Software Security Total Risk Management

SECURITY INNOVATION'S BLUEPRINT FOR EFFECTIVE PROGRAM DEVELOPMENT

SecurityInnovation
THE SOFTWARE SECURITY COMPANY

187 Ballardvale Street, Wilmington, MA 01887 +1.978.694.1008 **www.securityinnovation.com**

Table of Contents

Introduction

Current challenges of the financial services sector aside, risk management has a long and venerable tradition of practical success in the world of insurance premiums and credit card interest rates. In the world of IT, however, the successful application of risk management techniques has been more elusive.

This problem has been no more apparent than in IT application and software development. Despite compelling industry statistics that enterprises lose billions of dollars annually and suffer loss of public confidence through application availability downtime or other security breaches, effective risk management techniques have yet to take hold.

This article examines some of the major challenges of software security risk management and introduces the concept of Software Security Total Risk Management (SSTRM), an innovative programmatic approach by which enterprises can apply software security development and assessment best practices in order to meet the twin goals of enhancing business revenues and protecting against business losses.

Why Software Security Risk Management Matters

In less than thirty years, IT has grown from a province of largely academic concern to a driving force in world commerce. This trend, which started with the advent of distributed computing architectures in the early 1980s, accelerated dramatically with the rise of Web-based computing in the 1990s. Today, doing business without a Web-based sales channel – even if only to supplement bricks-and-mortar operations – is virtually inconceivable.

With the rise of IT-enabled business has come an explosion of demand for IT products to support people, process and technology throughout the enterprise. These include: protocols, Web browsers, Web servers, application servers, database servers, virtualization servers, storage servers, storage attached networks, network devices, the myriad tools which support the management of these products and the hundreds of thousands of applications which automate the business processes of buying, selling and performing work.

The growth and importance of IT security has paralleled this rise in importance of IT as a business enabler and competitive differentiator such that today, the rubric “IT security” has come to represent all that guarantees IT will fulfill its promise to transform the way business is done. Along the way, the concept of “security” has grown from an initial guarantee of continuous IT availability to a set of much broader guarantees: data confidentiality and integrity, identity protection and even operational efficiency as it relates to business continuity.

What is often overlooked amid this phenomenal growth of IT and IT security is the software that lies at the heart of every IT product, every IT device and every IT application. Software code, written by software developers, is what ensures that every IT device and every software application does what it is supposed to do – including carrying out its security functions.

In other words, the whole edifice of enterprise IT is only as secure as the secure software development foundations on which it is built.

Why Software Security Risk Management Gets Overlooked

Software development has a long and storied history. Since the advent of the first programmable devices in the 1940s, software development was considered the exclusive province of mathematicians, musicians or magicians. Individual or group creators of programming languages, operating systems and even specialty algorithms joined a secret pantheon known only to the elect who subscribed to common principles of elegance and innovation.

Practitioners of software security were part of an even more select group, representing not only the perspective of the general software developer, but typically possessing an even stronger mathematical focus, as well as an added sense of mission around the guarantees of availability, confidentiality and integrity that are at the core of IT security.

The business mind – with its focus on the practicality of profit making – has not always been comfortable with the community of software developers. Thus, even as the alliance between business and IT has flourished, and especially in recent years as enterprises have grown and differentiated themselves into multiple lines of business with numerous layers of managerial hierarchy, responsibility for software development and software security has tended to be pushed deep into individual lines of business – far from the province of executive business decision makers.

In recent years, the requirements of regulatory compliance (e.g., Sarbanes-Oxley, PCI) have given new impetus to the drive for IT security in general, generating widespread interest in systematic IT Security Risk Management. But even with this push, the esoteric traditions of software development have tended to give this dimension of IT security less visibility than is merited by its critical role; and until today, there has been no systematic and effective framework for Software Security Risk Management.

To understand the broad requirements for such a framework, a brief digression into the foundations of IT Risk Management (ITRM) is in order.

Foundations of IT Risk Management

Many industry articles dealing with ITRM assume a grasp of foundational concepts and proceed to address the underlying mathematics of IT risk estimation. While rigorous methodology is certainly crucial to effective ITRM, delving into methodological details is not my purpose here; rather, I would like to suggest that ITRM *is a particular perspective on the relationship between business and IT*, and establish the broad outlines of that perspective.

Many people assume that every discussion of ITRM is really a discussion about IT security. This is a mistake. IT *security* risk management is one particular variant of ITRM, which brings numerous additional considerations and merits its own special treatment (there can even be numerous variants *within* IT security risk management, such as software security risk management – as this article has already suggested).

The true starting point for building effective ITRM programs lies with a clear understanding of the relationship between business and IT systems. Because at the end of the day, *ITRM amounts to nothing more than the process of creating a strategy which manages the relationship between business and IT* (at all three levels: people, process and technology) – and then executing on that strategy.

Everyone knows that business and IT have long enjoyed a love-hate relationship. They come from different places and represent different motivations. Business is as old as the market and is about buying and selling and coming out ahead. Information technology is as recent as the semiconductor and is about accelerating all sorts of processes – including business processes – through automation.

The desire to improve the science of automation is a key element driving IT – something IT does not share with business. But it is also what makes IT essential to business: *business capitalizes on the process improvements made possible by IT, in order to drive market efficiencies and create competitive advantages that lead to greater profits*.

Embracing the IT promise of process improvement does not come cheap, however. It requires capital investment in equipment and personnel, as well as long-term maintenance. IT also introduces something more – something much less tangible than cost – into the world of business process: the values essential to the successful pursuit of the science of automation, *which are to a degree in competition with the values of business*. Finally, and much more obviously, applying IT automation to enhance pre-existing business processes *makes business dependent on IT in order to keep those processes running smoothly*.

Operational Integration, Availability and Security Risk Management

The first stirrings of ITRM can be found in the need to balance the relationship between the goals of business profitability, on the one hand, and two new factors on the other: *IT operational integration* and IT system uptime, or *availability.* Each of these new factors accompanying the introduction of IT into business can seriously erode business profits and thus constitutes a major business *risk*.

As suggested above, security is a relative newcomer to the ITRM scene. Security is driven by the dramatically increased value of IT physical assets, other business assets (including intellectual property) and certain kinds of data pertaining to the privacy or confidentiality of business transactions involving customers, employees or business partners. With the addition of security risks, ITRM becomes IT *Security* Risk Management (ITSRM).

If ITSRM is a matter of creating a *strategy* that would somehow balance these risks against the promise of business profits, the next questions become: "How much risk should I worry about?" and "How do I actually put a risk management strategy together?"

Common and easy answers to these questions are: "Whatever amount makes you worry" and "Any way you can." After all, most people get through the much more complicated balancing act of life with not much more of a profit strategy than meeting basic financial obligations (like your monthly mortgage payments) and trying to manage some important risks (like health problems). We let the rest take care of itself.

But in the same way that IT has made a science out of improving processes through automation, other sciences, designed to help create effective strategies in certain more limited arenas of life, like the business world, have emerged in recent years. These are the sciences of *decision support* (DS).

DS is too big to try to describe in this paper, but one small aspect of it is essential to the practice of risk management. Closely related to the actuarial science that has long been applied successfully in the insurance industry, DS provides guidance in the creation of all kinds of business strategies by looking at various types of risk and attempting to quantify them precisely in terms of *costs.* In addition to the cost of the risk itself, DS has also developed for estimating the cost to *remediate* the risk.

If the costs of those IT-introduced risks identified above, as well as the cost to remediate the risks, can be rolled up into IT investment costs, and these costs can then be compared against the profits that one expects to realize by entering into business in the first place, a simple DS formula emerges that can serve as the basis for an ITSRM strategy:

If Benefit > Cost, then take suggested action

Of course, developing a sound risk management strategy is not merely a matter of following this simple dictum. True risk management strategy development involves choosing from among a number of strategic alternatives: risk avoidance, risk transfer, risk acceptance or risk remediation.
And, needless to say, appropriately and reliably quantifying the cost of risk and the true cost of remediating risks

are the most difficult parts of developing a DS-based risk management strategy. In the life insurance industry, there are plenty of statistics on which to base estimates of life expectancy or natural disaster; but in the world of IT risk management, accurate estimates of the risks to operational efficiency, IT system availability or data security are notoriously difficult to generate.

Some will say that that this is good enough reason not to try and that ITSRM should be abandoned as an unrealizable dream. But in many ways, IT risk cost estimates are in reality no more challenging to estimate than the expected profit from a business venture. Profit estimation entails huge uncertainties: in the size of market demand, in the buying preferences of consumers, in the effect of competition. Yet few would embark on a new business venture without a healthy respect for the risks they faced; there is no reason to treat the challenge of incorporating IT into business any differently.

From ITSRM to Software Security Total Risk Management

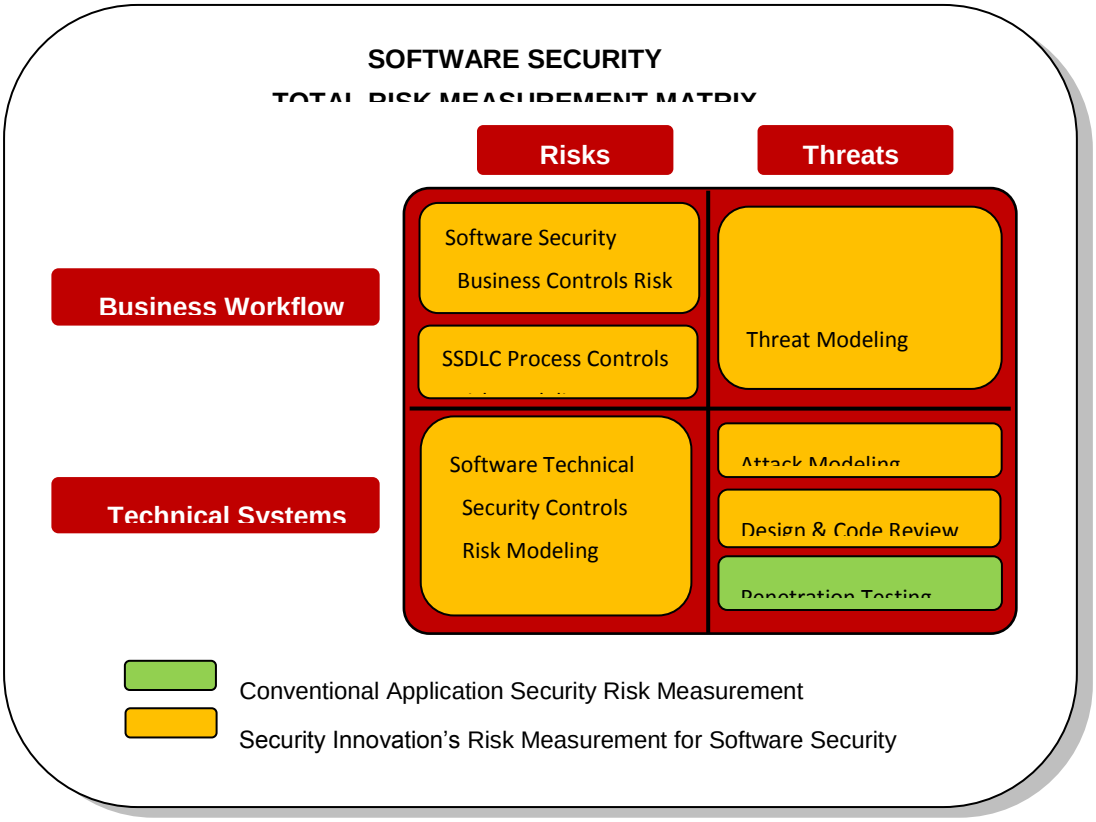
A recent industry breakthrough in developing a systematic approach to software security risk management is based on three important concepts: (1) an understanding of the role of software development and security within an enterprise; (2) an innovative approach to software security risk estimation, and (3) a fully-developed risk management process model, which can guide the enterprise through each essential step of an effective software security risk management program.

Building on the above simple model of ITSRM, Security Innovation's *Software Security Total Risk Management* (SSTRM) methodology represents a new, state-of-the-art approach that enables enterprises to more accurately assess software security vulnerabilities, prioritize them correctly, and develop a customized vulnerability remediation roadmap that will help manage business risk.

Conventional approaches to software security are not risk-based, typically encompassing no more than penetration testing of application functionality for some pre-determined set of common vulnerabilities. This approach frequently fails to address each application's unique code-, system- and workflow-level vulnerabilities. More importantly, it provides little practical guidance on prioritizing application defect remediation or creating a roadmap to guide enterprise software security posture improvements.

The Steps of the SSTRM

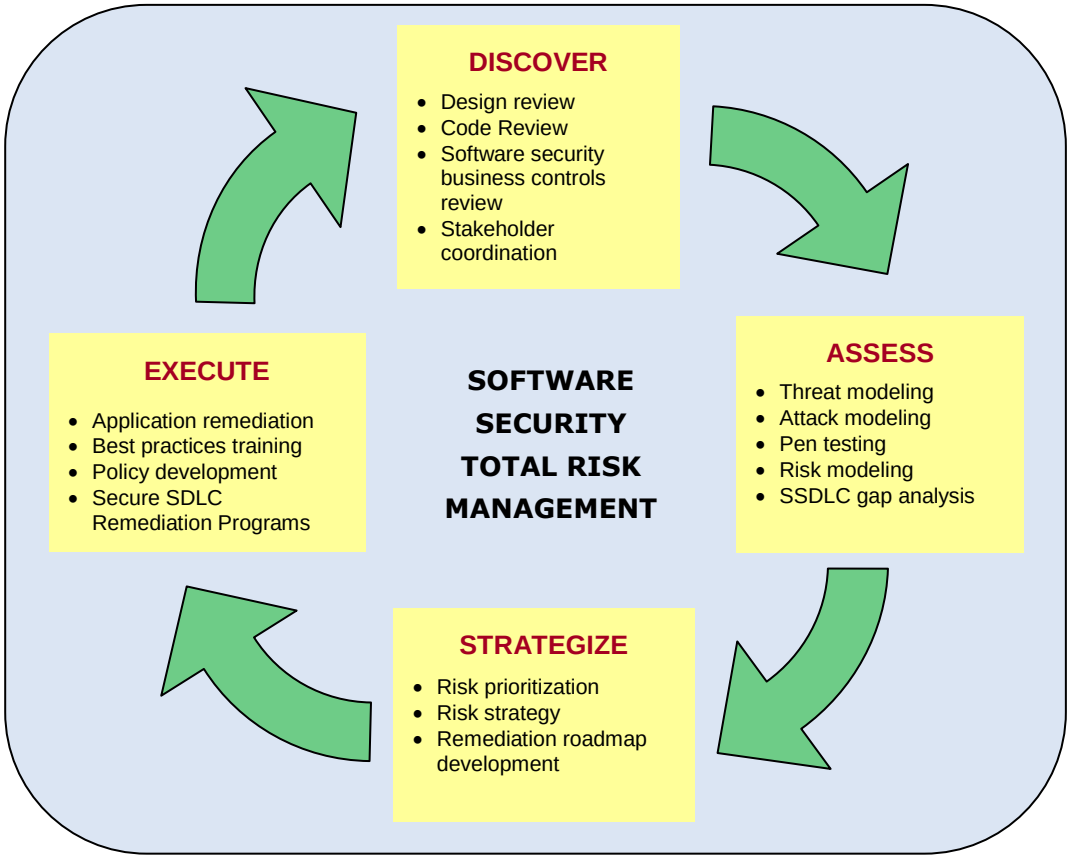
The SSTRM approach begins with a unified view of application security threats and risks at both business workflow and technical systems levels, to ensure that the business risk implications of application security vulnerabilities are correctly assessed. Different modeling techniques to address each threat and risk type are combined to augment the more conventional application penetration testing approach, as illustrated below.



Threat modeling of the application workflow, coupled with attack modeling and design/code review of the application implementation, traces all the ways in which application end-users and administrators might accidentally or intentionally exploit faulty application control logic or coding errors. These assessment activities yield a set of application software *vulnerabilities*, as well as estimates for the likelihood that the vulnerability will actually be exploited or otherwise come to pass.

Risk modeling, incorporating Secure Software Development Life Cycle (SSDLC) and other best practices drawn from such internationally-accepted standards as the ISO 2700x series and the Information Technology Infrastructure Library (ITIL), combines risk likelihood estimation (drawn from prior customer Business Impact Analysis or other appropriate asset valuation exercises) to ensure that application vulnerabilities are viewed in the broader risk context of business cost, including the costs of regulatory compliance, operational integration and IT availability.

The second part of the SSTRM approach applies the augmented risk measurement matrix within a best practices risk management process methodology incorporating the four basic steps of effective IT risk management: Discovery, Assessment, Strategy and Execution, as illustrated below.



Risk Discovery reviews software (application or systems-level technical specifications) in order to understand exactly how the technical system in question has been designed and deployed. In keeping with the risk measurement matrix requirements noted earlier, Discovery also includes the inventorying of application workflow requirements. Finally, stakeholder participation is carefully designed into the risk management process framework from the beginning in order to overcome the natural decoupling of the software development and security function from the executive risk management function and ensure that all intermediate steps in the process are validated before moving forward.

Risk Assessment marks the phase of the SSTRM approach which distinguishes the approach presented here from other traditional assessment techniques. While most organizations do little more than conduct "black box" penetration tests against software applications, Security Innovation applies threat methodology to identify high-priority software vulnerabilities that penetration testing alone may not find. Risk modeling takes the next step in quantifying software vulnerabilities in terms of business risk resulting from data breaches, loss of business IP or operational inefficiencies.

Risk Strategy prioritizes business risk due to software vulnerabilities, and then works with customers to choose among the set of risk management responses, including risk avoidance, transfer, remediation or acceptance. For those vulnerabilities requiring immediate remediation, the organization can work with a software security expert (consultant/vendor) to develop, in concert with stakeholders, a remediation "roadmap" consistent with the enterprise goals of an enhanced software security posture.

Execution performs the set of software security remediation activities that emerge from the strategy – from tactical software "patches" that fix high-priority vulnerabilities - to more strategic programs such as *Secure Software Development Life Cycle Best Practices*.

Summary

This article has introduced the concept of software security risk management as a special case of IT risk management. It summarized some of the characteristics of the relationship between the functions of business and IT software development that have hampered the developed of effective software risk management programs. The article also made the case that software security is, in principle, every bit as amenable to rigorous and effective risk management strategy planning as the other domains of IT Industry.