Imed EL FRAY[1, 2], Jerzy PEJAŚ[2]

Warsaw University of Life Sciences, Faculty of Applied Informatics and Mathematics (1)
West Pomeranian University of Technology, Faculty of Computer Science (2)

doi:10.15199/48.2017.01.22

A graph-based risk assessment and prediction in IT systems

***Abstract**. In this paper we propose a graph-based model for the description of a risk analysis process and a risk calculation in IT systems. Our model based on a graph (Graph Risk model in short) is compliant with international standards and recommendations. The graph model for risk assessment includes the best practices in IT Governance. Based on the proposed model there is possible to describe the structure of the security system, its components and relations, and then to make risk calculations for each embedded component of the evaluated system. Moreover, it allows to compare results obtained using different risk analysis method in the context of compliance with standards and recommendations. A simple example given at the end of this paper illustrates how to apply the proposed model to describe the security system and calculate its final risk values.*

***Streszczenie.** W artykule zaproponowano oparty na teorii grafów model opisu procesu analizy i oceny ryzyka w systemach informatycznych. Model ten (w skrócie model grafu ryzyk) jest zgodny z międzynarodowymi standardami i zaleceniami. Model grafowy do szacowania ryzyka budowany jest w oparciu o dobre praktyki z zakresu zarządzania IT. W oparciu o zaproponowany model możliwe jest opisanie struktury systemu bezpieczeństwa, jego komponentów i ich relacji, a następnie oszacowanie ryzyka każdego komponentu ocenianego systemu. Ponadto, proponowany model pozwala na porównanie wyników z innymi wynikami uzyskanymi za pomocą metody analizy ryzyka „FoMRA" w kontekście zgodności z normami i zaleceniami. Na końcu niniejszego artykułu podany jest przykład pokazujący, jak zastosować proponowany model do opisania systemu bezpieczeństwa SI i obliczyć jego wagę ryzyka. (**Model teorii grafów do szacowania ryzyka w systemach informatycznych**).*

Keywords: risk analysis, risk management, risk analysis methods, graph-based model.
Słowa kluczowe: analiza ryzyka, zarządzania ryzykiem, metody analizy ryzyka, modele grafowe.

Wstęp

Risk management and security management in IT systems became during the last few years an important subject of interest in almost all countries in the world. At the field of security analysis and management several ISO/IEC standards [1, 2, 3, 4, 5] have been established during the last 10 years. Along with the standards, some methods enabling risk analysis and management have been developed. Over 200 worldwide methods exist, however over 80% of them is not practically used or are publicly unavailable. Among other just only several are worldwide accepted. Their acceptance is coming out from the compatibility with standards, recommendations, etc. and updating and support activities.

The approach proposed by the Authors is based on creation of a graph consisting of constant number of vertices and edges. As a consequence, all operations and calculations will be performed inside the vertices, what is important for the calculation performance. That graph is built on the base of models used by the most of organizations applying quantitative and qualitative methods. A combination of both approaches could enable to perform risk analysis for complex systems. The obtained results will be compared to previous ones.

Risk Analysis Methods

On the base of ISO standards many quantitative and qualitative methods [6,7,8,9,10,11,12,13,14,15,16,17] for support of risk analysis and evaluation have been developed. Most of them are based on know-how solutions designed for the usage in government and public applications and have been designed by independent or government-supported organizations in many countries. Currently only the few of those methods are word-widely used (OCTAVE [16], CRAMM [14], MEHARI [11]) and they are available as commercial software and theoretical risk analysis support as well. The most of methods mentioned above are created by experts‟ teams active at given IT security areas and they are not supported by proves based on formal mathematical models.

A general advantage of quantitative and qualitative methods is the possibility to order risks according to priorities of their occurrence. These methods present the risk precisely and are effective in an identification of particular threats. They point areas of increased risk in relatively short time periods. Additionally, in the case of quantitative methods, due to the possibility of determination of incident's consequences by number values (financial measures are usually used), it is easy to perform worthwhileness analysis of countermeasures implementation in an organization during an indication of countermeasures.

One of disadvantage of qualitative and quantitative methods is the lack of possibility of a direct comparison of results generated by two different methods. Both methods require qualified staff and are labour-consuming. The another disadvantage in the case of methods based on mathematical models is the significant subjectivity in determination of some indices (e.g. assets values, threats and vulnerabilities weights, etc.). In addition, models presented in [7.8.9] do not predict to take into account relations between more frequent events with small impact and rare events with great influence on system security.

Methods Based on Tree Structures

Methods using tree structures [18, 19, 20, 21, 22, 23] are static. Events causing an occurrence of undesirable incidents in an organization are modelled logically.

The principal advantage of methods based on tree structures is that, due to the graphical presentation, they present in a simple way events leading to incidents occurrence. Such a representation helps to understand the whole investigated system functionality. An analysis can be performed quantitatively, qualitatively or combining both aspects (due to their advantages). These methods are especially useful in the case of complex systems.

The main disadvantage of methods based on tree structures is their static way of activity. It makes hard an evaluation of systems with cross-dependencies of events. An additional problem is the modelling of human behaviour.

Dynamic Analysis Method

Dynamic analysis methods [24] and Graphs [25, 26, 27], in contradiction to static methods using events tree or errors tree analysis, enable to consider dynamic scenarios. The usage of graph method enables to indicate an individual point or a pair of them responsible for a terror occurrence.

The most important examples are Graphs, Digraphs and Markov's models method [28].

The principal advantage of dynamic analysis method is the possibility to consider dynamic scenarios taking into account changes of system behaviour and human reaction for these changes. Graphical representation enables an easy understanding of system activity.

The disadvantage of these methods is the requirement of a good knowledge concerning the details of investigated systems. Due to complex calculations they are used for evaluation of small systems. High calculation power is required and the growth of system complexity increases the risk analysis time.

Description of a Formal Model of Risk Analysis

A formal model of information system "*IS*" is presented below. Mathematical structures specified below are created for the purpose of a precise definition of a calculation graph for risk values and an algorithm for its construction according to FoMRA model [6].

A formal definition of a graph *G* representing a process of an *IS* risk calculation is given below.

Definition 1. Let $G=(Ve,E)$ be a directed graph, where *Ve* - a set of vertexes and $E \subseteq Ve \times Ve = \{(u,v): u,v \in Ve\}$ - asset of edges. Additionally let us assume there exists such a function τ: $V_e \to T$, that any vertex has assigned the type belonging to a set *T* = { *SEC* , *IS*, *A, V, Th, S. DP*, $DP_{subclass}$, *DI*, $DI_{subclass}$, *SS*, *CM*,} where: *SEC* – a security system, *IS* – an information system, *A* means an asset, *V* – a vulnerability, *Th* – a threat, *S* – a risk scenario, *DP* – a measures reducing a potentiality, *DI* – a measures reducing an impact, $DP_{subclass}$ and $DI_{subclass}$ are definable vocabularies of subclasses of type *DP* and *DI* countermeasures , *SS* – a security service, and *CM* – a countermeasure. In such a case the graph *G* will be considered as a process of a system risk calculation.

Note 1. A description of any vertex *v* belonging to a set V_e of a graph G has a form <*verName, type, parameters, [formula, values]*>, where $verName \in V_e$ is a unique vertex name, $type \in T$ is its type, *parameters* – values of vertex parameters (unchangeable values), *formula* – determines a mathematical formula used for the calculation of *values* for a vertex. Elements *formula* and *values* are optional and are present for *constructed* vertexes only (see below). Further it is assumed that particular elements of the vertex description will be referenced as follows: *v.elem*, where v∈Ve, and *elem*∈(*verName, type, parameters, formula, values*). If elements *parameters, formula* or *values* will include more than one component, then references for particular components will be placed after the dot. As an example: for a vertex *v* with a description <*laptop, A, parameters=(price, weight)*> a reference for price should have the form: *v.parameters.price*.

Note 2. Definable vocabularies of subclasses $DP_{subclass}$ and $DI_{subclass}$ are sets of types of used measures reducing a potentiality and an impact action implemented for a given risk scenario. If sets of subclasses $DP_{subclass}$ and $DI_{subclass}$ have the forms $DP_{subclass}$ = {*dp_1*, *dp_2*,…, *dp_m*} and $DI_{subclass}$ = {*di_1*, *di_2*,…, *di_m*}, then descriptions $DP_{subclass}$.*dp_i* or $DI_{subclass}$.*di_j* mean an appropriate subclass assigned to a vertex of the graph *G*.

Definition 2. A set $Ve : X = \{v \in Ve : \tau(v) = X \wedge X \in T\}$ is the set of all vertexes of type $V \in T$ belonging to the graph $G=(Ve,E)$.

Every vertex of the graph G has assigned the formula and/or the parameters. Vertexes with assigned formula will be called further *constructed* vertexes, and consequently vertexes without formula will be called *primitive* vertexes. It should be noticed that the value of a *constructed* vertex is a result of calculations made on the base of the formula assigned.

Functions assigning values to primitive vertexes

Functions assigning values for *primitive* vertexes are defined below. There are $value_A$ and $value_V$ and $value_{CM}$ functions.

A function $value_A$ determines ***a value*** of a given asset depending on basic security parameters (*Confidentiality, Integrity, Availability "CIA"*) and additional ones, such as and (*Authenticity, Non Repudiation "Aut, NR"*)[1]:

(1) $value_A : (Ve : A) \to (\Re^*)^5$

This function can be set by use of an array, with a number of rows equal to a number of assets and five columns, where values from *i*-th row are values of an asset a_i depending on *CIA*, *Aut* and *NR* parameters. When this value is not defined for some of parameters *CIA, Aut* or *NR*, then zero value is assumed.

It is obvious that system assets can be exposed on some threats. In reality a given threat can be realized if and only if this asset has a relevant vulnerability. A natural vulnerability (so called "a natural exposure", independent from security measures used) can be the result of "force majeure" events, unintended and intended actions.

Another function $value_V$ defines a value of a given natural exposure depending on parameters (*Accident, Error, Voluntary "AEV"*):

(2) $value_{Th} : (Ve : Th) \to (\Re^*)^3$

This function, similarly like a previous one, can be set by a relevant array.

It is obvious that any system has some countermeasures reducing potentialities and impacts of threats influencing on IS system assets. Depending on these threats the appropriate measures will be used to reduce a given potentiality or impact. The values of those measures are defined below as follows:

(3) $value_{CM} : (Ve : CM) \to \Re^*$

Functions assigning formulas to constructed vertexes

Constructed vertexes are the ones of type SEC, *IS*, *S. DP, DI*, *SS*, $\text{DP}_{\text{subclass}}$ and $DI_{subclass}$. Let us assume that a set of formulas *F* is given. Then a set of functions assigning formulas to *constructed* vertexes of type *IS*, *S. DP, DI* and *SS* has the form:

(4) $value_X : (Ve : X) \to F, \quad for\ X = SEC, IS, S, DP, DI, SS$

The last two sets of functions $value_{DP_{subclass.dp\_i}}$ and $value_{DI_{subclass..di\_j}}$ assign values to vertexes with types belonging to subclasses $DP_{subclass}$ and $DI_{subclass}$:

(5) $value_{DP_{subclass.dp\_i}} : (Ve : DP_{subclass}.dp\_i) \to F, \ i = 1, \ldots, n$

[1] $\Re^* = [0, +\infty)$, the notation X^5 is understood as a fivefold Cartesian product of a given set with itself.

(6) $$value_{DI_{subclass.dp\_i}} : (Ve : DI_{subclass}.di\_j) \rightarrow F, \quad j = 1, \ldots, n$$

Both sets of functions are elements of the vocabularies $DP_{subclass}$ and $DI_{subclass}$.

The set of formulas *F* can have a form of simple mathematical functions, programme functions or *constructed* computing formulas. It depends on an analysis method used.

Definition 3. ***Security system*** SEC_{IS} ***of a system*** is a pair (*incd_v:SEC*, *formula_v:SEC*), where *incd_v:SEC* is a set of vertexes of *IS* type which are incident with a vertex *v* of *SEC* type,

(7) $$incd\_v : SEC = \{u \in Ve : ((u,v) \in E \wedge (\tau(v) = SEC)) \wedge (\tau(u) = IS)\}$$

such that vertexes of S type are incident with vertexes of *IS* type belonging to a set *incd_v:SEC*:

(8) $$incd\_v : IS = \{u \in Ve : ((u,v) \in E \wedge (v \in incd\_v : SEC)) \wedge (\tau(u) = S)\}$$

Definition 4. ***Information system*** IS is a subgraph $G_{IS}=(Ve_{IS}, E_{IS})$ of a graphG such that $Ve_{IS} \subseteq Ve$, $E_{IS} \subseteq E$, at least one vertex of *IS* type belongs to a set of vertexes of subgraph G_{IS} and there is a subset of vertexes of S type belonging to the set Ve_{IS}, i.e. $Ve_{IS}:S \subseteq Ve:S$.

(9) $$IS\{G_{IS} \subseteq G : Ve_{IS} \in Ve, E_{IS} \in E, \exists v \in Ve_{IS}(\tau(v) = IS) \wedge \exists Ve_{IS} : S\,(Ve_{IS} : S \subseteq Ve : S)\}$$

According to this definition, an information system IS is some set of assets with defined values, which could have vulnerabilities potentially used by threats.

Definition 5. ***General risk scenario of a system*** is a pair (*incd_v:S*, *formula_v:S*), where *incd_v:S* is a set of vertexes of types *A*, *T*, *V*, *DP* and *DI* are incident with a vertex *v* of *S* type,

(10) $$incd\_v : S = \{u \in Ve : ((u,v) \in E \wedge (\tau(v) = S)) \wedge (\tau(u) = A \vee T \vee V \vee DP \vee DI)\}$$

such that vertexes of types $DP_{subclass}$ and *V*, as well as of types $DI_{subclass}$ and *A*, are incident with vertexes of types *DP* and *DI*, respectively, belonging to a set *incd_v:S*:

(11) $$incd\_v : DP = \{u \in Ve : \begin{pmatrix} (u,v) \in E \wedge (\tau(v) = DP) \wedge \\ (v \in incd\_v : S) \end{pmatrix} \wedge ((\tau(u) \in DP_{subclass}) \vee (\tau(u) = Th))\}$$

(12) $$incd\_v : DI = \{u \in Ve : \begin{pmatrix} (u,v) \in E \wedge (\tau(v) = DI) \wedge \\ (v \in incd\_v : S) \end{pmatrix} \wedge ((\tau(u) \in DI_{subclass}) \vee (\tau(u) = A))\}$$

with those vertexes some vertexes of *SS* type are incident:

(13) $$incd\_v : SS = \{u \in Ve : \begin{pmatrix} (u,v) \in E \wedge \\ \begin{pmatrix} v \in incd\_v : DP_{subclass} \vee \\ v \in incd\_v : DI_{subclass} \end{pmatrix} \end{pmatrix} \wedge (\tau(u) = SS)\}$$

and the last ones are incident with vertexes of *CM* type:

(14) $$incd\_v : CM = \{u \in Ve : \begin{pmatrix} (u,v) \in E \wedge \\ (v \in incd\_v : SD) \end{pmatrix} \wedge (\tau(u) = CM)\}$$

A notation *formula_v:S* is the formula assigned to a vertex S by means of $value_S$ function (see: eq. 3.4) and dependent on values of vertexes incident with a vertex *v*:S.

Definition 6. General potentiality action (an action associated with measures implemented in an organization and reducing a probability of vulnerability) of IS is a pair (*incd_v:DP*, *formula_v:DP*), where *incd_v:DP* is a set of vertexes of types belonging to the set $DP_{subclass}$ and vertexes of *V* type incident with a vertex *v* of *DP* type,

(15) $$incd\_v : DP = \{u \in Ve : ((u,v) \in E \wedge (\tau(v) = DP)) \wedge ((\tau(u) \in DP_{subclass}) \vee (\tau(u) = Th))\}$$

and *formula_v:DP* is a formula assigned to a vertex of *DP* type by means of $value_{DP}$ function (see: eq. (4)) and dependent on values of vertexes incident with a vertex *v:DP*.

This pair defines potentiality actions implemented for a given risk scenario. These actions can be preventive ones (e.g. according to [8]) or preventive and dissuasive (e.g. according to [6]).

A general impact action is defined in similar way as general potentiality action mentioned above.

Definition 7. General impact action (an action associated with measures implemented in an organization and reducing an impact) of IS is a pair (*incd_v:DI*, *formula_v:DI*), where *incd_v:DI* is a set of vertexes of types belonging to the set $DI_{subclass}$ and vertexes of type *DI* incident to the *v* vertex of type DI,

(16) $$incd\_v : DI = \{u \in Ve : ((u,v) \in E \wedge (\tau(v) = DI)) \wedge ((\tau(u) \in DI_{subclass}) \vee (\tau(u) = A))\}$$

and *formula_v:DI* is a formula assigned to a vertex of *DI* type by means of $value_{DI}$ function (see eq. (4)) and dependent on values of vertexes incident with a vertex *v:DI*.

This set defines impact actions implemented for a given risk scenario. These actions can be *detective* and *corrective* ones (e.g. according to [8]) or *protective*, *palliative* (e.g. according to [6]).

In similar manner dedicated potentiality and impact actions can be defined in a risk analysis graph *G* for countermeasures various subclasses $DP_{subclass}$ and $DI_{subclass}$, and the security service and countermeasures as well. These definitions are omitted. Although it should be noticed that these definitions together with Definition 3, 4, 5 and 6 define unambiguously a structure of a graph and relations of its vertexes.

The graph presented on Figure 2 shows all activities required for risk value calculations in a security system compliant with ISO/IEC 270xx standards.

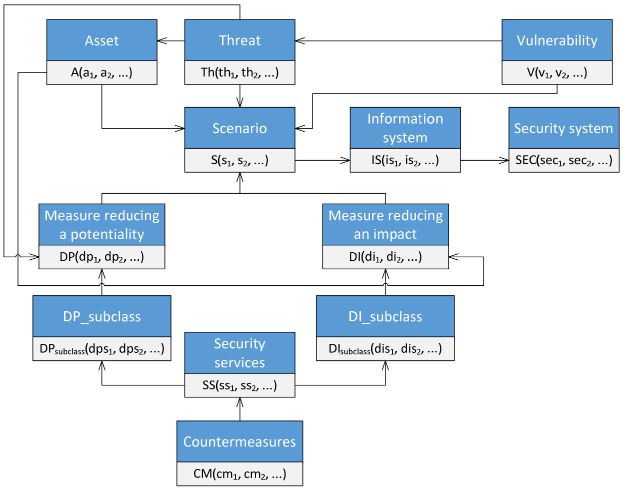

Fig. 1. General structure of a risk calculation graph

Results and Discussions

Accordance of obtained graphs with FoMRA method and performance of calculations for simple and complex systems have been verified. An example of results is given below. For the purpose of this paper the example is cut to one asset only (source code). According to the algorithm presented in Section 3, the $value_A(a_1) = (4,3,2,2,-)$ depend on security parameters *CIA* and *AUT*, as well. Let us consider the set of threats *Th* containing *th₁,th₂∈Th*. By a threats *th* we mean „Intentional erasure (direct or indirect), theft or destruction of program or data containers" with $value_{Th}(th_1) = (0,0,3)$ and *th* "diversion of program source code" with $value_{Th}(th_2) = (0,0,2)$. Let th_1, th_2 be a threats function which meets $th_1, th_2 \in X$ for some $X \in T$.

Additionally, we consider some set *V* of vulnerabilities, where v∈V. A vulnerability *v* is „ No visitor sign-in or escort required for building access".

Let S be the set of risks scenarios, where $s \in S$ and s be a scenarios function, where $s \in T$. Further we consider three scenario, namely s_1, s_2, s_3, that corresponds to „ theft or erasure of removable media containing application source code within the IT premises, by an authorized visitor, theft of archive tapes of programs within the media storage premises, by a non authorized visitor and erasure of archive data files by operational personnel".

The success of a given scenario depends on implemented security measures $DP_{subclass}$ and $DI_{subclass}$ reducing the potentiality and impact related to this scenario (see Sections 3). The security measures depend on $DP_{subclass}$ and $DI_{subclass}$, *SS* security services (for example for s_1 (- Surveillance of sensitive locations, Backup of system software and applications, - Insurance of consequential losses) and *CM* countermeasures for surveillance of sensitive locations (a video surveillance system with possibility to keep records for a long period, a surveillance team, sufficient resources to cover the eventuality of multiple alarms, procedures for surveillance and intervention).

Table 1 Calculated values of simple and complex vertexes

Scen	*value_A*	*value_V*	*Value_m:DPsubclassdp-i*			*Value_m:DIsubclassdp-i*			*value_v:DP*	*value_v:DI*	*value_z:S*	*Value_h:IS*
			dp_1	*dp_2*	*dp_3*	*di_1*	*di_2*	*di_1*				
S_1	4 (A)	3 (V)	3	3	1	1	2	2	3	3	3	3
S_2	4 (A)	3 (V)	3	2	2	2	3	2	3	3	3	3
S_3	4 (A)	2 (V)	2	1	1	1	3	2	2	2	2	2

Results presented in the Tab. 1 are compliant with results obtained using FoMRA method. In the case of simultaneous calculations for few scenarios a Graph method was a little bit faster. Increased number of assets, scenarios and complexity of a system significantly improved performance in comparison with calculations performed according to the traditional Mehari method.

Values $value_{z:S} = value_{h:IS}$, because only one system was examined. For some of the scenarios, values *DP*$_{subclass}$ and *DI* $_{subclass}$ have assigned 1, as it is in Tab. 1. This figure shows that countermeasures are not applied when it is not possible to be protected from certain events. The results in Tab. 1 reflect the results obtained using Mehari method. In the case of calculation for several scenarios, risk calculation speed is significantly faster in the case of the graph-based model. When the number of assets, scenarios, etc. is increasing, then calculation speed in the proposed model also increase, but faster than calculation speed of the traditional Mehari method. The difference is because, the calculation cycle in the case of FoMRA is repeated for each scenario from the beginning to the end. In the case of the graph-based model the procedure uses results from calculations performed earlier (the same calculations are not repeated).

Calculation time for both approaches (Graph, FoMRA) is presented on the Fig. 2.

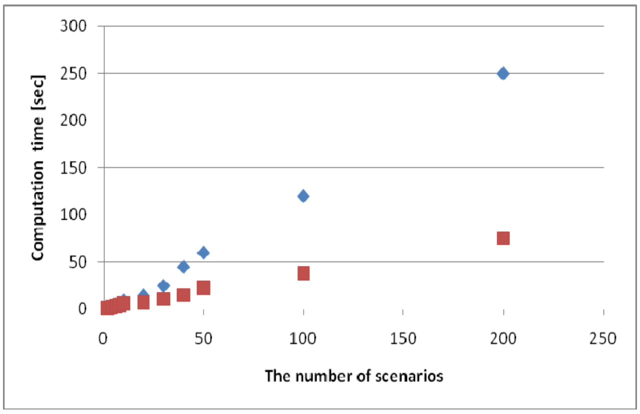

Fig. 2. The number of studied scenarios in dependence of time (■ - computing time for FoMRA method, ♦ - computing time for proposed method).

Summary

The presented Graph is compliant with the family of standards 270xx. This Graph seems to be useful for a future comparison of results obtained with its usage with the ones obtained using methods and approaches presented in Section 2.1.

Significant advantages are that it is universal and open Graph. The most of methods used, e.g. CRAMM, COBRA, FoMRA, MEHARI mentioned above, assume *a priori* some values and constants. Additionally, the presented Graph allows a free selection of any scheme for an assets classification, potentialities, impacts, risks, etc. It allows to classify assets considering different security parameters. Further works will be focused on an extension of the graph with additional vertexes allowing to evaluate an influence of implemented countermeasures, assets state changes, threats and vulnerabilities without the necessity of risk analysis repetitions, what should lead to the significant reduction of costs and time.

***Authors**: dr hab. inż. Imed El Fray, Warsaw University of Life Sciences, Faculty of Applied Informatics and Mathematics, ul. Nowoursynowska 159, 02-776 Warszawa. Imed_el_fray@sggw.pl*
dr hab. inż. Jerzy Pejaś, Zachodniopomorski Uniwersytet Technologiczny w Szczecinie, Katedra Inżynierii Oprogramowania, ul. Żołnierska 52, 71-210 Szczecin E-mail: {ielfray, jpejas@wi.zut.edu.pl}

REFERENCES

[1] PN-ISO/IEC 13335-1 Information technology -- Security techniques -- Management of information and communications

technology security -- Part 1: Concepts and models for information and communications technology security management, :1999
[2] PN-ISO/IEC 27002 Information technology -- Security techniques -- Code of practice for information security management, 2014
[3] PN-ISO/IEC 27001 Information technology -- Security techniques -- Information security management systems – Requirements, 2014
[4] ISO/IEC 27003 Information technology – Security techniques – Information security management system implementation guidance, 2010
[5] ISO/IEC 27005 Information technology -- Security techniques -- Information security risk management, 2014
[6] El Fray, I., A comparative study of risk assessment methods, MEHARI & CRAMM with a new formal model of risk assessment (FoMRA) in information systems. Computer Information Systems and Industrial Management. Springer Berlin Heidelberg, (2012), 428-442
[7] Federal Information Processing Standard (FIPS) 65, Guideline for Automatic Data Processing Risk Analysis, National Bureau of Standard, US 1979
[8] R. Baskerville: Information Systems Security Design Methods: Implications for Information Systems Development, Computing Surveys 25 (4), 1993, 375-414
[9] D. B. Parker, Computer Security Management, Reston Publishing Company Inc., Reston VA, 1981
[10] Méthodologie d'Analyse des Risques Informatique et d'Optimation par Niveau « MARION », CLUSIF, France 1998
[11] Méthode Harmonisée d'Analyse de Risques « MEHARI » CLUSIF, France 2007
[12] Microsoft Security Assessment Tool « MSAT », http://technet. Microsoft.com/en-us/security/cc18512.aspx
[13] G. Stoneburner, A. Goguen, A. Feringa, Risk Management Guide for Information Technology Systems, Recommendations of the National Institute of Standards and Technology, Technology Administration. U.S. Department of Commerce, Special Publication 800-30, Washington DC 2002
[14] CCTA Risk Analysis and Management Method « CRAMM », Central Computing and Telecommunications Agency, United Kingdom Government, UK 1987
[15] Consultative, Objective and Bi-functional *Risk Analysis* « COBRA », Disaster Recovery Planning Group, UK 1991
[16] Operationally Critical Threat, Asset, and Vulnerability Evaluation « OCTAVE » Carnegie Mellon University, US 2006
[17] Control Objectives for Information and related Technology « COBIT » ISACA, IT Governance Institute, US 2007
[18] K. Pickard, P. Muller, B. Bertsche, Multiple failure mode and effects analysis-an approach to risk assessment of multiple failures with FMEA. IEEE Xplore, 2005
[19] R. Ferdous, F .I. Khan, B. Veitch, P. R. Amyotte: Methodology for computer-aided faulttreeanalysis, Elsevier, 85 (1), 2007, 70-80
[20] M. Rausand: System Analysis Event Tree Analysis, 2005
[21] J.D. Andrews, L.M. Ridley: Application of the cause-consequence diagram method to static systems, Elsevier, 75 (1), 2002, 47-58
[22] M. Ferjencik, R. Kuracina: Mort worksheet or how to make mort analysis easy, Elsevier, 151 (1), 2008, 143-154
[23] Nri mort user's manual second edition, 2009
[24] A. HakobyanT. Aldemir, R. Denning, S. Dunagan, D. Kunsman, B. Rutt, U. Katalyurek: Dynamic generation of accident progression event trees, Elsevier, 238 (12), 2008, 3457-3467
[25] M. Bartlett, E. E. Hurdle, E. M. Kelly: Integrated system fault diagnostics utilising digraph and fault tree-based approaches, Elsevier, 94 (6), 2009, 1107-1115
[26] J. Kontio: Risk management, compliance and competitive advantage: process, responsibilities and stakeholders", 2008
[27] S. M. Jacoub, H. H. Ammar: A methodology for architectural-level reliability risk analysis, IEEE Transctions on Softwre Engineering, 28 (6), 2002, 529-547
[28] Reliability/availability of electrical & mechanical systems for command, control, communications, computer, intelligence, surveillance and reconnaissance (c4isr) facilities, 200727