cryptosystems// Communications of the ACM, v. 21, n. 2, February 1978. P. 120-126.

[26] Savelieva A. Formal methods and tools for evaluating cryptographic systems security // St. Petersburg, ISP RAS, In Proceedings of the Second Spring Young Researchers' Colloquium on Software Engineering (SYRCoSE'2008), 2008, Vol 1. ISBN 978-5-91474-006-8. Pp. 33-36.

[27] Schneier B. Beyond Fear. Thinking Sensibly about Security in an Uncertain World. Copernicus Books (September 2003)

[28] Verma R. Protocol Specification and Verification. Lectures on COSC 6397 – Information Assurance. University of Houston, 2006. Available at: www2.cs.uh.edu/~rmverma/M2L1.ppt

[29] Wolfram S. The Mathematica Book, 4th edition. Cambridge University Press and Wolfram Media, Cambridge, 1999, 1470 p.

[30] Zhou Y., Feng D. Side-Channel Attacks: Ten Years After Its Publication and the Impacts on Cryptographic Module Security Testing // Physical Security Testing Workshop (Hawaii, September 26-29, 2005. Available at: http://eprint.iacr.org/2005/388.pdf

Manuscript received March 15, 2009.

Alexandra A. Savelieva is a first-year postgraduate student of Software Engineering Department, State University – Higher School of Economics (e-mail: alexandra.savelieva@gmail.com; optional phone: +7(962)902-4310).

Research supervised by Prof. Sergey M. Avdoshin, Head of Software Engineering Department, State University – Higher School of Economics, Russia (e-mail: savdoshin@hse.ru; optional phone: +7(495)771-3238).