Towards Mixed-Mode Risk Management – A Concept

Andrzej Bialas(✉) and Barbara Flisiuk

Institute of Innovative Technologies EMAG,
Leopolda 31, 40-189 Katowice, Poland
{andrzej.bialas,barbara.flisiuk}@ibemag.pl

Abstract. The paper concerns the risk management issue. Different approaches – static and real-time (dynamic) are reviewed, as well as their advantages and gaps. A broadly used static risk assessment/management (SRA/M) process is comprehensive, complex, invoked periodically, but it omits fluctuating risk factors and has limited ability to adapt itself to the changing risk picture. The paper proposes a mixed-mode approach. The SRA/M process is transformed towards the adaptive risk management (ARM) process. This adaptation is based on real-time risk management (RTRM) results gathered during a period between static risk assessments. All risk management processes are expressed in a pseudo-code. The method is exemplified by a simple but representative case study.

Keywords: Risk management · Real-time risk management · Adaptive risk management

1 Introduction

The paper concerns an event-based advanced risk management methodology which embraces static and dynamic aspects of the risk nature in security.

Risk management is a continuous process. It includes the identification, analysis, and assessment of potential hazards in a system or hazards related to a certain activity. Once the risk picture is recognized, there are some risk control measures proposed to eliminate or reduce potential harms to people, environment, or other assets. In addition, the risk management process embraces risk monitoring and communication. ISO 31000 [1] is the basic risk management standard. Examples of the most recognized risk management methods and techniques are included in ISO/IEC 31010 [2]. This issue was discussed in [3–5].

The methods described in the standards have static character, are performed periodically and are able to analyze steady state and low frequency risk factors. The main gap in this approach is related to the fact that fluctuating threats or hazards, occurring between the performed analyses, are omitted, though they can be dangerous as well.

The objective of the paper is to analyze whether fluctuating, dynamic phenomena may be taken into consideration during the risk management process and to present a new mixed-mode approach.

Section 2 includes a short review of static and dynamic risk management approaches. Section 3 presents the static approach and how to integrate real-time risk

W. Zamojski et al. (eds.), *Advances in Dependability Engineering of Complex Systems*,
Advances in Intelligent Systems and Computing 582, DOI 10.1007/978-3-319-59415-6_2

management in this approach to obtain the mixed-mode approach. Section 4 discusses a short example of the mixed-mode method, while the final section concludes research results and discusses further steps.

2 Risk Management – Static, Iterative and Real-Time Approaches

Today's risk analysis methods (including those described in [2–4, 6]) are based mainly on a static risk management approach. This approach is typically applied in strategic risk management where the risk is assessed based on static factors existing over relatively long periods of time. The analyses are made periodically with respect to the current situation and historical data. However, risk levels that are determined might not reflect adequately the changes occurring in the business environment, as fluctuating risk factors are not taken into consideration.

These drawbacks can be eliminated by means of the iterative risk management approach (IRM), also called adaptive risk management approach (ARM). For example, the paper [7] uses this approach to mitigate the uncertainty problem during a sample risk analysis for the technical component used in the oil and gas industry. These analyses are conducted periodically too, still, each successive analysis is made in an improved manner (with respect to the knowledge acquired over time). The idea is to use the experiences learnt from the operation of security measures that had been selected and implemented based on the previous analysis. Though this approach is definitely a flexible one, it is still related to some disadvantages. For example, it is difficult to identify suitable risk thresholds when multiple risks are acting together, and when different scenarios and threshold assumptions have to be considered. Whenever the methods and thresholds change, it is difficult to compare the results.

As fluctuating risk factors and the related increased uncertainty are disregarded, currently used risk management methods of these two approaches are often ineffective. Thus it is necessary to employ risk management methods that would assess and reduce risk in real time. Static and iterative methods should be supported by methods that would adapt dynamically to the fast changing picture of threats and changes in the protected objects (their assets and vulnerabilities). The newly developed solutions should provide on-line communication, statistical data about incidents to support risk management, data from security analyzers or on-line security monitoring systems.

Cyber security needs on-line monitoring of the existing risks due to fast changes of fluctuating risk factors. It deals with both previously identified and emerging threats and vulnerabilities. New technologies, such as cloud computing, mobile computing, virtualization, Internet of Things, etc., generate new, specific threats and vulnerabilities. This was motivation to elaborate real-time risk management (RTRM) approaches which are typical of operational risk management. According to [8], RTRM systems should comprise the following elements:

- instantaneous knowledge – changes in assets, threats, vulnerabilities, risk categories and levels should be visible instantly;

- comprehensive visibility – to have a clear picture of assets and the related vulnerabilities and potential impacts, one must have consistent data, visibility, and alerts;
- constant controls assessment and adjustment – security measures should be assessed periodically and persistently in order to address new or changing risks.

The objective of the real-time risk management approach is to view the entire IT infrastructure in this respect and ensure a deep technical insight into each technology of this infrastructure. This approach makes it easier to monitor security events in real time, to find their correlations, to interpret, alarm and visualize the current security picture. According to [8], advanced real-time risk management systems have to be supported by uninterrupted threat monitoring, knowledge correlation and alerting engines, and, as far as possible, automatic response.

The role of the Security Information and Event Management (SIEM) has been growing recently [9, 10]. SIEM systems are designed to aggregate, analyze and present large volumes of security-related data acquired from the network and security devices with a view to detecting threats, incidents, vulnerabilities, frauds, including proper reaction and warning against them.

Cyber security cannot be properly managed when its status is not measured. Measures can be based on indicators coming from different sources:

- SIEM and data analytics tools which provide valuable information on actual or potential compromise on the network,
- threat intelligence services, compliance management, vulnerability management, penetration testing and audits are helpful to identify data losses and provide valuable information on actual or impending attacks.

The holistic approach is recommended – using all this information [11] the cyber security status can be identified and presented to decision makers. A very important issue is to maintain and share security-related information [12, 13].

Real-time risk management is used mainly for cyber security risks. There are no integrated, holistic approaches that would embrace the static-, adaptive- and real-time risk management. It can be strong motivation for researchers to work out methods and tools in this field. The paper proposes a concept of such a method.

There are many event-based risk management techniques applied in the security domain. Similar techniques exist in the safety domain, along with system theory-based techniques to better tackle issues in this domain, such as complex software-intensive systems, complex human-machine interactions and systems-of-systems [14].

3 Mixed-Mode Risk Management Approach – A Concept

A mixed-mode risk management concept is based on the static risk management loop supplemented by real-time risk management facilities. The methodology should comply with ISO 31000 [1]. There are a lot of static risk analysis methods [2]. The elaborated mixed-mode risk management framework will be based on a commonly known and relatively simple method: consequence-probability matrix. Other, more complex methods can be considered in the future. Both process- and asset-oriented approaches are possible but the paper is focused on the asset-oriented one.

3.1 Static Risk Management (SRM) Process

The static risk assessment is performed periodically, e.g. yearly (Organizational Security Policy specifies “when”). Let us assume that when this time occurs the SRA_event is triggered, initiating the main loop of the SRM process – presented below in a pseudo-code. For any quadruple <asset, threat, vulnerability, existing countermeasures> the likelihood and consequences of the hazardous event caused by a threat are assessed and then the risk level is calculated using these two parameters (usually as a product of them).

```
IF SRA_event THEN
 FOR each identified elementary asset
  FOR each threat relevant to elementary asset
   FOR each vulnerability relevant to threat
    COMMENT start existing risk assessment ("as-is")
    BeforeCountermeas :=
          IdentifyExistingCountermeasures()
    BeforeLikelihoodLevel := AssessLikelihood()
    BeforeConsequencesLevel := AssessConsequences()
    BeforeRiskLevel :=
          BeforeLikelihoodLevel*BeforeConsequencesLevel
    COMMENT initializing variables for main
    risk management loop
    AfterCountermeas := BeforeCountermeas
    AfterLikelihoodLevel := BeforeLikelihoodLevel
    AfterConsequencesLevel := BeforeConsequencesLevel
    AfterRiskLevel := BeforeRiskLevel
    COMMENT main risk management loop
     WHILE AfterRiskLevel > RiskAcceptanceLevel
      AfterCountermeas :=
          SelectBetterCountermeasures()
      AfterLikelihoodLevel := AssessLikelihood()
      AfterConsequencesLevel := AssessConsequences()
      AfterRiskLevel :=
          AfterLikelihoodLevel*AfterConsequencesLevel
     ENDWHILE
   ENDFOR (vulnerabilities)
  ENDFOR (threats)
 ENDFOR (assets)
ENDIF (SRA_event)
```

As a result of that, the risk pictures before and after the mitigation are identified. Please note that the heuristic, human activities are expressed as "`functions()`", and marked *italic*.

Risk management embraces risk analysis, risk assessment, and the selection of countermeasures, when the current risk level (`AfterRiskLevel`) exceeds the `RiskAcceptanceLevel`. The method assumes that during the preliminary analysis (`BeforeRiskLevel`) certain countermeasures exist and mitigate the risk to a certain extent, usually insufficiently. These measures ought to be revised. They can be accepted or replaced by their more effective combinations, and then the risk should be reassessed. Countermeasure is understood here as a coherent, diversified set of elementary security measures.

Summing up, for any triple <asset, threat, vulnerability>, called risk scenario, the risk before and after reduction is assessed and the existing and applied countermeasures are specified as the static risk record:

`srr=(asset, threat, vulnerability, BeforeCountermeas, BeforeLikelihoodLevel, BeforeConsequencesLevel, Before-RiskLevel, AfterCountermeas, AfterLikelihoodLevel, After-ConsequencesLevel, AfterRiskLevel).`

The static risk can be considered as a sum of these records for all scenarios:

$$SR = \bigcup\nolimits_{all\,scenarios} \{srr\} \tag{1}$$

The static risk (SR) assessment creates a general picture of the risk situation in a certain moment of the system life cycle. These moments can be shown as points on a discrete time scale: $SR(T_0)$, $SR(T_1)$, $\ldots SR(T_i)$, $SR(T_{i+1})$, $SR(T_{i+2})$, $\ldots SR(T_n)$.

Please note that $SR(T_{i+1})$ is determined using the risk factors identified at the end of $SR(T_i)$ and it represents the current static risk picture, valid until the new analysis result, i.e. $SR(T_{i+2})$ replaces the old one, e.g. after one year. Between any consecutive time points many unpredicted, negative phenomena may occur. The time period between these points is called the SRA period. It is difficult to raise the frequency of static risk assessments to increase the preciseness of assessments because SRA is a time and cost consuming process. RTRM may be helpful to solve this problem.

3.2 Real-Time (Dynamic) Risk Management (RTRM) Process

Real-time risk assessment allows to detect changes of risk relevant factors which influence the overall risk picture and its elements, like: assets, threats, vulnerabilities, countermeasures, likelihood, consequences, etc. RTRM allows to react to these changes as well. This reaction occurs in a relatively short time, allows to revise the current static risk picture and to properly respond to incidents or their symptoms.

Please note that during the previous static risk assessment certain threats or vulnerabilities may be unknown, and some of them may not be identified properly. For this reason the security system (set of the managed countermeasures) is unable to react to them. Similarly, after the previous static risk assessment some organizational changes may occur as well as changes in the business processes, IT infrastructure or business environment. Due to that the countermeasures, selected or updated last time, do not suit the new situation.

The risk assessment and change management shortcomings may cause security incidents, which should be properly managed (reaction, mitigation of damages, lessons learnt, corrective actions within the protection system). Factors, e.g. incident symptoms, non-compliances should be identified immediately and properly managed by the real-time risk management process. The factors influencing a current risk level are diversified, derived from different sources and have different dynamics and nature. The paper proposes means and ways to manage them in a unified way and to identify how the current risk factors modify the previously identified static risk picture.

The real-time risk management process needs input, i.e. information about risk relevant factors. Table 1 presents six sources of information about incidents or their symptoms which can be considered as the input for the real-time risk management process. They should be compared with these specified in the paper [11]. The SEM (Security event management) and SIM (Security information management) systems are distinguished because they have different input. Both can be replaced by SIEM. Sources are activated on request when a behaviour anomaly is detected, when an incident occurs or a certain variable exceeds the assumed level. These sources have different response (identification) time and nature with respect to the incident or its symptom. For each component representing sources, the main outputs are specified along with the activities required in the RTRM process. “Ex ante” concerns incident symptoms, and “ex post” – incident consequences.

Sources of information constitute an intermediate layer of components between the secured object and the RTRM process. The layer is identified and presents, in a unique way, all risk relevant factors whose changes may influence the current overall risk picture. Based on this input, the RTRM process will focus only on the areas of the risk picture which are impacted by these factors. The areas may concern all scenarios (called RT scenarios) related to the given asset or group of assets.

The real-time risk management process runs similarly to SRM, though it is preceded by the preliminary stage of identifying the RT scenarios, which should be reassessed by the RTRM process. This is necessary for SEM, SIM, Penetration tests, and vulnerability scanners, because their output usually points at threats and vulnerabilities only. For this reason, during the preliminary stage all relevant assets for pairs <threat, vulnerability> should be identified. These activities are represented below by the `CompleteRiskScenarios()` function. Generally it is an operation on the database storing the risk scenarios.

One or more completed RT scenarios <asset, threat, vulnerability> are provided as the input for the RTRM process. These scenarios define an area of the overall risk picture which requires reassessment, because the changed risk-related factors may influence the risk level in this area.

Any change in any source component output generates the event `RTRM_event` initiating the RTRM process:

```
IF RTRM_event THEN
 CompleteRiskScenarios()
 COMMENT one or more completed risk scenarios defining
 the possible impacted area as the RTRM input
 FOR each identified elementary asset belonging to
  the RT scenarios subset
  FOR each threat relevant to elementary asset
   FOR each vulnerability relevant to threat
    COMMENT start existing risk assessment
    BeforeRTRMCountermeas :=
          IdentifyExistingRTRMCountermeasures()
    BeforeRTRMLikelihoodLevel := AssessLikelihood()
    BeforeRTRMConsequencesLevel := AssessConsequences()
    BeforeRTRMRiskLevel :=
          BeforeRTRMLikelihoodLevel
          *BeforeRTRMConsequencesLevel
    COMMENT initializing variables for main
    RTRM risk management loop
    AfterRTRMCountermeas :=
           BeforeRTRMCountermeas
    AfterRTRMLikelihoodLevel :=
           BeforeRTRMLikelihoodLevel
    AfterConsequencesLevel :=
           BeforeConsequencesLevel
    AfterRTRMRiskLevel :=
           BeforeRTRMRiskLevel
    COMMENT main RTRM risk management loop
     WHILE AfterRTRMRiskLevel > RiskAcceptanceLevel
      AfterRTRMCountermeas :=
           SelectBetterCountermeasures()
      AfterRTRMLikelihoodLevel := AssessLikelihood()
      AfterRTRMConsequencesLevel :=
           AssessConsequences()
      AfterRTRMRiskLevel :=
           AfterRTRMLikelihoodLevel
           *AfterRTRMConsequencesLevel
     ENDWHILE
   ENDFOR (vulnerabilities)
  ENDFOR (threats)
 ENDFOR (assets)
ENDIF (RTRM_event)
```

Table 1. Real-time risk management process – inputs and activities

Source	Description	Invoked	Identification time	Nature	Main output	RTRM activities
SEM	Real-time monitoring of the IT system behaviour (events), the correlations analysis of events, issuing warnings and aggregated information	By behaviour anomalies	Fast	Ex ante	Threat symptoms Possible vulnerability	Find relevant assets and perform the RTRM process with respect to the <threat, vulnerab.>
SIEM	Long term storage, analysis, reporting of log data, issuing aggregated alerting data	By behaviour anomalies	Fast	Ex ante	Threat symptoms Possible vulnerability	Find relevant assets and perform the RTRM process with respect to the pair <threat, vulnerab.>
Incident management	Identifies causes, consequences and circumstances of the occurred incident	On incident	Fast	Ex post	Breached asset, Damages, Occurred threat, Exploited vulnerability	Perform RTRM process for relevant assets using available information
Penetration tests and vulnerability scanners	Controlled attack on a computer system that looks for security weaknesses, potentially gaining access to the computer features and data	On request	Fast	Ex ante	Not properly countered threats, Possible vulnerabilities	Find relevant assets and perform the RTRM process with respect to the <threat, vulnerab.>
Audit process	Identifies non compliances including exposures to new threats, new vulnerabilities, non-efficient countermeasures, organizational and procedural shortcomings, etc.	On request	Slow	Ex ante	Not properly countered threats Possible vulnerabilities Possible damages	Perform RTRM process for relevant assets using available information

(*continued*)

Table 1. (*continued*)

Source	Description	Invoked	Identification time	Nature	Main output	RTRM activities
Security effectiveness measures	Provide aggregating security-related characteristics	On warning (issued when a measure exceeds the assumed level)	Slow	Ex ante	Breached asset, Damages, Occurred threat, Exploited vulnerability, Incident rates, Protection costs, Security management status	Perform RTRM process for relevant assets using available information

Summing up, for any triple <asset, threat, vulnerability> belonging to the assessed area, the risk before and after reduction is assessed and the existing and applied countermeasures are specified as the real-time risk record:

```
rtr=(asset, threat, vulnerability, BeforeRTRMCountermeas, BeforeRTRMLikelihoodLevel, BeforeRTRMConsequencesLevel, BeforeRTRMRiskLevel, AfterRTRMCountermeas, AfterRTRMLikelihoodLevel, AfterRTRMConsequencesLevel, AfterRTRMRiskLevel).
```

The real-time risk can be considered as a sum of these records for all possible RT scenarios:

$$RTR = \bigcup\nolimits_{all\,RT\,scenarios} \{rtr\} \tag{2}$$

The RTRM risk assessment creates a dynamic picture of the risk situation during a certain SRA period i. This picture consists of a set of scenarios in the area impacted by changes of the risk relevant factors. These moments can be shown as points on a discrete time scale too, in the range of the given SRA period i: $RTR(T_{i,0})$, $RTR(T_{i,1})$, $\ldots RTR(T_{i,j})$, $RTR(T_{i,j+1})$, $RTR(T_{i,j+2})$, $\ldots RTR(T_{i,m})$, where $RTR(T_{i,m})$ is the real-time risk accumulated at the end of SRA period i (*m* RTRM events occurred). Please note that $RTR(T_{i,j})$ is determined using the risk factors identified when `RTRM_event` j occurs.

3.3 Mixed-Mode Risk Management Process

In the pure static approach, $SRA(T_{i+1})$ is determined with the use of risk factors identified at the end of $SR(T_i)$ and it represents the current static risk picture, valid until the new analysis result, i.e. $SR(T_{i+2})$ replaces the old one. SRA is focused on steady state or low frequency risk factors. The real-time-risk management process is performed parallel to the static risk management during the given SRA period and is focused on dynamically changing, fluctuating risk factors. The RTRM process can be invoked many times during this period and particular events may concern different areas of the risk picture. The RTRM process samples dynamically occurring events and reacts to them. The dynamic risk picture cumulates all dynamic risks within the SRA period.

The authors propose a new integrated approach combining the static, real-time and adaptive approaches.

It is assumed that at the beginning of the SRA period i, a real-time risk analyzer is initialized with the current static assessment results: $RTR\left(T_{i,0}\right) = SR(T_i)$. This ensures a common starting point of both assessments. Starting from this point the real-time risk is cumulated until the end of period i. The accumulated result $RTR\left(T_{i,m}\right)$ is used in the adaptive risk assessment for the next period i + 1 (SRA and ARA periods are the same).

When the SRA/ARA period ends, the dynamic risk picture parameters are considered by the new ARM process allowing to determine the adaptive risk $AR(T_{i+1})$:

$$AR(T_{i+1}) = ARF\left[SR(T_{i+1}),\, RTR\,\left(T_{i,m}\right)\right] \tag{3}$$

The new SRA results, i.e. $SR(T_{i+1})$, are refined by the RTR results $RTR\,\left(T_{i,m}\right)$ accumulated at the end of the previous period. This refinement, expressed by the *Adaptive Risk Function* (*ARF)*, is a complex heuristic process in which many fuzzy factors and relations must be taken into consideration. Three groups of data are used on the input to determine the adaptive risk for the next period $(AR(T_{i+1}))$:

- the static risk picture at the beginning of the previous SRA/ARA period $(SR(T_i) = RTR\left(T_{i,0}\right))$: `BeforeCountermeas, BeforeLikelihoodLevel, BeforeConsequencesLevel, BeforeRiskLevel`; it expresses the common reference point, the context of assessment;
- the updated static risk picture at the beginning of the next SRA/ARA period $(SR(T_{i+1}))$: `AfterCountermeas, AfterLikelihoodLevel, AfterConsequencesLevel, AfterRiskLevel`; it represents the risk related to steady state or low frequency risk factors;
- the cumulated real-time risk $\left(RTR\left(T_{i,m}\right)\right)$ at the end of the previous SRA/ARA period expresses the "corrections" caused by real symptoms or incidents occurred and the knowledge related to them: `AfterRTRMCountermeas, AfterRTRMLikelihoodLevel, AfterRTRMConsequencesLevel, AfterRTRMRiskLevel`; it represents dynamically changing, fluctuating risk factors.

It is necessary to answer some difficult questions, like: how to merge the static and dynamic countermeasures, how they will impact jointly threats and vulnerabilities, how to assess the risk level and consequences in this situation, etc. The automation of this decision process may be advantageous though it still remains a challenge.

The above described mixed-mode risk management process focused on the selected risk scenario is summarized in Fig. 1. Please note the static and real-time paths and the process adaptation (a decision improvement by RTR acquired data) before the next step. Please note that only the most important elements of risk records are shown.

4 Exemplification of Mixed-Mode Risk Management Process

The exemplification of the presented concept is restricted to the selected example embracing one asset A_x. A more extensive validation is planned after the methodology implementation on the software platform.

It is assumed that likelihood levels and consequences levels (SR/RTR/AR) are measured in the range from 0 to 10, causing the risk range from 0 to 100. Risk acceptance levels are 60.

Let us consider a scenario: Threat T_y exploiting the vulnerability V_z breaches the asset A_x despite of the existing countermeasure C_p. It is also assumed that every countermeasure represents a set of different, working coherently security measures: technical, physical and organizational. A short notation for risk records, instead of self-explaining ones, is proposed as well:

SCM_m=BeforeCountermeas, SCM_{m+1}=AfterCountermeas,
$RTCM_n$=BeforeRTRCountermeas, $RTCM_{n+1}$=AfterRTRCountermeas.

1. The first static risk assessment (T_1), started with $SCM_{1,0}=C_p$ is performed on request and gives the following results: $SR(T_1)=(SCM_0, 70, SCM_1, 55)$, where BeforeRiskLevel=70, AfterRiskLevel=55. These data are used to initialize the real-time risk record $RTR(T_{1,0})=SR(T_1)$.
2. Let us assume that during the T_1 period an incident related to the considered threat T_y occurs (RTR_event). The incident lessons learnt conclude that the likelihood and consequences (i.e. the risk) were underestimated (probably they are higher than 55) in reality. In order to maintain the risk on this level, better countermeasures should be applied (to reduce vulnerability/threat impact). The existing countermeasures SCM_1 are improved to $RTCM_2$, causing the risk level modification: $RTR(T_{1,1}) = (RTCM_0 = SCM_1, 55, RTCM_1, 54)$. The RTR risk level is a little lower (54), because $RTCM_1$ is more effective than $RTCM_0$.
3. Let us assume that during T_1 SIEM discovers an anomaly in the area related to T_y (RTR_event), interpreted as attack symptoms. The security problem analysis concludes that certain risk scenarios related to these symptoms, including the considered scenario, should be reassessed. As a result of this RTR reassessment, the countermeasures were improved again: $RTR(T_{1,2}) = (RTCM_1, 54, RTCM_2, 53)$ and RTR risk level decreased (54–>53).
4. Let us assume that during T_1 the audit in the data center detects a previously unknown vulnerability (RTR_event), reinitiating the RTRM process. The

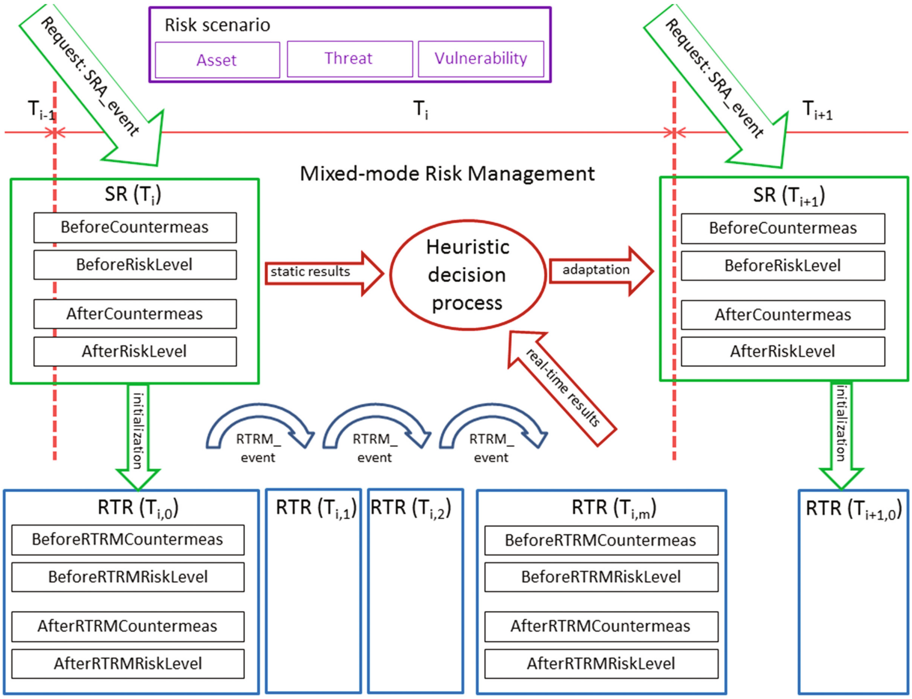

Fig. 1. General scheme of the mixed-mode risk management

reassessment concludes that this vulnerability should be removed by further countermeasures improvement ($\mathtt{RTCM_3}$): $\mathtt{RTR(T_{1,3})} = (\mathtt{RTCM_2}, 53, \mathtt{RTCM_3}, 52)$.

5. The security policy statement regulates the times of the next static risk assessment which has to be done. The $\mathtt{SR(T_2)}$ assessment takes into consideration the current risk situation, including all RTR assessments and countermeasures applied during T_1 (an adaptive approach): $\mathtt{SR(T_2)} = \mathtt{AR(T_2)} = \mathtt{ARF[SR(T_1), RTR(T_{1,3})]}$.

The $\mathtt{SR(T_2)}$ assessment results are used to initialize $\mathtt{RTR(T_{2,0})}$. The adaptive risk management system starts to watch for RTR_events during the next time period T_2.

5 Conclusions

The paper is focused on the integration of static and dynamic risk management methods to minimize drawbacks, when they are used separately. The mixed-mode risk management method is proposed.

The static approach is comprehensive and embraces relatively long time periods, e.g. one year, and its loop is invoked on the security managers' request. Between static risk assessments the fluctuating risk factors (new threats, their symptoms, unconformities, new vulnerabilities, etc.) may occur suddenly and countermeasures are not able to counter them. Thus during these periods the real-time risk management loop is invoked on each dynamically occurred event, allowing for the reaction and correction of countermeasures. The security related data sampled by the RTR manager during this period are used to update the risk picture for the next "static" time period. Section 4 presents an example how the mixed-mode approach can be used.

The subject of discussion is very extensive because it comprises the integration of several subsystems, like risk manager, incident manager, SIEM, audit subsystem, etc. Most of them exist in the OSCAD software platform [15]. For this reason a more comprehensive validation is planned using this platform. This will allow to reach a certain level of automation. Future research should embrace also the unified representation of threats, vulnerabilities, assets, countermeasures and relations between them within the mixed-mode risk manager using renown standards.

References

1. ISO 31000:2009, Risk management – Principles and guidelines
2. ISO/IEC 31010:2009 – Risk Management – Risk Assessment Techniques
3. Rausand, M.: Risk Assessment: Theory, Methods, and Applications. Statistics in Practice (Book 86). Wiley (2011)
4. Hokstad, P., Utne, I.B., Vatn, J. (eds.): Risk and Interdependencies in Critical Infrastructures: A Guideline for Analysis. Reliability Engineering. Springer-Verlag, London (2012). doi:10.1007/978-1-4471-4661-2_2
5. Bialas, A.: Risk management in critical infrastructure—foundation for its sustainable work. Sustainability **8**, 240 (2016). http://www.mdpi.com/2071-1050/8/3/240/htm
6. ENISA. http://rm-inv.enisa.europa.eu/methods. Access date: Dec 2016
7. Bjerga, T., Aven, T.: Adaptive risk management using new risk perspectives – an example from the oil and gas industry. Reliab. Eng. Syst. Saf. **134**, 75–82 (2015)
8. Oltsik, J.: Real-Time Risk Management (2010). http://la.trendmicro.com/media/misc/real-time-risk-management-en.pdf. Access date: Nov 2016
9. NetIQ Sentinel. https://www.netiq.com/products/sentinel/. Access date: Jan 2017
10. QRadar IBM. http://searchsecurity.techtarget.com/feature/IBM-Security-QRadar-SIEM-product-overview. Access date: Jan 2017
11. Marvell, S.: The real and present threat of a cyber breach demands real-time risk management, Acuity Risk Management (2015)
12. TAXII. http://taxiiproject.github.io/about/. Access date: Jan 2017
13. MITRE. https://cve.mitre.org/cve/. Access date: Jan 2017
14. Dulac, N.: A Framework for Dynamic Safety and Risk Management Modeling in Complex Engineering Systems, Ph. D. thesis, Massachusetts Institute of Technology (2007). http://sunnyday.mit.edu/safer-world/dulac-dissertation.pdf. Access date: Mar 2017
15. OSCAD project. http://www.oscad.eu/index.php/en/. Access date: Dec 2016

http://www.springer.com/978-3-319-59414-9

Advances in Dependability Engineering of Complex Systems
Proceedings of the Twelfth International Conference on Dependability and Complex Systems DepCoS-RELCOMEX, July 2 - 6, 2017, Brunów, Poland
Zamojski, W.; Mazurkiewicz, J.; Sugier, J.; Walkowiak, T.; Kacprzyk, J. (Eds.)
2018, XVI, 488 p. 201 illus., Softcover
ISBN: 978-3-319-59414-9