DE GRUYTER OPEN

Int. J. Appl. Math. Comput. Sci., 2014, Vol. 24, No. 1, 213–225
DOI: 10.2478/amcs-2014-0016

amcs

A NEW LIGHTWEIGHT METHOD FOR SECURITY RISK ASSESSMENT BASED ON FUZZY COGNITIVE MAPS

PIOTR SZWED, PAWEŁ SKRZYŃSKI

Department of Applied Computer Science
AGH University of Science and Technology, al. Mickiewicza 30, 30-059 Cracow, Poland
e-mail: {pszwed,skrzynia}@agh.edu.pl

For contemporary software systems, security is considered to be a key quality factor and the analysis of IT security risk becomes an indispensable stage during software deployment. However, performing risk assessment according to methodologies and standards issued for the public sector or large institutions can be too costly and time consuming. Current business practice tends to circumvent risk assessment by defining sets of standard safeguards and applying them to all developed systems. This leads to a substantial gap: threats are not re-evaluated for particular systems and the selection of security functions is not based on risk models. This paper discusses a new lightweight risk assessment method aimed at filling this gap. In this proposal, Fuzzy Cognitive Maps (FCMs) are used to capture dependencies between assets, and FCM-based reasoning is performed to calculate risks. An application of the method is studied using an example of an e-health system providing remote telemonitoring, data storage and teleconsultation services. Lessons learned indicate that the proposed method is an efficient and low-cost approach, giving instantaneous feedback and enabling reasoning on the effectiveness of the security system.

Keywords: security, risk assessment, telemedicine, fuzzy cognitive maps.

1. Introduction

With the proliferation of Internet based technologies, most newly developed IT systems provide remote access using a public networking infrastructure. This brings obvious business benefits and opportunities, but also increases vulnerability to new types of threats. In consequence, security related issues start to play an important role in a software lifecycle and analysis of IT security risk is often considered to be an indispensable stage during software deployment, often expressed in requirements.

The explosion of new technologies coincided with putting the focus on productivity, cost reduction and moving towards an agile development. Changes observed in the last few years were weakly followed by the evolution of risk assessment methodologies and standards, whose roots date back to the 1980s. These methodologies are often issued by public agencies and dedicated to the public sector or large institutions. Performing security risk assessment inline with these methodologies can be an immense effort that would hinder the potential benefits of agility.

Current business practice often tends to reject risk evaluation based on standard models comprised of threats and their likelihoods, vulnerabilities and estimated losses, by replacing them with best practices, assets valuation and scenario based analyses. The first two approaches simply relate safeguards to types of assets or their value within an organization, the last relies on testing and analyzing well known failure scenarios.

This leads to a substantial gap: threats are not re-evaluated for particular systems (because it is too costly) and a selection of security functions is not based on risk models, but on lists of safeguards, which are prepared according to company best practices.

This paper proposes a new *lightweight* risk assessment method that aims at filling the gap between heavy risk assessment methodologies and agile business practices. The method involves building risk models and performing a risk calculation based on the Fuzzy Cognitive Maps (FCMs) approach. FCMs are used to capture dependencies between assets, and FCM-based reasoning is applied to aggregate risks assigned to lower-level assets (e.g., hardware, software modules, communications, people) to such high level assets as services, maintained data and processes. An application

of the method is studied on an example of an e-health system providing remote telemonitoring, data storage and teleconsultation services.

The paper is organized as follows. In Section 2 we provide an overview of risk assessment methods. Section 3 introduces fuzzy cognitive maps and is followed by Section 4, in which the risk assessment methodology is described. Further, in Section 5 the analyzed system is presented, then in Section 6 an application of the proposed risk assessment method is discussed. Finally, Section 7 provides concluding remarks.

2. Related works

According to Guttman and Roback (1995) as well as Hoo (2000), *security* is the protection afforded to an information system in order to preserve the integrity of data and system functions, their availability, authenticity and confidentiality.

Risk assessment has its roots in the nuclear power industry, where probabilistic models were built to analyze potentially catastrophic faults in nuclear power facilities (Hoo, 2000). In 1979 the National Bureau of Standards proposed the Annual Loss Expectancy (ALE) metric (Institute for Computer Sciences and Technology, 1979) as applicable for non safety-critical systems. It defined risk as a sum of products of *frequencies* of harmful events and induced *losses* expressed in dollars. This approach to risk characterization influenced many methodologies and standards, e.g., CRAMM[1] or recently NIST 800-30 (Ross, 2011). In some frameworks the statistical term *frequency* is replaced by *likelihood* or *probability*, *loss* by *impact*. Furthermore, as it is difficult to estimate absolute values of probabilities and losses, ordinal scales (low, medium, high) defining coarse levels are used.

In spite of the popularity of the ALE metric, its application to risk assessment is considered problematic due to a cognitive bias in estimating likelihoods of threats (Hubbard and Evans, 2010), lack of statistical data, difficulties in calculating losses and extremely high costs of the whole process.

In numerous standards and methods listed in the ENISA Inventory[2], including most popular: ISO/IEC 27005 (ISO/IEC, 2011), NIST 800-30 (Ross, 2011) and CRAMM, risk assessment is not only perceived as a method of estimating risks; it is rather considered a complex process in the management of IT system security built up of several activities, such as identification of assets, threats and vulnerabilities, likelihoods of their occurrences, potential losses and theoretical effectiveness of security measures. Hence, the standards, apart from defining risk scoring methods, specify organizational foundations for performing risk assessment in the broader context of IT security risk management. It can be observed that risk assessment performed strictly in compliance with a selected standard can be a large and costly endeavor.

Practical implementations of risk assessment and management include various approaches. *Integrated business risk-management frameworks*, e.g., SABSA[3], abstract from technical details and embed IT security within a holistic business risk management context. *Valuation-driven methodologies* ignore difficult to assess likelihoods and simply recommend safeguards using as a sole criterion estimated values of assets. *Scenario analysis approaches* focus on eliciting and evaluating scenarios compromising security. Finally, *best practices* rely on standardized lists of safeguards eligible for given types of assets.

Parallel to business practice, ongoing (mainly academic) efforts aiming at building risk models going beyond ALE and applying them to real or hypothetical systems might be observed. In several cases they were followed by proposals of methodologies or guidelines, often accompanied by dedicated interactive software packages. Furthermore, these guidelines were frequently combined with modeling techniques that are widely applied in reliability and safety engineering, such as fault trees, event trees, Markov chains, and FMEA (Failure Mode Effects Analysis) (Vesely *et al*., 1981; Birolini, 2000; Stamatis, 2003). These techniques provide a representation of system operations and undesirable events, and a validation of the system safety level (Craft *et al*., 1998; Modarres *et al*., 1999; Bowles and Wan, 2001; Stathiakis *et al*., 2003; Cervesato and Meadows, 2003).

Han *et al*. (2004) described an expansible vulnerability model in order to qualitatively assess the security of an active network and active nodes, aiming at solving a problem that is more suited for an active network than a traditional one. Eom *et al*. (2007) introduced a risk assessment method based on asset valuation and quantification. Baudrit *et al*. (2006) proposed a risk assessment method of node transmission and possibility exposure. Sun *et al*. (2006) introduced a risk assessment model based on DS evidence reasoning. The disadvantages of all those methods are related to the strong subjectivity of premises. Hence, Chen (2006) put forward a quantitative hierarchical threat assessment model and a corresponding quantitative calculation method exploiting the statistics of system attacks that occurred in the past. Wang *et al.* (2011) analyze network security by using a probable attack graph generated on the basis of security case reasoning, carrying out qualitative risk assessment for the network system mainly from an

[1] http://www.cramm.com/.

[2] http://rm-inv.enisa.europa.eu/methods/rm_ra_methods.html.

[3] http://www.sabsa-institute.org/the-sabsa-method.

attack perspective.

Within the last couple of years, risk assessment techniques have evolved towards integrating real time and intelligent functions. In particular, great attention has been paid to artificial immunology due to such advantages as self-organization, self-adaptability, diversity and self-learning. Although research results have been applied only to invasion and fault detection, the application in information security risk assessment has just begun (Chiang and Braun, 2007; Peng, 2007).

One of the challenges in risk analysis and management is the identification of relationships between risk factors and risks. The complexity of the method to analyze these relationships, the time to complete the analysis, and the robustness and trustworthiness of the method are important features to be considered.

Attack trees, proposed by Schneier (1999), specify which combinations of adversarial actions should be employed to compromise an asset (the goal of an attack). Hence, a tree with AND-OR nodes represents several attack scenarios. As each tree node can be assigned with various attributes: a probability, a cost of an adversarial action or a loss, various metrics can be calculated indicating the probability of success of a given attack and helping to find potential vulnerabilities. An advantage of the approach is that it allows analyzing the system from an attacker perspective and evaluating the efficiency of the countermeasures applied. Nevertheless, the method has several limitations: it requires a deep knowledge of potential attackers, an assignment of numerical values to tree nodes can be a difficult task, and it is not clear how the model can be linked with the results of a business analysis or an architectural design.

An application of attack trees to assess security risks in heterogeneous telecommunication networks was proposed by Szpyrka *et al*. (2013). The authors introduced a two-stage method in which firstly possible attack scenarios were modeled with attack trees and then an approach based on Bayesian networks was applied to calculate risks for the network elements.

An interesting method for risk assessment applied to the safety of intelligent buildings was described by Mikulik and Zajdel (2009). Being an adaptation of the *formal safety assessment* approach, the method proposes a model based on fuzzy logic in which different habitat factors are taken into consideration to establish the risk profile of a building.

Lazzerini and Mkrtchyan (2011) proposed a method using Extended Fuzzy Cognitive Maps (E-FCMs) to analyze the relationships between risk factors and risks. E-FCMs are suggested by Hagiwara (1992) to represent causal relationships in a more natural way. The main differences between E-FCMs and conventional fuzzy cognitive maps (discussed in Section 3) are the following: E-FCMs have nonlinear membership functions, conditional weights, and time delay weights. In particular, the last feature can be useful when modeling sequences of dependent activities (e.g., occurring during a development of a software product). In their paper, Lazzerini and Mkrtchyan proposed a framework adopting a pessimistic approach to assess the overall risk of a system or a project using E-FCMs. Moreover, they extended E-FCMs by introducing a special graphical representation suitable for risk analysis. The method was applied to software project management. However, their approach is more suitable for project risk analysis, whereas IT security risk falls rather into the operational risk category. Hence, it cannot be directly applied to the problem described in this paper.

An information security risk assessment model and a corresponding risk calculation method, which are based on danger theory were introduced by Zhuang *et al*. (2009). The approach addresses the problem of strong subjectivity and aims at improving accuracy and the real time performance of current information security risk assessment systems, by reference to a dynamic response characteristic of danger theory in immunology.

An application of a new method of risk analysis to an e-health system of monitoring vital signs was discussed by Maglogiannis *et al*. (2006). The method utilized the CRAMM approach for identifying and evaluating assets, threats, and vulnerabilities. The system was considered safety-critical and for the calculation of risks a Bayesian network model, which presented concisely all interactions of undesirable events in the system, was developed. However the method is "heavy" as it requires probability benchmarking, which is difficult to perform and costly. Moreover the method is highly affected by data reliability related to the probability of the occurrence of the undesirable event.

3. Fuzzy cognitive maps

Cognitive maps were first proposed by Axelrod (1976) as a tool for modeling political decisions, then extended by Kosko (1986; 1992) by introducing fuzzy values. A large number of applications of fuzzy cognititive maps was reported, e.g., in project risk modeling (Lazzerini and Mkrtchyan, 2011), crisis management and decision making, analysis of development of economic systems, and the introduction of new technologies (Jetter and Schweinfort, 2011), ecosystem analysis (Ozesmi, 2004), signal processing and decision support in medicine. A survey on fuzzy cognitive maps and their applications can be found in the works of Aguilar (2005) and Papageorgiou (2011).

FCMs are directed graphs whose vertices represent concepts, whereas edges are used to express causal relations between them. A set of concepts $C = \{c_1, \ldots, c_n\}$ appearing in a model encompasses events,

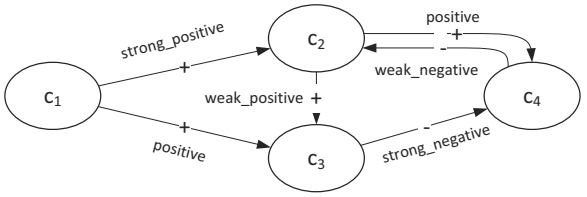

Fig. 1. Example of an FCM graph. Vertices are assigned with concepts, directed arcs with linguistic weights of specifying influence.

conditions or other relevant factors. A system state is an n-dimensional vector of concept activation levels ($n = |C|$) that can be real values belonging to $[0, 1]$ or $[-1, 1]$.

Causal relations between concepts are represented in FCMs by edges and assigned weights. A positive weight of an edge linking two concepts c_i and c_j models a situation where an increase of the level of c_i results in a growing c_j; a negative weight is used to describe the opposite rapport. In the simplest form of FCM, the values from the set $\{-1, 0, 1\}$ are used as weights. They are graphically represented as a minus $(-)$ sign attached to an edge, an absence of edge or a plus $(+)$ sign. While building FCM models, more fine-grained causal relations can be introduced. They are usually specified as linguistic values, e.g., *strong_negative*, *negative*, *medium_negative*, *neutral*, *medium_positive*, *positive*, *strong_positive*, and in a computational model they are mapped on values uniformly distributed over $[-1, 1]$.

Causal relations between concepts in the FCM can be represented by $n \times n$ influence matrix $E = [e_{ij}]$, whose elements e_{ij} are weights assigned to edges linking c_i and c_j, or have 0 values if there is no link between them.

Figure 1 gives an example of an FCM graph whose vertices were assigned with concepts c_1, c_2, c_3 and c_4, whereas the edges were assigned with linguistic weights defining mutual influences. The corresponding E matrix is defined by

$$E = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & -0.33 \\ 0.66 & 0.33 & 0 & 0 \\ 0 & 0.66 & -1 & 0 \end{bmatrix}. \tag{1}$$

The selection of values corresponding to linguistic values is arbitrary; in the example, the values -1, -0.66, -0.33, 0, 0.33, 0.66 and 1 were used.

Reasoning with the FCM consists in building a sequence of states: $\alpha = A(0), A(1), \ldots, A(k), \ldots$, starting from an initial vector of activation levels of concepts. Consecutive elements are calculated according to

$$A_i(k+1) = S_i(\sum_{j=1}^{n} e_{ij}\, A_j(k)). \tag{2}$$

In the $(k+1)$-th iteration the vector $A(k)$ is multiplied by the influence matrix E, then the resulting activation levels of concepts are mapped onto the assumed range by means of an *activation* (or *splashing*) function.

The selection of the activation function depends on assumptions regarding the calculation model, in particular the selected range and the decision to use continuous or discrete values. Multiplication of an n-dimensional square matrix E, both containing elements whose absolute values are bounded by 1, results in a vector having elements in $[-n, n]$. Values from this interval should be mapped by an activation function into the range $[-1, 1]$ (or $[0, 1]$) preserving monotonicity and satisfying $S(0) = 0$ (or $S(0) = 0.5$ in the second case). In further analysis, two activation functions were used:

$$S_{cut}(x) = \begin{cases} -1 & \text{if } x < -1, \\ x & \text{if } -1 \le x \le 1, \\ 1 & \text{if } x > 1, \end{cases} \tag{3}$$

$$S_{exp}(x) = \begin{cases} 1 - \exp(-mx) & \text{if } x \ge 0, \\ -1 + \exp(-mx) & \text{if } x < 0. \end{cases} \tag{4}$$

Basically, a sequence of consecutive states $\alpha = A(0), A(1), \ldots, A(k), \ldots$ is infinite. However, it was shown that after k iterations, where k is a number close to the rank of matrix E, a steady state is reached or a cycle occurs. This observation is not surprising, as it is analogous to properties of Markov chains. Hence, the stop criterion for the reasoning algorithm in the k step is the following:

$$\exists j < k\colon d(A(k), A(j)) < \epsilon, \tag{5}$$

where d is a distance and ϵ a small value, e.g., 10^{-2}.

A sequence of states α can be interpreted in two ways. Firstly, it can be treated as a representation of a dynamic behavior of the modeled system. In this case there exist implicit temporal relations between consecutive system states and the whole sequence describes an evolution of the system in the form of a *scenario*. Under the second interpretation the sequence represents a non-monotonic fuzzy inference process, in which selected elements of a steady state are interpreted as reasoning results. An occurrence of a cycle can be treated as a form of undecidability.

In this paper FCMs are considered to be a tool for risks modeling, and the focus is put on the second approach.

4. Methodology of risk assessment

The methodology for risk assessment comprises basic steps common to various standards and guidelines (see

Guttman and Roback, 1995; ISO/IEC, 2011; Ross, 2011; Landoll, 2005). The salient difference is the use of an FCM model capturing influences between assets and allowing their dependencies to be tracked during a risk aggregation.

4.1. Conceptual model. The assumed conceptual model (Fig. 2) assigns an abstract *utility value* to an *asset* and organizes assets into the *added value tree*, a hierarchical structure in which components of a lower level deliver value to parent elements. The top of the tree is occupied by key processes; they are identified according to business drivers. The utilities of processes depend on data used and invoked services. Various data sources users, sensors and external data providers), contribute to the utility of data. Services depend on software, hardware and communication, but also on involved staff, physical infrastructure (buildings, rooms, electricity) and external services (e.g., Public Key Infrastructure). The dependency relation between assets is depicted in Fig. 2 by open arrows. Closed arrows indicate an underlying type hierarchy used in architectural views, e.g., a *Sensor* is a kind of *Hardware*, but also a *DataSource*.

Utility values assigned to assets can be interpreted as aggregations of various quality attributes: security, reliability, usability, etc. Changes of utility values assigned to lower-level assets influence higher-level components that use them. It should be observed that the tree structure of dependencies between classes of assets results in a lattice of dependencies between instances of assets, e.g., data analysis, data storage and access services depend on the database (software).

The risk model presented in Fig. 3 assumes that the utility of an asset can be compromised by a threat, which decreases its value. In the presented approach we opt for asset-based identification of threats in opposition to approaches focused on adversarial actions or threat agents, e.g., attack trees (Schneier, 1999).

Negative influence of a threat on an asset can be compensated by an appropriate countermeasure. Countermeasures themselves do not add value to the utility, they only reduce the risk. On the other hand, certain IT security components, e.g., LDAP or centralized data access auditing services, can be considered assets and not only countermeasures.

Finally, the problem of defining risk in this setting arises. In many areas of security and safety analysis, the assessed risk is related to possible financial losses, in particular with regard to IT systems developed within financial institutions: banks or assurance companies. Small individual events resulting in losses are accumulated and taken into account, as influencing a risk profile, if they exceed a certain fixed threshold. On the other hand, in safety critical systems, e.g., the monitoring of vital signs, radiotherapy, aerospace

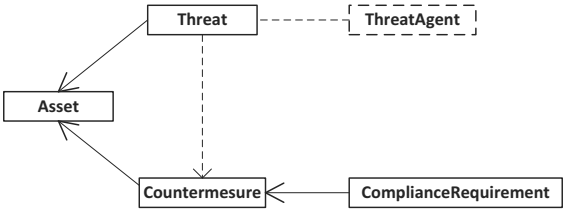

Fig. 3. Relations between assets, threats and countermeasures.

or the railway industry, each system failure is considered an unquantified catastrophic loss, whose occurrence is a condition for rejecting the software during assessment. This is not the case of the e-health system discussed in the next section, which is dedicated to the monitoring of chronic diseases, thus having soft safety requirements. It is also difficult to estimate a financial loss caused by potential failures, as it would require a specification of the business environment in which the system is deployed.

For evaluation purposes, we define

- *utility* assigned to assets as a value from range $[-1, 1]$,
- *risk* related to an asset as the negative difference between the assumed utility and the value calculated at the end of the reasoning process.

The reasoning process takes into account influences of threats and countermeasures directly linked to assets, but also changes in utility resulting from relations captured in the added value tree.

4.2. Risk assessment process. The risk assessment process is shown in Fig. 4. Rounded rectangles represent process steps (activities), rectangles with continuous borders—input or output information, and rectangles marked with a dotted line—internally developed artifacts. The process comprises six steps briefly discussed below.

1. *Identification of assets*. The input for this step is existent documents specifying a system vision, an operational concept and an architecture, but also interviews with designers and development teams. The outcome is a list of assets identifying key processes, services, data, software modules, hardware, communication, providers of external data and services, people involved and physical premises.

2. *Building added value trees*. This step aims at making an assessment of how lower-level assets contribute

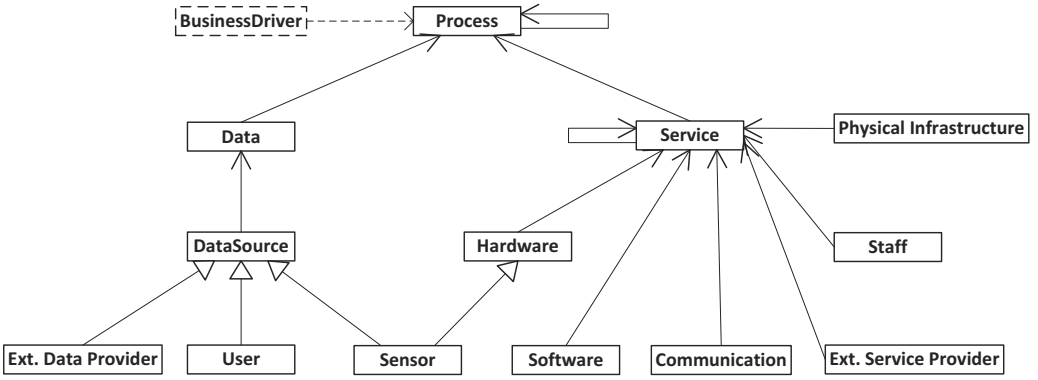

Fig. 2. Classes of assets appearing in an added value tree.

to higher-level ones, e.g., hardware, software and communication channels contribute to services and data, whereas services to processes. Influences are expressed by linguistic values, which are assigned during interviews with software developers and brainstorming sessions. Technically, the obtained added value tree is represented as an FCM influence matrix.

3. *Identification of threats*. For this purpose, a general taxonomy of threats, e.g., an available ontology, can be used and customized to the analyzed case. We use an asset-based model of threats, i.e. we identify threats that are related to a particular asset.

4. *Risk assessment for individual assets*. As a basic tool we use a questionnaire in which various stakeholders involved reply to questions concerning the countermeasures applied. A list of standard countermeasures reflecting the best practices in the field of IT security is used and adapted to a particular set of assets. The outcome of this phase is an assignment of risk values (real numbers normalized to the interval $[0, 1]$) to assets.

5. *Risk aggregation*. This step consists of FCM reasoning aiming at establishing how risks assigned to low-level assets accumulate to yield risk profiles of high-level assets. It also involves preparations required, e.g., normalization of an FCM influence matrix.

6. *Interpretation of results*. In particular, this step may include *what if* analyses, when an application of additional countermeasures at various levels of individual assets is assumed and Step 5 is repeated.

4.3. Discussion. Several methodology steps, in particular those aiming at information gathering, e.g., identification of assets and threats, are also present in various risk assessment methodologies. However, the scope of collected data and analysis tools differs for the methodology discussed and classical "heavy" methods like CRAMM or NIST 800-30.

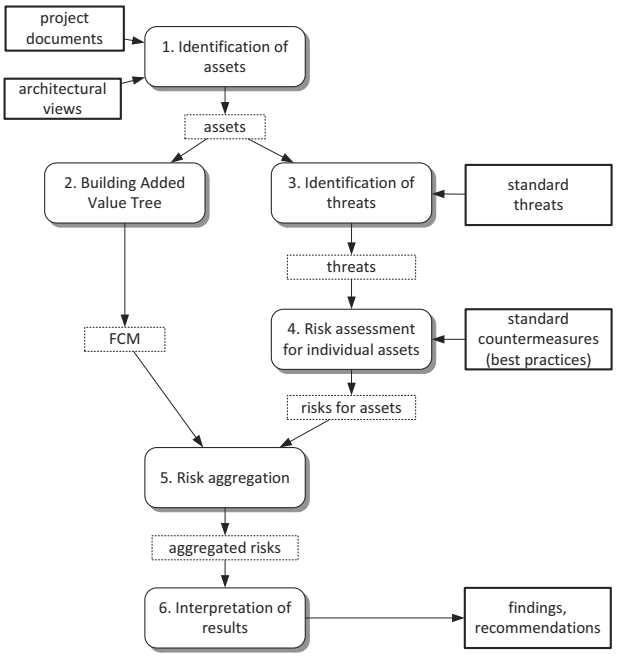

Fig. 4. Risk assessment process.

First of all, methods based on the ALE metric require historical data and benchmarking to assess probabilities of threat occurrences. Such analysis is possible if relevant data are available. In other cases, arbitrary assessments have to be made. A controversial step of other methods is an assessment of potential financial losses caused by threats. This requires an exact knowledge of how a compromised asset is deployed and used within a business model. Moreover, such estimations should combine two types of information: a deep knowledge about the system architecture and a business profile, e.g., the number of

transactions and the average transaction value.

We consider the method lightweight because it meets the following conditions:

- It reuses the data developed as part of a software engineering process by importing information from architectural views. This task can be automated: the authors are currently working on a tool that automatically builds an added value tree from architectural models expressed in the ArchiMate language (The Open Group, 2012).
- Statistical analysis is not required in the method; however, historical data can be implicitly considered best practices: recommended safeguards/countermeasures at assets levels.
- As the method attempts to assess IT security risk in isolation of business environment, an assessment of financial losses, which is usually very hard and error-prone, is not required.

The last feature, however, limits its application to large systems, for which boundaries between business and application layers are blurred. For such systems, comprising several dozen or even hundreds business processes, integrated business risk-management frameworks like SABSA seem to be more appropriate.

5. Presentation of the SWOP system

SWOP is an e-health system dedicated to patients suffering from chronic conditions (*SWOP* is the acronym of the Polish name *System Wspomagania Opieki Przewlekłej*). The main goal of the system is to help patients in self-management of a chronic disease through the monitoring of symptoms, self-assessment, informing about necessary actions when symptom levels indicate a problem, as well as interactions with health care professionals.

On a regular basis, patients manually or automatically send results of self-observations or self-measurements specific for their chronic disease, e.g., hypertension, asthma, diabetes, osteoarthritis. A set of implemented communication modules provides great flexibility in configuring the parameters, the operational modes of sensors and communication channels (WiFi, WAN, GPRS). Communication is always secured with cryptographic protocols provided by TLS (Transport Layer Security). The data entered are stored in a database and automatically analyzed to determine patients' status, trends in the course of the disease and the risk of symptom exacerbation. Then, patients are provided with the results of the assessment, which may be in the form of messages transmitted from the system to the terminal used by a specific patient, e.g., a personal computer or a smartphone (as SMS notifications).

Medical staff are also provided with tools allowing them to configure certain parameters used in medical analyses. The system offers capabilities of asynchronous communication between patients and the personnel providing support to them (virtual carers, leading physicians or other health professionals). If needed, the assistance of a specialist may be requested. Moreover, the system gives an option of transferring patients' data from external HL7-compliant health information systems.

The architecture of the system is presented in Fig. 5. Personal Telemonitoring Devices (sensors) (1) gather raw medical data and transmit them via a Bluetooth interface to Mobile Client Application on Patient's Smartphone (2). After initial validation, the health parameters are sent to SWOP Server. Data are filled out in medical questionnaires available on Patient's Smartphones or in Browser Based Clients and then transmitted to the server using SSL encrypted connection. Certification Server based on Nginx Server (3) receives data and routes them to one of Application Server instances. Application Server (4) is the component where the main system logic resides; it is written in the Python programming language and served via a WSGI interface by Gunicorn Server. Application Server is responsible for authorization, data validation and generation of notifications, as well as communication with Database (5) and Data Analysis (6) servers. Database, hosted on two independent servers in a master–slave configuration, is designed to securely store critical patients' data. Data Analysis Server is responsible for identification of trends in the course of a disease.

Additional information related to the system architecture, the technologies used and particular communication solutions can be found in the works of Szwed (2013), Szwed *et al*. (2013), or Kobylarz and Danda (2013).

6. Risk analysis for the SWOP system

In this section we discuss using the example of SWOP system subsequent steps of the risk analysis process performed according to the methodology defined in Section 4.2.

While selecting the scope of risk analysis, we decided to include three areas: *IT security*, understood as protection against adversarial actions and accidental leak of sensitive data, *business continuity* that can be mapped on such quality attributes as reliability and availability of services, and protection against *operational incidents*, such as errors in entered data or process execution. For a telemedicine system, these can stem from low patient skills, low quality sensors, and unmotivated or untrained staff.

6.1. Identification of assets. The first step of risk assessment was performed as a brainstorming session

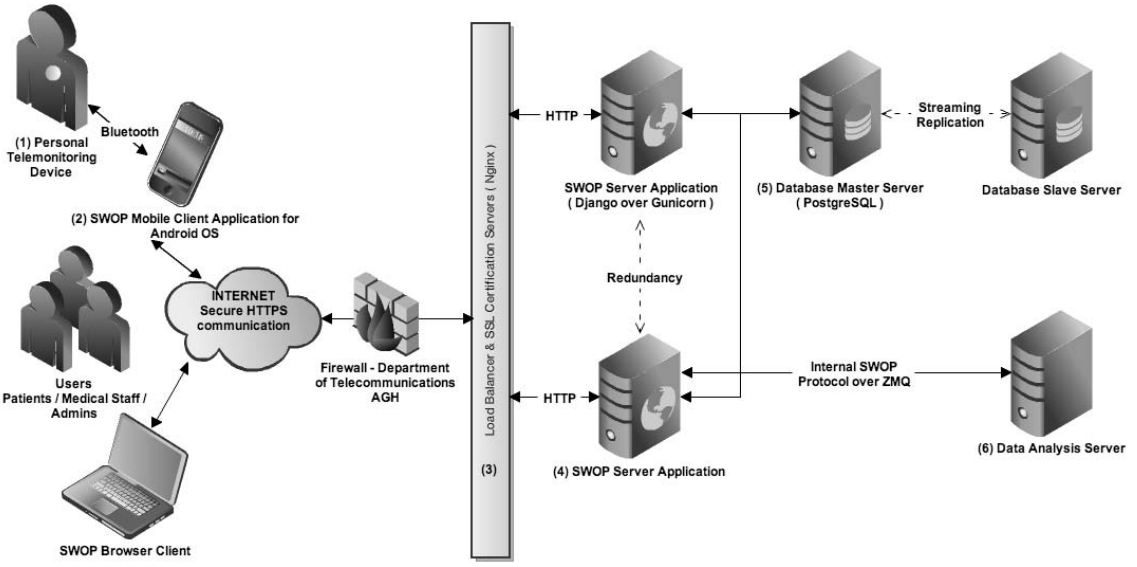

Fig. 5. Architecture of the SWOP system.

in which the members of the project development team participated. During the session, existent project documents and architectural views were analyzed and discussed. The following assets were identified:

1. Processes: telemonitoring, storing medical records, access to medical records, teleconsultation;
2. Services: data storage and retrieval, data transfer, analysis, SMS and e-mail notification;
3. Data: measurements, medical records and configuration data;
4. Software modules: Nginx proxy, SWOP application, SWOP database, SWOP mobile application, sensor software and SWOP data analysis;
5. Hardware modules: proxy firewall server, application server, DB server, data analysis server, smartphones and sensors;
6. Communication: external network https over WLAN, LAN and GPRS, internal network (HTTP) and Bluetooth for sensor to smartphone connections;
7. People: patients, medical and technical staff;
8. Infrastructure provided by a third party (communications, electricity).

6.2. Building the added value tree. The assets identified in the previous step constitute a network of dependent elements, i.e., the processes depend on services that are provided by software and hardware modules and refer to data which are stored and exchanged within the system as shown in Fig. 6. Influences between assets were identified based on architectural views, but particular weights were established during interviews with software architects and developers. They were then described in the form of an FCM influence matrix, using the following linguistic values: *high*, *significant*, *medium*, *low* and *none*.

To give an example, the utility of the *telemonitoring process* is *highly* influenced by the *Data storage* and *data transfer* services, *significantly* influenced by the *data analysis* service, and the utility of the *measured data* is influenced at *medium* level by the *SMS notification* and *e-mail notification* services and at a *low* level by *configuration data*.

Analogous statements were made for all assets. In most cases positive influences were assigned, however, in special cases, negative values were used to indicate that one asset can be replaced by another, e.g., the *SMS notification* and *e-mail notification* services were linked with *medium* negative influences.

6.3. Threats. The identification of threats was based on available sources (e.g., Guttman and Roback, 1995; Ross, 2011; Landoll, 2005), as well as on previous experience. The elicited list of threats to be considered in a vulnerability analysis comprised 58 elements grouped in 11 families corresponding to classes of assets.

The families are *process* (e.g., bad design), *software* (e.g., quality failures, lack of maintenance, malware), *hardware* (quality failures, resource exhaustion), *communications* (protocol weakness, service disruption), *data* (confidentiality or integrity breach), *external services* (loss of PKI, SMS gate, PaaS, SaaS), *external data providers* (errors in HL7 interfaces), *physical in-*

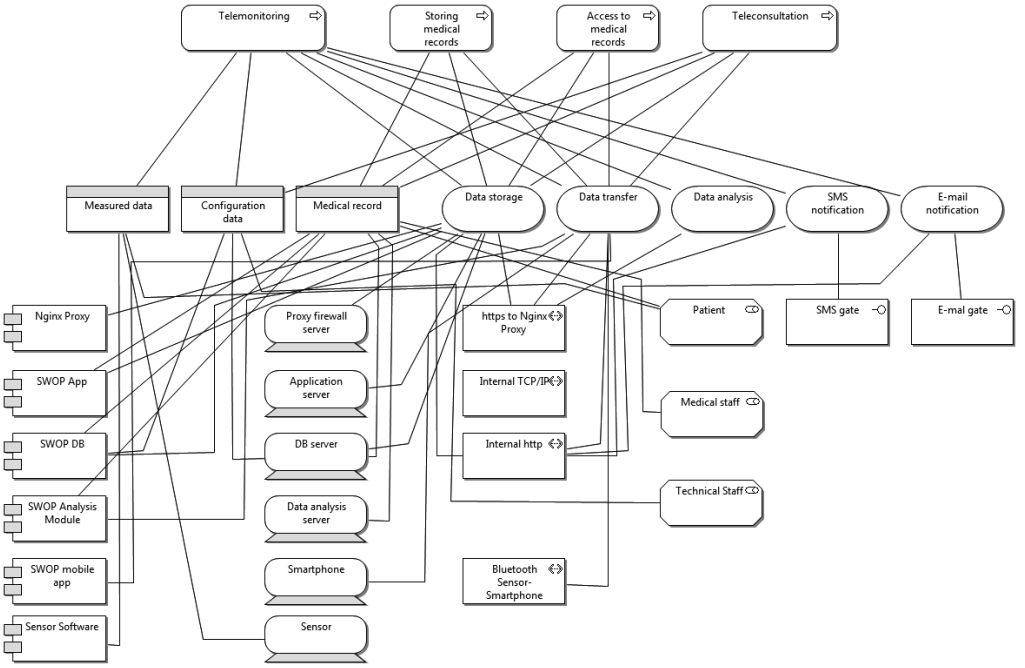

Fig. 6. Added value tree of assets: hardware, software, communication, people and external services contribute to data objects and services, which finally influence the utility of processes at the top of the diagram. The diagram was built based on architecture description in the ArchiMate language.

frastructure (premises, electricity, air condition), *people* (including threats related to patients, medical and technical staff), *natural disasters*, *medical decisions*, *economical conditions* and *legal*.

6.4. Risk assessment for individual assets. This step in the risk assessment process combines two activities identified in various methodologies, namely, the analyses of vulnerabilities and effectiveness of countermeasures. Technically, the assessment is performed using questionnaires in which answers reflecting best practices are attributed with weights describing their influence on a risk profile.

In the case of the SWOP system, we used a questionnaire comprising about 140 questions divided into 11 groups of threats and countermeasures.

The logical structure of a sample questionnaire related to the mobile application is presented in Table 1. For each question (a security feature), at most three answers (ratings) were defined. The answers were attributed with weights $q_{ij} \in [0, 1]$ that can be interpreted as their impact on the asset's risk profile. The weights are assigned after a voting process (questionnaires for the given asset type are prepared in advance and they represent "best practices"). Moreover, the influences of features can be differentiated with weight w_i shown in the last column of the table. These weights are not visible to the interrogated members of the development team, software architects and other involved stakeholders. The values that are underlined in Table 1 represent the answers for the SWOP system.

It should be observed that a questionnaire defines in fact the structure of a fuzzy cognitive map, in which weights express influences. Moreover, they were selected in a voting process, which is a typical practice in FCM construction.

The risk RA_s for an asset s is calculated with the formula (6) based on the values of answers a_{ij} to k_s questions Q_i, $i = 1, \ldots, k_s$. Values 1 and 0 are used for positive and negative answers. Hence, $a_{ij} = 1$ if the j-th answer to i-th question is given and 0 in other cases:

$$RA_s = \frac{1}{W} \sum_{i=1}^{k_s} w_i \sum_{j=1}^{3} a_{ij} q_{ij}, \text{ where } W = \sum_{i=1}^{k_s} w_i. \quad (6)$$

The normalization factor W in (6) plays an analogous role as an activation function in (2).

To illustrate the calculations, the answers to the questionnaire obtained during the interview with the project development team were marked in Table 1 by using underlined, bold font. The application of the formula (6) yields the value 0.38, which indicates that threats cannot be fully neutralized by countermeasures (which would hold if the calculated value was equal to 0). The values resulting from the questionnaires relating to particular assets were then used in the next step, aiming at the calculation of aggregated risks.

Table 1. Risk assessment questionnaire related to the mobile application. Answers obtained during an interview with a project development team are marked with underlined bold font.

Question Q_i	$Answer_1$	q_{i1}	$Answer_2$	q_{i2}	$Answer_3$	q_{i3}	w_i
Does the mobile application store a user name and/or password in the local memory/database?	yes, as not encoded data	1.0	**no**	0.0	as encoded data	0.5	1.0
Has the application code used to build the executable version been obfuscated?	yes	0.2	**no**	0.8			0.6
Does the communication with the middleware involve a third party proxy server?	yes	0.9	**no**	0.1			0.7
Is the application available at an official distribution channel (ex. Google Play, AppStore)?	yes	0.2	**no**	0.8			0.4
Does the communication use SSL?	**yes**	0	no	1	no verification of SSL certificate	0.5	1.0
Is antivirus software installed on the mobile device?	yes	0.1	no	0.9	**lack of information**	0.5	0.4

6.5. Calculation of aggregated risk with the FCM. The calculations were preceded by a normalization of the matrix of influences. While preparing the matrix, we used five linguistic variables to describe the influence: *high*, *significant*, *medium*, *low* and *none*. Then, they were mapped to weights $\{1.0, 0.75, 0.5, 0.25, 0\}$, and for each row $i = 1, \ldots, n$ the normalized values of influences were determined according to

$$\overline{e}_{ij} = \begin{cases} 0 & \text{if } e_{ij} = 0, \\ \exp(m \cdot e_{ij})/Z_i & \text{if } e_{ij} \geq 0, \end{cases} \tag{7}$$

where

$$Z_i = \sum_{\substack{j=1 \\ e_{ij} \neq 0}}^{n} \exp(m \cdot e_{ij})$$

and m is a positive constant. (In the calculations, the value $m = 1.0$ was used.)

Such normalization gives a probability distribution. Motivation for assuming the assumed distribution stems from game theory. Suppose that a high-level asset a_h depends on low-level assets $a_{l_1}, \ldots, a_{l_k}$, with influences $e_{hl_1}, \ldots, e_{hl_1}$. If a *threat agent* treated as an adversarial player is to select a low-level asset to launch an attack on, it should choose an element a_{l_m} giving the highest influence e_{hl_m} on the risk profile of a_h. However, the player can make errors in the estimation of influences. The resulting probability of adversarial actions depends on the distribution of errors, which, in general, is difficult to track. However, assuming a double exponential distribution of errors, we arrive at a *logit* model (Anderson *et al*., 1992) given by the formula (7).

For the final calculation of aggregated risks, two sequences of vectors were constructed:

$$\alpha^{nr} = A^{nr}(0), \ldots, A^{nr}(i), \ldots$$

and

$$\alpha^{r} = A^{r}(0), \ldots, A^{r}(i), \ldots$$

by successively applying the FCM state equation (2).

The *no-risk sequence* α^{nr} starts with a vector $A^{nr}(0)$, in which all elements expressing the utility of assets are set to 1. For the *risk sequence* α^{r}, the initial vector $A^{r}(0)$ is the difference of vectors of asset utilities $A^{nr}(0)$ and related risks RA established in the previous phase, using the formula (6): $A^{r}(0) = A^{nr}(0) - RA$.

Finally, by subtracting the corresponding elements of α^{nr} and α^{r}, we obtain a sequence of aggregated risk values,

$$\rho = R(0), \ldots, R(i), \ldots,$$

where $R(i) = A^{nr}(i) - A^{r}(i)$. This sequence converges to values that express aggregated risks for all assets at different levels of the added value tree.

Figure 7 shows the results of risk calculations for three groups of assets in the SWOP system: data, services and processes. They were obtained by applying activation functions S_{cut} (left) and S_{exp} (right) defined by the formulas (3) and (4). For the function S_{exp}, the value of the constant m in the formula (4) was set to 2.0. The comparison indicates that qualitative results for both functions are identical.

While interpreting the results of the calculations, the issue of converting them back to linguistic values appears, e.g., low, medium, high, often expected by stakeholders responsible for decision-making. To support such conversion, we established the values of two thresholds: $LM = 0.26$ and $MH = 0.52$ by conducting simple experiments: calculating risks in the cases of absence and presence of all safeguards. The resulting interval $[0.07, 0.86]$ was uniformly divided into three intervals corresponding to low, medium and high risk levels. The obtained thresholds pertain to the S_{cut} activation function, for S_{exp} they are about three times smaller.

6.6. Results of assessment. Our findings indicate a low level of aggregated risks related to assets placed at the top of the utility tree (processes, data and services). The medium risk for compound assets, e.g., the mobile application presented in the example in Section 6.4, is caused by the fact that a prototype, still not deployed in a production environment, was evaluated. We assume that, for a target deployment, several safeguards will be activated, e.g., using a trusted certificate authority, official

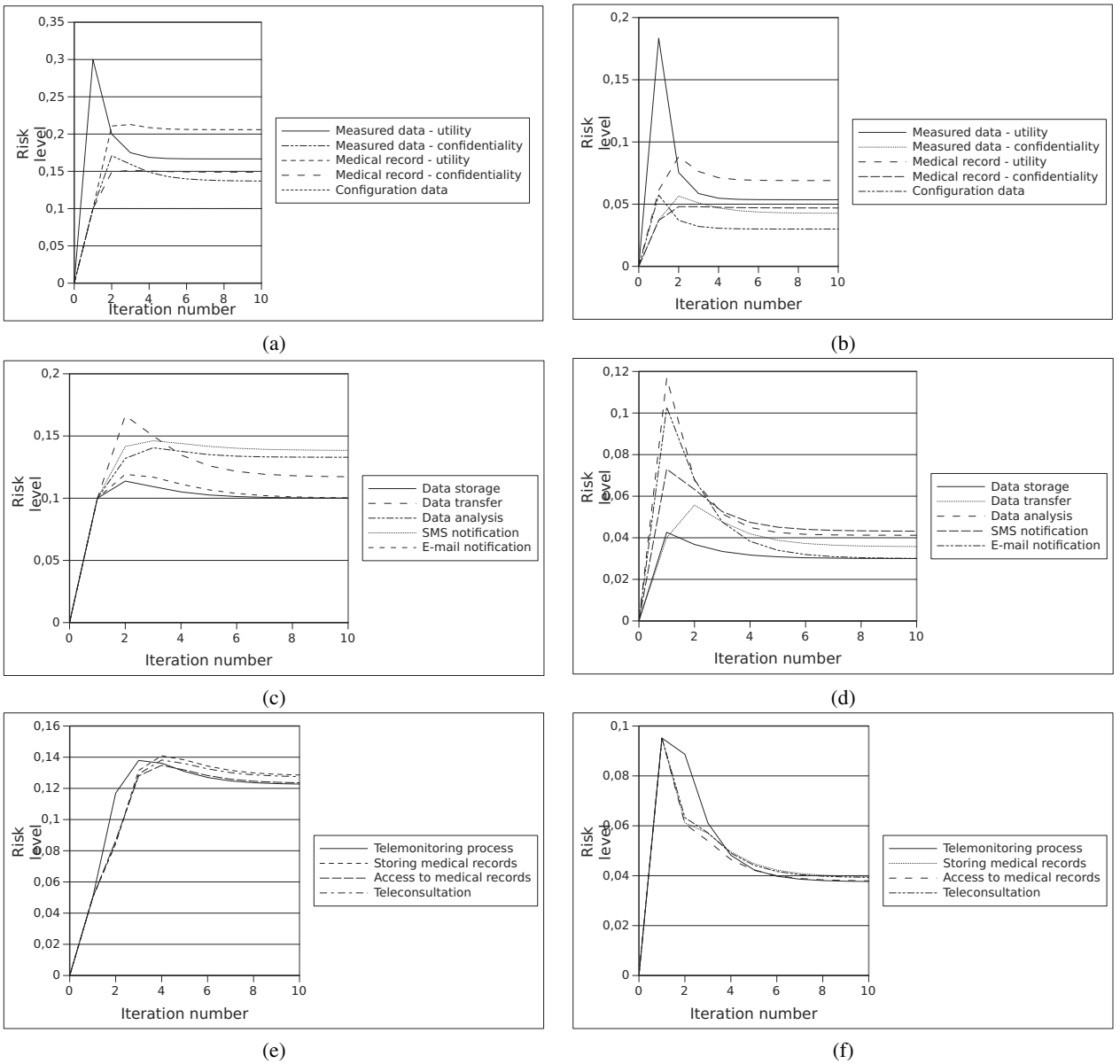

Fig. 7. Reasoning about risks related to data (a)–(b), services (c)–(d) and processes (e)–(f). Activation functions: S_{cut} (a), (c), (e) and S_{exp} (b), (d), (f) given by the formulas (3) and (4) were used.

distribution channels for the mobile application, UPS, regular backups, access control to physical premises, etc.

Our attention was attracted by relatively higher risks for the utility of medical records and measured data. In general, these risks are rather operational, than related to IT security. They are caused by threats falling into the category *people*, i.e., *low skill level*, *subjective selfobservation*, *Low selfdiscipline* or *Technology-related anger* for patients and *low attention level*, *epidemic illness*, *staff turnover*, *lack of professional behavior* for staff. Such risks can be partly mitigated by providing training, as well as by implementing still-absent reminders notifying patients about the necessity of feeding data.

7. Conclusions

This paper presented a new method for risk assessment of IT systems based on FCMs. The method includes steps present in various standards and methodologies: identification of assets, threats, analysis of vulnerabilities and effectiveness of countermeasures. However, it relies on FCM reasoning to calculate risks. The cornerstone of the proposed method is the *added value tree*, expressing dependencies between assets. A salient feature of the method is that it uses an abstract term *utility* (and loss of utility caused by a threat) in place of financial loss. This makes the method applicable for IT systems, for which financial loss is difficult to estimate. Moreover, at a lower

level assessment the method incorporates the widespread *best practices* approach to IT security, representing those by questionnaires.

We studied the method using the example of a SWOP e-health system and described its stages: preparing lists of assets based on architectural views and interviews, building an influence matrix reflecting an added value tree, identifying threats, calculating non-aggregated risks related to assets with the use of questionnaires, and finally performing reasoning with FCM techniques.

The proposed method can be considered a *lightweight* approach to risk assessment, suitable for small and medium-size systems. In the case of the SWOP system, the data were collected during three interviews and brainstorming sessions. In the meantime, questionnaires used in previous analyses by the assessment team were adapted to reflect the specific assets and threats.

The lessons learned indicate that the proposed method is an efficient and low-cost approach, giving instantaneous feedback and enabling reasoning on the effectiveness of a security system. It can be considered an alternative to heavy assessment processes defined by standards.

References

Aguilar, J. (2005). A survey about fuzzy cognitive maps papers, *International Journal* **3**(2): 27–33.

Anderson, S., De Palma, A. and Thisse, J. (1992). *Discrete Choice Theory of Product Differentiation*, MIT Press, Boston, MA.

Axelrod, R.M. (1976). *Structure of Decision: The Cognitive Maps of Political Elites*, Princeton University Press, New York, NY.

Baudrit, C., Dubois, D. and Guyonnet, D. (2006). Joint propagation and exploitation of probabilistic and possibilistic information in risk assessment, *IEEE Transactions on Fuzzy Systems* **14**(5): 593–608.

Birolini, A. (2000). *Reliability Engineering: Theory and Practice*, 3rd Edn., Springer-Verlag, Berlin.

Bowles, J.B. and Wan, C. (2001). Software failure modes and effects analysis for a small embedded control system, *Proceedings of the Annual Reliability and Maintainability Symposium, Philadelphia, PA, USA*, pp. 1–6.

Cervesato, I. and Meadows, C. (2003). Fault-tree representation of NPATRL security requirements, *Proceedings of the 3rd Workshop on Issues in the Theory of Security, Warsaw, Poland*, pp. 1–10.

Chen, X.Z. (2006). Hierarchical threat assessment and quantitative calculation method of network security threatening state, *Journal of Software* **17**(4): 885–897.

Chiang, F. and Braun, R. (2007). Self-adaptability and vulnerability assessment of secure autonomic communication networks, *Proceedings of the 10th Asia-Pacific Conference on Network Operations and Management Symposium: Managing Next Generation Networks and Services, APNOMS'07, Sapporo, Japan*, pp. 112–122.

Craft, R., Vandewart, R., Wyss, G. and Funkhouser, D. (1998). An open framework for risk management 1, *21st National Information Systems Security Conference, Arlington, VA, USA*.

Eom, J.-H., Park, S.-H., Han, Y.-J. and Chung, T.-M. (2007). Risk assessment method based on business process-oriented asset evaluation for information system security, *Proceedings of the 7th International Conference on Computational Science, Beijing, China*, pp. 1024–1031.

Guttman, B. and Roback, E.A. (1995). An introduction to computer security: The NIST handbook, *Security* **800**(12): 1–290.

Hagiwara, M. (1992). Extended fuzzy cognitive maps, *Proceedings of the IEEE International Conference on Fuzzy Systems, San Diego, CA, USA*, pp. 795–801.

Han, Y.-J., Yang, J.S., Chang, B.H., Na, J.C. and Chung, T.-M. (2004). The vulnerability assessment for active networks: Model, policy, procedures, and performance evaluations, *in* A. Laganà, M.L. Gavrilova, V. Kumar, Y. Mun, C.J.K. Tan and O. Geruasi (Eds.), *ICCSA (1),* Lecture Notes in Computer Science, Vol. 3034, Springer, Berlin/Heidelberg, pp. 191–198.

Hoo, K.J.S. (2000). How much is enough? A risk-management approach to computer security, *Working Paper*, Stanford University, Stanford, CA, pp. 1–99.

Hubbard, D. and Evans, D. (2010). Problems with scoring methods and ordinal scales in risk assessment, *Journal of Research and Development* **54**(3): 1–10.

Institute for Computer Sciences and Technology (1979). *Guideline for Automatic Data Processing Risk Analysis*, National Bureau of Standards, Washington, DC.

ISO/IEC (2011). Information technology—Security techniques—Information security risk management, *Technical Report ISO/IEC 27005:2011,* International Organization for Standardization, Washington, DC.

Jetter, A. and Schweinfort, W. (2011). Building scenarios with fuzzy cognitive maps: An exploratory study of solar energy, *Futures* **43**(1): 52–66.

Kobylarz, D. and Danda, J. (2013). A common interface for bluetooth-based health monitoring devices, *29th Southern Biomedical Engineering Conference (SBEC), Ho Chi Minh City, Vietnam*, pp. 153–154.

Kosko, B. (1986). Fuzzy cognitive maps, *International Journal of Machine Studies* **24**(1): 65–75.

Kosko, B. (1992). *Neural Networks and Fuzzy Systems: A Dynamical Systems Approach to Machine Intelligence*, Prentice Hall, Englewood Cliffs, NJ.

Landoll, D.J. (2005). *The Security Risk Assessment Handbook: A Complete Guide for Performing Security Risk Assessments*, Auerbach Publications, Boca Raton, FL.

Lazzerini, B. and Mkrtchyan, L. (2011). Analyzing risk impact factors using extended fuzzy cognitive maps, *IEEE Systems Journal* **5**(2): 288–297.

Maglogiannis, I., Zafiropoulos, E., Platis, A. and Lambrinoudakis, C. (2006). Risk analysis of a patient monitoring system using Bayesian network modeling, *Journal of Biomedical Informatics* **39**(6): 637–647.

Mikulik, J. and Zajdel, M. (2009). Automatic risk control based on FSA methodology adaptation for safety assessment in intelligent buildings, *International Journal of Applied Mathematics and Computer Science* **19**(2): 317–326, DOI: 10.2478/v10006-009-0027-1.

Modarres, M., Kaminskiy, M. and Krivtsov, V. (1999). *Reliability Engineering and Risk Analysis*, CRC Press, New York, NY.

Ozesmi, U. Ozesmi, S. (2004). Ecological models based on people's knowledge: A multi-step fuzzy cognitive mapping approach, *Ecological Modelling* **176**(1–2): 43–64.

Papageorgiou, E.I. (2011). Learning algorithms for fuzzy cognitive maps—A review study, *IEEE Transactions on Systems* **42**(2): 1–14.

Peng L.X. (2007). Model danger theory based network risk assessment, *Journal of University of Electron Science and Technology* **36**(6).

Ross, R.S. (2011). Guide for conducting risk assessments, *NIST Special Publication SP-800-30 Rev 1*, September, p. 85.

Schneier, B. (1999). Attack trees, *Dr. Dobb's Journal* **24**(12): 21–29.

Stamatis, D. H. (2003). *Failure Mode and Effect Analysis: FMEA from Theory to Execution*, ASQ Quality Press, Milwaukee, WI.

Stathiakis, N., Chronaki, C., Skipenes, E., Henriksen, E., Charalambus, E., Sykianakis, A., Vrouchos, G., Antonakis, N., Tsiknakis, M. and Orphanoudakis, S. (2003). Risk assessment of a cardiology ehealth service in HYGEIAnet, *Computers in Cardiology (CIC'2003), Cambridge, MA, USA*, pp. 201–204.

Sun, L., Srivastava, R.P. and Mock, T.J. (2006). An information systems security risk assessment model under the Dempster–Shafer theory of belief functions, *Journal of Management Information Systems* **22**(4): 109–142.

Szpyrka, M., Jasiul, B., Wrona, K. and Dziedzic, F. (2013). Telecommunications networks risk assessment with Bayesian networks, *in* K. Saeed, R. Chaki, A. Cortesi and S.T. Wierzchon (Eds.), *Computer Information Systems and Industrial Management*, Lecture Notes in Computer Sience, Vol. 8104, Springer-Verlag, Berlin, pp. 277–288.

Szwed, P. (2013). Application of fuzzy ontological reasoning in an implementation of medical guidelines, *6th International Conference on Human System Interaction (HSI), Sopot, Poland*, pp. 342–349.

Szwed, P., Skrzynski, P. and Grodniewicz, P. (2013). Risk assessment for SWOP telemonitoring system based on fuzzy cognitive maps, *in* A. Dziech and A. Czyżewski (Eds.), *Multimedia Communications, Services and Security*, Communications in Computer and Information Science, Vol. 368, Springer, Berlin/Heidelberg, pp. 233–247.

The Open Group (2012). Open Group Standard, Archimate 2.0 Specification, `www.opengroup.org`.

Vesely, W.E., Goldberg, F.F., Roberts, N.H. and Haasl, D.F. (1981). Fault tree handbook, *Technical Report Nureg-0492*, Nuclear Regulatory Commission, Washington, DC.

Wang Y., Zhu, A. and Zhang, J. (2011). Research on and application of the analyzing method of network security based on security case reasoning, *International Conference on Control, Automation and Systems Engineering (CASE), Tokyo, Japan,* pp. 1–4.

Zhuang, Y., Li, X., Xu, B. and Zhou, B. (2009). Information security risk assessment based on artificial immune danger theory, *Proceedings of the 2009 4th International Multi-Conference on Computing in the Global Information Technology, ICCGI'09, Cannes, France*, pp. 169–174.

Piotr Szwed received the M.Sc. degree in electronics and control engineering in 1988 and the Ph.D. degree in computer science in 1999, both from the AGH University of Science and Technology in Cracow, Poland. Since 1999 he has been working there as an assistant professor. Currently he is with the Department of Applied Computer Science at the Faculty of Electrical Engineering, Automatics, Computer Science and Biomedical Engineering. His research topics cover various aspects of software engineering including modeling, assessment of software architectures and formal verification. He is particularly interested in fuzzy reasoning, application of ontologies and Petri nets.

Pawel Skrzyński graduated from the AGH University of Science and Technology: received there an M.Sc. in computer science (2002), an M.Sc. in business and management (2004), and a Ph.D. in computer science (2011). Interested in enterprise architecture, software development, distributed systems, software and IT security, Java technologies, mobile technologies. Works at the Department of Applied Computer Science and has been involved in development and management of several big IT projects including Internet/mobile banking systems for the biggest Polish banks, and a VISP project (EU FP6).

Received: 19 March 2013
Revised: 12 August 2013
Re-revised: 30 August 2013