SECURE PROJECT MANAGEMENT

Governance and security - Adopting an enterprise software security framework - Security and project management - Maturity of Practice.

Definitions of Security Governance

- The term governance applied to any subject can have a wide range of interpretations and definitions.
- The term "security," as used here, includes software security, information security,application security, cyber security, network security, and information assurance.
- It does not include disciplines typically considered to reside within the domain of physical security, such as facilities,executive protection, and criminal investigations.
- The process of establishing and maintaining a framework and supporting management structure and processes to provide assurance that information security strategies
  - are aligned with and support business objectives
  -are consistent with applicable laws and regulations through adherence to policies and internal controls, and
  - provide assignment of responsibility, all in an effort to manage risk.

7.2.2. Characteristics of Effective Security Governance and Management

- One of the best measures that an organization is addressing security as a governance and management concern is a consistent and reinforcing set of beliefs, behaviors, capabilities, and actions that match up with security best practices and standards.
- These measures aid in building a security-conscious culture.
- They can be expressed as statements about the organization's current behavior and condition.

Characteristics:

- Security is managed as an enterprise issue, horizontally, vertically,and cross-functionally throughout the organization.

Executive leaders understand their accountability and responsibility with respect to security for the organization; for their stakeholders; for the communities they serve, including the Internet community; and for the protection of critical national infrastructures and economic and national security interests.

- Security is treated as a business requirement. It is considered a cost of doing business and perceived as an investment rather than an expense or a discretionary budget-line item.
- Security policy is set at the top of the organization with input from key stakeholders.Business units and staff are not allowed to decide unilaterally howmuch security they want. Adequate and sustained funding and allocation of adequate security resources are a given.
- Security is considered an integral part of normal strategic, capital,project, and operational planning cycles. Security has achievable,measurable objectives that are integrated into strategic and project plans and implemented with effective controls and metrics. Reviews and audits of plans identify security weaknesses and deficiencies and requirements for the continuity of operations and measure progress against plans of action and milestones.
- Security is addressed as part of any new project initiation, acquisition, or relationship and as part of ongoing project management. Security requirements are addressed throughout all system/software development life-cycle phases, including acquisition, initiation, requirements engineering, system architecture and design, development, testing, operations, maintenance, and retirement.

Adopting an enterprise software security framework:

- Most organizations no longer take for granted that their deployed
- applications are secure.
- But even after conducting penetration tests,network and hosting security personnel spend considerable time chasing incidents.
- How to build security into their software applications for a few years now.

7.3.1. Common Pitfalls

Whether tackling the problem formally or informally, top-down or bottomup, organizations hit the same roadblocks as they prepare to build and buy more secure applications.

Lack of Software Security Goals and Vision

- The first hurdle for software security is cultural. It's about how software resists attack, not how well you protect the environment in which the software is deployed.
- Organizations are beginning to absorb this concept, but they don't know exactly what to do about it.

.

Creating a New Group

- Some organizations respond to the software security problem by creating a group to address it.
- Headcount and attention are necessary, but it's a mistake to place this headcount on an island by itself.
- Software security resources must be placed into development teams and seen as advocates for security, integration, and overcoming development roadblocks.

Software Security Best Practices Nonexistent

- Security analysts won't be much more effective than penetration testing tools if they don't know what to look for when they analyze software architecture and code.
- Instead, build technology-specific prescriptive guidance for developers.
- If the guidance doesn't explain exactly what to do and how to do it, it's not specific enough. Specific guidance removes the guesswork from the developer's mind and solves the problem of consistency between
  security analysts.

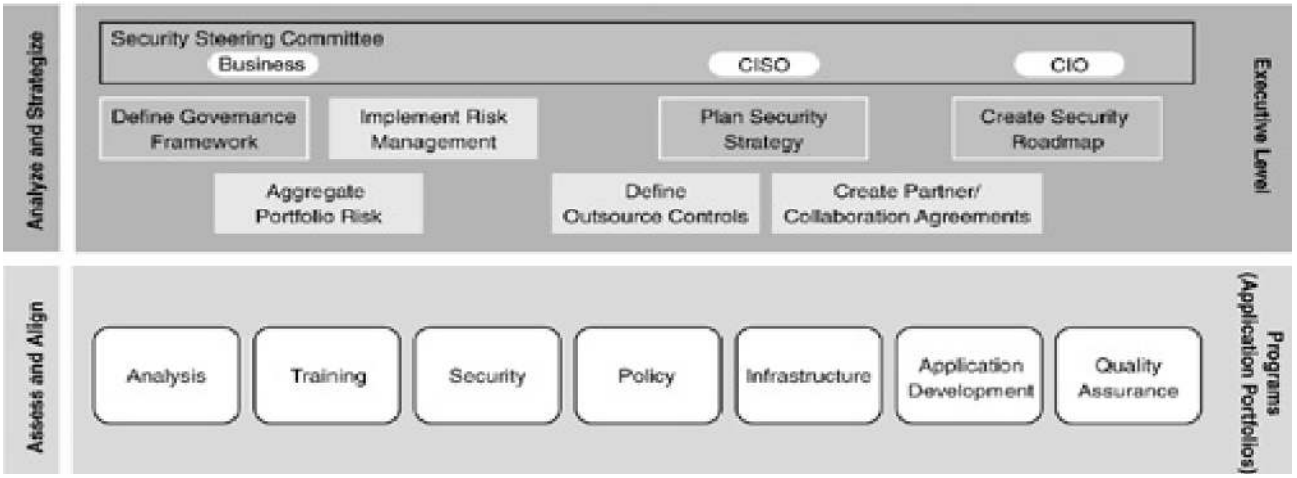
Software Risk Doesn't Support Decision Making

- Although most organizations view critical security risks as having the utmost importance, project managers constantly struggle to apply risk management techniques.
- Even if a technical vulnerability is identified, analysts often don't fully understand its probability and impact.

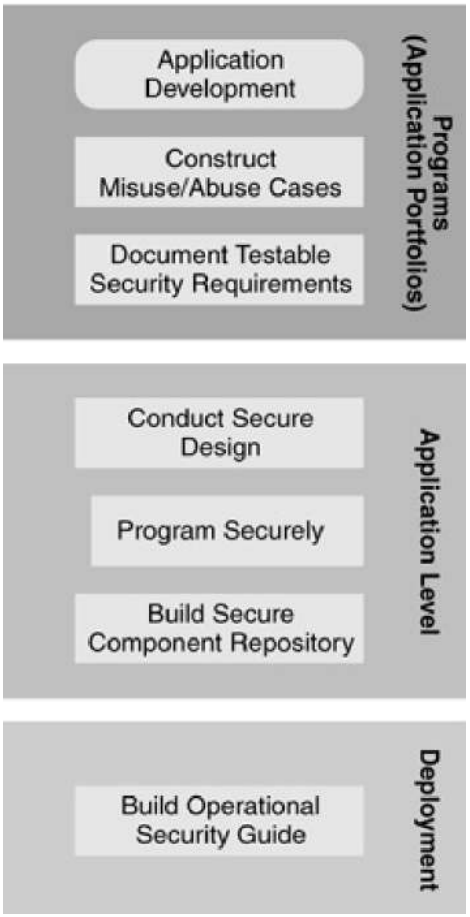
- Rarely does an organization use a risk-management framework to consistently calculate a risk's impact at the project-management or portfolio level.
- Establish a common risk framework as part of governance efforts to gain business owners' understanding and respect if you want the organization to choose security risk over time-to-market or if you need additional capital when making release decisions.

7.3.2. Framing the Solution

- An enterprise software security framework (ESSF) is a new way of hinking about software security more completely at the enterprise level, targeting the problem directly without demands for massive headcount,role changes, or turning an IT shop upside down to prioritize security ahead of supporting the business that funds it.
- Because every organization possesses different strengths and weaknesses and, most important, faces different risks as a result of using software.



"Who, What, When" Structure



- Construct misuse/abuse cases.
- Model the threats each application faces.
- Assess applications against a threat model, including misuse/abuse cases.
- Train using vulnerability case studies.
- Define standards based on risk and vulnerabilities.
- Avoid relying on security-feature checklists.
- Avoid relying solely on API-guide security standards and training.
- Like adopting framework activities

7.3.3. Define a Roadmap

Each competency depends somewhat on the others, and growing each effectively demands thoughtful collaboration. It's foolish to attempt to understand all the subtle interdependencies from the start and attempt a "big bang" rollout.

Patience.

- It will take at least three to five years to create a working, evolving software security machine.

- Initial organization-wide successes can be shown within a year.
- Use that time to obtain more buy-in and a bigger budget, and target getting each pursuit into the toddler stage within the three-year timeframe.

Customers.

- The customers are the software groups that support the
- organization's lines of business.
- Each milestone in the roadmap should represent a value provided to the development organization,
  not another hurdle.