International Journal of Trend in Research and Development, Volume 3(4), ISSN: 2394-9333
www.ijtrd.com

Risk Analysis in Software Architecture

[1]Sreenwanti Pal and [2]Sonali Adhikary and [3]Hema Gaikwad
[1.2]Student (MBA-IT), [3]Assistant Professor,
[1,2,3]Symbiosis Institute of Computer Studies and Research, Symbiosis International University, Pune, India

Abstract-- Assessment of architectural risk is a risk management process to map out flaws in a software architecture and determines risks to business information assets that arise from those flaws. Flaws are found through the process of architectural risk assessment, that reveal information assets to risk, risks are prioritized based on their impingement to the business, risk mitigation techniques are developed and applied, and the software is reassessed to determine the efficacy of the mitigations.

Keywords-- *Risk, Architectural Risk, Risk Management Framework*

I. INTRODUCTION

It begins with the terms in the Software Risk Assessment .The next section describes the actual process of risk management, which is broken down into various sections. The emphasis is on risk analysis. This document fit in the larger risk management framework. It mainly emphasises on architectural risk analysis of software threats and vulnerabilities and assessing their impacts on assets.

II. SOFTWARE RISK ASSESSMENT TERMINOLOGY

A. Information Assets

Risk management mainly focuses on information assets which must be protected. Security practitioners concern themselves with the availability, confidentiality, integrity, and auditability of information assets. Information assets vary in how critical they are to the business. It's the body of knowledge that is organized and managed as a single entity. An organization's information assets have financial value.

B. Threats

It always violates the protection of information assets of organization. A threat is an expression of intention to inflict evil injury or damage .It is a potential for harm. Examples of threats are: security auditing tools that probe potential vulnerabilities, hardware failures, performance delays etc. In most of the situations threats can't be controlled directly by the software system.

C. Vulnerabilities

Software can also be vulnerable because of a *lacuna* in the architecture. Flaws are fundamental failures which means that problem are always associated with software no matter how well it is implemented. SQL-injection attacks is one popular vulnerability. Input filtering routine quickly eliminates the problem and failure to authenticate between multiple cooperating applications is an architectural flaw. Unmitigated vulnerabilities require risk management planning to deal with impacts to assets.

D. Attacks

An attacker takes advantage of a vulnerability to threaten an asset.

E. Impacts

Impacts are consequences of successful attack. E.g. in terms of revenue. Brand reputation damage, loss of market share are the impacts to the company's marketing failure. Secondary software failures may incur more maintenance costs, more customer support cost, higher cost of development etc. To manage its risk at a more granular level software architectural risk analysis is performed.

F. Risks

Risk is the probability of a threat taking the advantage of a vulnerability and impact the company .Software architecture risk management identifies risks in software and then deals with them.

G. Mitigations

Mitigation of risk goes through the cycle of prioritizing, implementing, and maintaining the suitable risk-reducing measures. Mitigating is like changing the architecture of the software to reduce the impact of the risk.

H. Risk Analysis

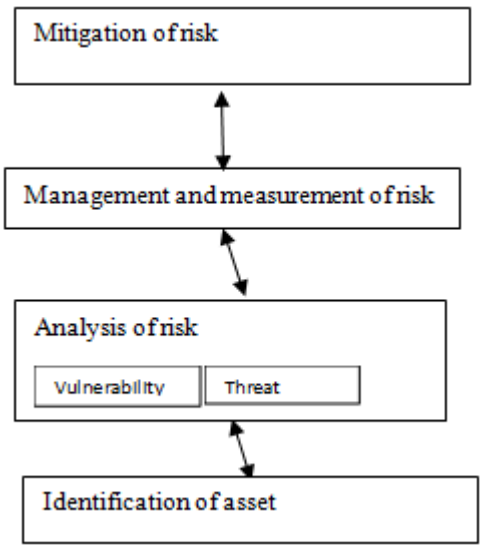
Risk analysis is the process of assessing and analysing risks. It is the iterative process.

I. Risk Management

Risk management is the process of managing risk using organization specific framework or policies that uses risk analysis, mitigations, metric to manage risk.

J. Software Architectural Risk Management

Risk management deals with assessing and addressing risk in software life cycle. It comprises of four processes: (1) management and measurement of risk, (2) analysis of risk, (3) mitigation of risk, and (4) identification of asset. In each phase, risk analysis is guided by business impact. The Software architectural risk analysis deals with identification and evaluation of risks and impacts and guidance of risk-reducing measures. Fig. below depicts the process of risk analysis and management.



Risk management continuously revaluates the business's risks from software throughout its lifetime. The table below (taken from NIST SP800-34 [2]) indicates the risk management activities at various times during the life cycle of a software system.

SLC Phase	Phase Characteristics	Risk Management Activities
Initiation	The need for software is expressed and the purpose and scope of the software is documented.	Information assets are identified. Business impacts related to violation of the information assets are identified.
Development or Acquisition	The software is designed, purchased, programmed, developed, or otherwise constructed.	The risks identified during this phase can be used to support the security analyses of the software and may lead to architecture or design trade-offs during development.
Implementation	The system security features are configured, enabled, tested, and verified.	The risk management process supports the assessment of the system. Decisions regarding risks identified must be made prior to system operation.
Operation or Maintenance	The system performs its functions. Typically the system is being modified on an ongoing basis through the addition of hardware and software and by changes to organizational processes, policies, and procedures.	Risk management activities are performed for periodic system reauthorization (or reaccreditation) or whenever major changes are made to the software in its operational, production environment (e.g., new features or functionality).

K. Identification of Assets

Risk management starts by identifying the assets which must be protected. Info assets can be customers personal info, company's financial information, order information to fulfil orders and collect revenue.

To identify information assets, one must look into the management which determines goals and constraint of software and the management can easily identify the business impact of failures. Information assets can be of type of databases, credentials, audit records, financial information, and intellectual Property etc.

L. Analysis of Risk

Analysis of risk is the second step in the risk management process. After threat identification and risk mapping this step guide for risk reduction and risk mitigation. Vulnerabilities leads to threat which further leads to risk and impact on information assets of organization. The risk assessment consists of six fundamental stages:

1. Application characterization
2. Architectural vulnerability assessment
3. Threat analysis
4. Risk likelihood determination
5. Risk impact determination
6. Risk mitigation

These activities are described below.

M. Characterization of Application

Architectural risks assessment is easier for well-defined boundaries, many artifacts are required for review. Following: Some system-level artifacts are useful in the architectural risk assessment process. These include SDLC artifacts, questionnaires and interviews are useful in gathering information. Policy documents, system documentation audit reports, system security plans etc. can also provide important information about the security controls used by and planned for the software.

• software business case • functional and non-functional requirements • enterprise architecture requirements • use case documents • software development plan • transactions	• quality assurance plan • test plan • risk management plan • software acceptance plan • problem resolution plan • risk list
• Documentation of the system and data criticality and sensitivity	• Information storage protection • Technical controls

• System security management controls used for the software	used for the software

N. Software Architectural Risk Analysis

Three activities are there in architectural risk analysis:

1. Vulnerability analysis,
2. Ambiguity analysis,
3. Platform vulnerability analysis.

O. Vulnerability Analysis

There are a lot of known vulnerabilities in software security literature. For example, a static code checker can display bugs like buffer overflows. It cannot identify security vulnerabilities like transitive trust.

P. Ambiguity Analysis

Ambiguity points towards a rich source of vulnerabilities when it exists between requirements or specifications and development.

For example, a web application might state that an administrator can lock an account

Q. Platform Vulnerability Analysis

Software architectural risk assessment include an analysis of the vulnerabilities associated with the application's execution environment. E.g. operating system vulnerabilities, network vulnerabilities etc.

R. Vulnerability Classification

Vulnerabilities classification allows for pattern recognition. Detailed seven vulnerability classes:

- incomplete parameter validation
- inconsistent parameter validation
- implicit sharing of privileged/confidential data
- asynchronous validation/inadequate serialization
- inadequate identification/authentication/authorization
- violable prohibition/limit
- exploitable logic error

S. Analysis of Threats

It violates the protection of information assets. Threats mapped to vulnerabilities helps to exploit the system.

The table below, which was developed by NIST [4, p. 14], summarizes potential threat sources:

Threat Source	Motivation	Threat Actions
Cracker	Challenge Ego Rebellion	• System profiling • Social engineering • System intrusion, break-ins • Unauthorized system access
Computer criminal	Destruction of information Illegal information disclosure Monetary gain Unauthorized data alteration	• Computer crime • Fraudulent act • Spoofing • System intrusion • Botnets • Malware: Trojan, virus, worm, spyware • Spam • Phishing
Terrorist	Blackmail Destruction	• System penetration • System tampering etc.
Industrial espionage	Competitive advantage Economic espionage Blackmail	• Economic exploitation • Social engineering • System penetration • Unauthorized system access

Insiders (poorly trained, disgruntled, etc.)	Curiosity Ego Lack of procedures or training	• Interception • System bugs • System intrusion • System sabotage • Unauthorized system access

T. Threats and Vulnerabilities Mapping

Threats and Vulnerabilities together always indicates system risk. Shirey [5] provides a model of risks to a computer system related to disclosure, deception, disruption, and usurpation. Threats may aims at the following risk classes:

1. Disclosure
2. Deception
3. Disruption
4. Usurpation: unauthorized system control access

U. Risk Determination approach

Risk prioritization and mitigation is useful after knowing software threats and vulnerabilities.

The following factors are best for estimation:

1. Threat capability and reach
2. Vulnerability impact
3. Current measure effectiveness

Threats levels can be described in the table below

High	The three qualities are all weak
Medium	One of the three qualities is compensating, but the others are not.
Low	Two or more of the three qualities are compensating.

V. Determination of Risk Impact

Risk impact determination is one of the important process. For risk impact determination we follow three measures:

First we identify threatened asset

Second we measure business impact

Third we determine the locality of impact

W. Risk Exposure Statement

The risk exposure statement includes risk occurrences with risk impact. The table indicate how to give risk exposure statement

Likelihood	High	Medium	High	High
	Medium	Low	Medium	High
	Low	Low	Low	Medium

	Low	Medium	High
	Impact		

There are some who would argue that the role of the bank supervisor is to minimize or even eliminate bank failure; but this view is mistaken in my judgment. The willingness to take risk is essential to the growth of the free market economy…[i]f all savers and their financial intermediaries invested in only risk-free assets, the potential for business growth would never be realized [6].

X. Mitigation of Risk

After knowing the risk what are the procedures, policies and standards we follow to to mitigate the risk associated.

Risk mitigation always incur cost so it should be proper and should follow specific rule base on the type of software architecture. Risk mitigation is a detailed process so skilled and trained personals are required for the same.

Risk mitigation is iterative in nature. It can also inject and introduce new types of risks and vulnerabilities in the system.

Y. Management and Measurement of Risk

We need to prioritize the risk based on its quality and mitigate it based on its priority. As all risks cannot be mitigated due to cost factor .Proper metrics are there to manage the risk in after the quantitative analysis based on its architecture. Andrew Jaquith [7] provides guidelines that security metrics must adhere to:

1. Be consistently measured.
2. Be cheap to gather
3. Contain units of measure.
4. Be expressed as a number.

CONCLUSION

In the paper we tried to find out what are the various software architectural risks. How to find the risks, vulnerabilities and threats of the software architecture system which is evolving mainly due to change in software. Ambiguity analysis is important for quantitative risk assessment.

Acknowledgement

Reviews by Hema Gaikawad of Symbiosis Institute Of Computer Studies And Research Pune. Errors and omissions are the authors'.

References

[1] Michelle Keeney, JD, PhD, et al. *Insider Threat Study: Computer System Sabotage in Critical Infrastructure Sectors*, May 2005, http://www.secretservice.gov/ntac_its.shtml.

[2] M. Swanson, A. Wohl, L. Pope, T. Grance, J. Hash, R. Thomas, "Contingency Planning Guide for Information Technology Systems," NIST (2001).

[3] R. Abbott, J.Chin, J. Donnelley, W. Konigsford, S. Tokubo, and D. Webb, "Security Analysis and Enhancements of Computer Operating Systems," Technical Report NBSIR 76-1041, ICET, National Bureau of Standards, Washington, DC 20234 (Apr. 1976).

[4] National Institute of Standards and Technology. Risk Management Guide for Information Technology Systems (NIST 800-30). http://csrc.nist.gov/publications/nistpubs/800-30/sp800-30.pdf (2002).

[5] R. Shirey, *Security Architecture for Internet Protocols: A Guide for Protocol Designs and Standards*, Internet Draft: draft-irtf-psrg-secarch-sect1-00.txt (Nov. 1994).

[6] Address to the Garn Institute of Finance, University of Utah, November 30, 1994.

[7] Andrew Jaquith, Yankee Group, CIO Asia, "A Few Good Metrics", http://cio-asia.com/ShowPage.aspx?pagetype=2&articleid=2560&pubid=5&issueid=63(link is external) (2005).