Gergely Mészáros

E-mail: meszaros.gergely@gmail.com

OPEN SOURCE SECURITY FROM RISK MANAGEMENT PERSPECTIVE

Resume:

Acceptance of open source components has become commonplace in the business world over the last decade. Given their usage is not unimaginable in governmental, military or critical infrastructure applications, it is crucial to develop policies to avoid gamble and properly eliminate or mitigate the risk evoked by utilization of open sourced components. In this presentation we identify the place and significance of Open Source in the context of various risk management principles, concentrating on unique features like development methods, legal aspects and security consequences.

Keywords: infosec, risk management, open source

Introduction

Risk management is a relatively young discipline that is constantly evolving. Many publications and standards has been released about the topic in the past decades, covering wide range of applications and theory. In our presentation we investigate the consecvences of Open Source (FOSS[1]) in the broader field of information risk management (IRM) with respect to some military and business information security standards.

Attitude towards Open Source components utilization increasingly favorable. It is certainly not uncommon to embrace wide range of open sourced components in commercial projects, even in military or governmental applications [1], [2]. In our paper we are referring the phrase « open source » conforming to the Open Source Definition (OSD) of OSI[2] [3]. It's important to understand that impact of Open Source can not be entirely avoided by simply rejecting FOSS applications like word processors or database servers. In development it is very common to use open source libraries, widgets and components. Even if our security policy completely forbid open source in development, what can be a major burden per se, our commercial applications and libraries still may (and most likely will) include some open source components.

Therefore it is crucial to understand and properly handle the risk of Open Source in almost any context of information security. In this paper at we summarize the relevant concepts of information security risk management recommendations and standards, and emphasize different aspects of open source security regarding these concepts. Finally, some interesting vulnerabilities will be identified and possible risk security controls will be demonstrated through some examples.

Risk management principles

Different standards and frameworks exist to handle diverse forms of risk. Non-exhaustive list of mayor stakeholders can be COSO, ITIL, OCTAVE and COBIT in enterprise risk management [4], ISO 31000 family of standards relating to Risk management in general, ISO/IEC 27005 and ISO/IEC 27002:2005[3] standards in the field of security risk management or Department of Army's FM 5-19 Field Manual coping with military risk.

The nomenclature is often different, but frameworks usually describe risk management as a cycle of acts, parallel to the well known PDCA[4] management model in business. ISO variant of this model shown in fig. 1.

[1] Free (Libre) and Open Source Software – F(L)OSS

[2] Open Source Initiative – an organization dedicated to promoting open-source software

[3] ISO/IEC 27002:2005 has been revised by ISO/IEC 27002:2013

[4] Plan Do Check Act – (PDCA)

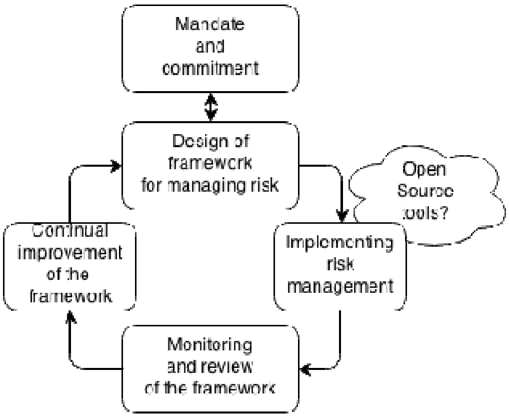
Mandate and commitment
Design of framework for managing risk
Open Source tools?
Implementing risk management
Monitoring and review of the framework
Continual improvement of the framework

Figure 1.: Risk management PDCA model
Source: G. Mészáros, based on ISO 31000 model

Here we can identify the first connection point to Open Source as a mere tool of risk management. Besides the numerous commercial solutions, several open source risk management softwares also has been developed including CRAMS, SOMAP.org and OpenGamma. Some of them can be valid option even for large financial institution [4].

However, organization may use many Open Source softwares like management and surveillance solutions, office applications, networking tools and so on. In this regard, open source can be considered a subject of IT risk management.

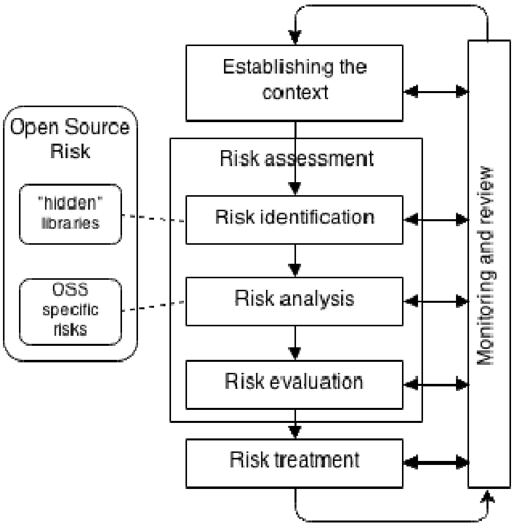

Figure 2.: Open source and risk assessment
Source: G. Mészáros, based on ISO 31000 model

Risk treatment can involve:

- avoidance, deciding not to start or continue with the activity;
- risk taking, by taking or even increasing risk in order to pursue opportunity;
- risk removal, by removing the risk source;
- changing the likelihood;
- changing the consequences;
- risk sharing;
- retaining the risk by informed choice.

As we can see one of the (quiet natural) risk treatment technique is the avoidance of risk. Unfortunately in case of Open Source this is not so obvious process. Impact of Open Source is constantly increasing [5], [6] and one point avoidance might be undesirable. Furthermore, without an accurate (and tedious) open source component exploration process, it might not be possible to detect open source libraries and widgets used in our commercial applications. If we use open source components in our organization either directly or indirectly that can affect overall security.

ISACA's Risk IT framework which complements COBIT[5] defines IT risk as a component of the overall risk universe of the enterprise. While in many enterprises IT-related risk is considered to be a component of operational risk, Risk IT Framework is against of depicting IT risk with a hierarchic dependency on one of the other risk categories [7]. In this interpretation IT risk is related to every categories to some degree as demonstrated in fig. 3.

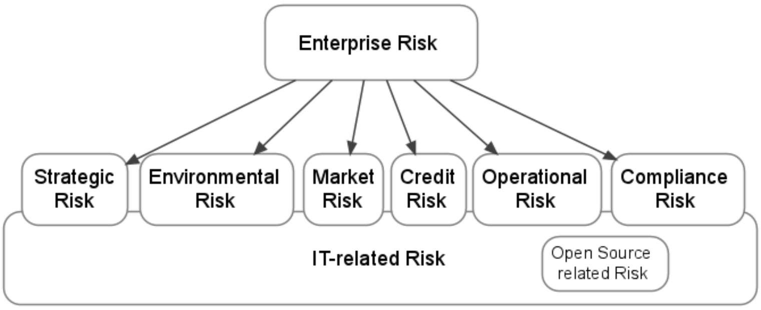

Figure 3.: Open source risk in IT according to ISACA

Source: G. Mészáros

In summary:

- FOSS can be considered as our tool of choice for risk management.
- Open Source components might be hard to identify (myriads of libraries and widgets exist and can appear in closed source);
- and may affect to wide range of risk types.

Considerations of Enterprise Risk Management

In the safety field, it is generally recognized that consequences are only negative and therefore the management of safety risk is focused on prevention and mitigation of harm [8]. In Enterprise Risk Management (ERM) risk is interpreted in a broader sense and it focuses on monetary aspects. According to the definition of ISO/IEC Guide 73:2002 risk is a combination of the probability of an event and its consequence [9]. Consequence need not necessarily be a negative effect. “Risk management and risk taking aren’t opposites, but two sides of the same coin.” [10]

The guidance created by the Federal Financial Institutions Examination Council identified Open Source risk factors related to these areas as follows [11]:

- Strategic Risks:
  - ability to customize;
  - compatibility and interoperability;
  - maturity;
  - forking;
  - systems integration and support;
  - total cost of ownership (TCO).
- Operational risks:
  - code integrity;
  - documentation;
  - contingency planning;
  - external support.
- Legal risks (considered under compliance in other sources):
  - licensing;
  - infringement;
  - warranties and indemnities.

[5] ISACA, C OBI T 4.1, 2008, www.isaca.org

As we can see, in this sense contacting with Open Source not only poses threats but might have "upsides", opportunities that can be exploited. In the business, missing an opportunity also can be considered a risk. Applying risk mitigation controls on Open Source also can induce upside. For example, joining to a community in order to control software vulnerabilities, can also give some control over the direction of development.

John J. Hampton in his book Fundamentals of Enterprise Risk Management mentions some special risk categories, where the risk owner can not be exactly identified: strategic risk, subculture risk, leadership and life cycle risk [12]. Though these categories are not security related, they might have security consequences. Regarding Open Source, subculture risk might be especially interesting. Different subcultures in the organization has different perception of values, and this may lead to tension or conflicts. Hampton recites Charles Handy's four types of subcultures: Bureaucratic Culture, Team Culture, Spider's Web Culture (all parties focus on the leader) and Individual Culture. Each culture has its own benefits and drawbacks.

Some modern organizations tend to follow the team culture, but most large organization is Bureaucratic. It can be an interesting research area which culture type the Open Source communities follow. In case of clash, joining to a community in order to mitigate or control risk might induce new risks.

Some enterprises is aware of the importance of Open Source risk management, and advances strategies were implemented. For example HP's whitepaper about best practices in open source governance describes a complete methodology consisting four main parts [13]:

- open source program office
- open source review board
- open source policy manual
- legal FOSS expertise.

This level of complex system can be required to properly handle open source systems. However by its own admission, HP reached a high but hardly perfect level of 4 in a scale of 5 in open source governance strategy [13].

In summary:

- Open source risk management can have an upside.
- Implementation advanced ERM methodologies over Open Source may uncover interesting risk factors.
- Some enterprises are already open source risk conscious.

Risk management , the Military Way

US Army risk management starts in '80s, and by the early 1990s, the Army established a goal to integrate risk management into all US Army processes [14]. Currently Field manual FM 5-19 superseding FM 100-14 define composite risk management to the military decision making process (MDMP) [15].

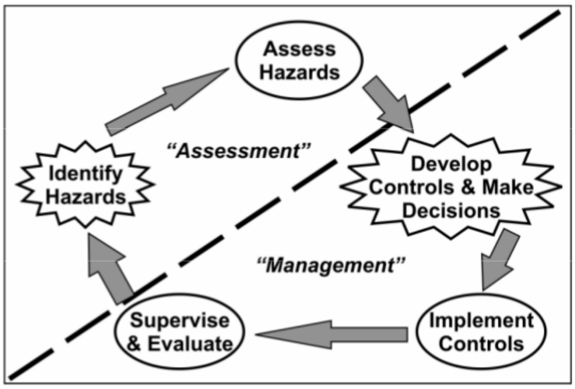

Figure 4.: Military risk management process overview
Source: FM 5-19 [15]

The model is similar in many aspects to the commercial solutions previously reviewed. The Manual defines simple and clear criteria to risk handling controls:

- Suitability: the control must remove the hazard or mitigate the residual risk to an acceptable level.
- Feasibility: the unit must have the capability to implement the control.
- Acceptability: the benefit gained by implementing the control must justify the cost in resources and time. (The assessment of acceptability is largely subjective).

Assessment factors used for hazard identification in nonmission (and mission) specific activities are the following:

- activity (mission);
- disrupters (enemy);
- terrain and weather;
- people (troops);
- time;
- legal (civilian considerations).

CRM does not differentiate between the sources of the hazard, only the effect of the hazard is important, not its source [15]. In this context risk factors of Open Source can be identified as:

- problems about the activity (software bugs, design problems etc.);
- enemy activity (crackers, spyware, agency attack etc.);
- people (conscious about threats of FOSS usage, distribution etc.);
- and legal (FOSS licenses, policy etc.)

Summary

In our paper we presented different views of risk management and their possible connections to Open Source. Primary goal of this research was to gain broader understanding in different risk management techniques and frameworks and identify possible relationships with open source software to support further research. The findings presented are excerpt of this work.

We showed that Open Source may influence multiple stages of Risk Management, albeit identification is not always obvious. Although our research field is security related, we find that important conclusions might be borrowed from the ERM. The military approach, while being excellent for the army needs, might be overly mission oriented for risk management of complex civil information environments like governmental organizations or critical infrastructures.

In our future work we will try to identify and organize the exact threat categories and related mitigation controls. For this purpose The FAIR taxonomy seems to be most appropriate, being security-oriented in origin, but the impact criteria apply to all IT-related risks, bridging the gap between ERM and IRM/ITsec [16].

References

[1] Alexis O’Connor, Kian Win Ong, Ted Sander, and Matt Ferlo, “Government Policies on Open Source.” 2010 [Online]. Available: http://www.cs.washington.edu/education/courses/csep590/04au/clearedprojects/Ferlo.pdf

[2] CIOinsight, “Open Source Turns Strategic,” CIO Insight, 2005. [Online]. Available: http://www.cioinsight.com/c/a/Research/Open-Source-Turns-Strategic/. [Accessed: 26-Oct-2012]

[3] “The Open Source Definition | Open Source Initiative.” [Online]. Available: http://opensource.org/docs/osd. [Accessed: 03-May-2013]

[4] Kosta Peric, “Open Source Software For Risk Analytics - A Valid Option,” Forbes, 2012. [Online]. Available: http://www.forbes.com/sites/kostaperic/2012/09/04/open-source-software-for-risk-analytics-a-valid-option/. [Accessed: 24-May-2014]

[5] Amit Deshpande and Dirk Riehle, “The Total Growth of Open Source,” in Proceedings of the Fourth Conference on Open Source Systems (OSS 2008), Springer Verlag, 2008, pp. 197–209 [Online]. Available: http://dirkriehle.com/publications/2008-2/the-total-growth-of-open-source/. [Accessed: 26-Mar-2014]

[6] Ø. Hauge, C. Ayala, and R. Conradi, “Adoption of open source software in software-intensive organizations–A systematic literature review,” Inf. Softw. Technol., vol. 52, no. 11, pp. 1133–1154, 2010.

[7] Information Systems Audit and Control Association, The risk IT framework principles, process details, management guidelines, maturity models. Rolling Meadows, IL: ISACA, 2009 [Online]. Available: http://www.isaca.org/Knowledge-Center/Research/ResearchDeliverables/Pages/The-Risk-IT-Framework.aspx. [Accessed: 23-May-2014]

[8] AIRMIC, ALARM, and IRM, “A Risk Management Standard,” Int. J., vol. 1, no. 1, 2002 [Online]. Available: http://www.ferma.eu/risk-management/standards/risk-management-standard/. [Accessed: 10-May-2014]

[9] ISO/IEC Guide 73:2002 Risk management - Vocabulary - Guidelines for use in standards, First edition. 2002.

[10] Michel Crouhy, Dan Galai, and Robert Mark, The Essentials of Risk Management. McGraw Hill, 2006.

[11] Federal Financial Institutions Examination Council, “Risk Management of Free and Open Source Software.” FFIEC [Online]. Available: http://www.federalreserve.gov/boarddocs/srletters/2004/SR0417a1.pdf. [Accessed: 02-May-2014]

[12] J. J. Hampton, Fundamentals of enterprise risk management how top companies assess risk, manage exposures, and seize opportunities. New York: American Management Association, 2009. ISBN: 978-0-8144-1492-7

[13] Bruno Cornec, “Open Source Governance,” on JDEV 2013. 13-Sep-2013 [Online]. Available: http://www.slideshare.net/eurolinux/governance-v25. [Accessed: 20-May-2014]

[14] Department of the army, “FM 100-14 Risk Management Field Manual,” 1998 [Online]. Available: https://archive.org/details/milmanual-fm-100-14-risk-management. [Accessed: 10-May-2014]

[15] Department of the army, “FM 5-19 Composite Risk Management,” 2006 [Online]. Available: http://www.cid.army.mil/documents/Safety/Safety%20References/FM%205-19%20Composite%20Risk%20Management.pdf. [Accessed: 10-May-2014]

[16] The Open Group, Technical Standard: Risk Taxonomy. The Open Group, 2009 [Online]. Available: www.opengroup.org/onlinepubs/9699919899/toc.pdf. [Accessed: 22-May-2014]

Having a comprehensive information security risk management (ISRM) strategy will help you overcome these challenges. Moreover, it will enable you to help senior management gain a better understanding of the organizationâ€™s current security posture and the wisdom of investing in data protection. In this post, I will share some tips about how to create an effective ISRM strategy and what a good program looks like.Â What makes a good information security risk management approach? As mentioned earlier, ISRM is an ongoing process of identifying, assessing, and responding to security risks. To manage risks effectively, organizations should evaluate the likelihood of events that can pose risk to the IT environment and the potential impact of each risk. Open Source Risk Engine (ORE) provides. contemporary risk analytics and value adjustments (XVAs). interfaces for trade/market data and system configuration (API and XML).Â Open Source Risk Engine is open source software, provided under the Modified BSD License, which permits using, modifying the code base as well as incorporating it into commercial applications. About. The Open Source Risk Engineâ€™s objective is to offer open source as the basis for risk modelling and analytics at financial institutions. It grew from work developed on QuantLib by market professionals and academics. ORE wants to take this to the next level. Quick Links.