CERT'S PODCASTS: SECURITY FOR BUSINESS LEADERS: SHOW NOTES

Becoming a Smart Buyer of Software

Key Message: Managing software that is developed by an outside organization can be more challenging than building it yourself.

Executive Summary

Acquiring or purchasing software from a reputable outside party is often viewed as being more cost effective. Relying on outside experts means you don't have to build and manage your own software development capability. That said, acquisition program managers must have experience in all aspects of system and software development, as well as solid people skills. These are necessary to effectively manage your relationship with a software contractor throughout the development life cycle and to ensure that their product can be successfully deployed in your operational environment.

In this podcast, Brian Gallagher, director of the Acquisition Support Program at the SEI, discusses what business leaders need to know when acquiring or purchasing software, along with implications for security.

PART 1: BUYING VS. BUILDING SOFTWARE

Buying Software

- is viewed as being more cost effective
- doesn't require an organizational capability in software development or IT systems, and the management attention to oversee it
- relies on outside experts and specialists who understand system and IT development and what it takes to provide system and IT services

Buying Software Is Not As Easy As You May Think

That said, there are more similarities than differences when buying software vs. building it. An acquirer or purchaser of software can't just turn over the keys to an outside developer. They need to:

- understand business needs, threats, and opportunities, and how to translate these into requirements
- provide leadership throughout the engagement
- transition the new system into the business

So acquisition program managers need to understand system engineering, architecture, leadership, transition, maintenance, and support activities – and that they are still responsible for ensuring success of all of these activities.

A common misunderstanding is that when you outsource software development, you no longer need to be knowledgeable or responsible.

Retain a Lean, Agile Management Layer

Managers need to ensure that business needs are continuously communicated to the developing partner throughout the life cycle – and that the partner addresses them effectively. Needs will evolve and change.

Big disconnects can occur when the partner organization is ready to install and transition the new system into the business. The requirements and challenges are often underestimated and underappreciated.

Acquisition program managers must have experience in all aspects of system/software development, along with solid people and organizational change skills.

Life Cycle Considerations

A good working definition of acquisition is moving from "I need" or "I want" to "I got."

This is not solely about contracting or purchasing. It is cradle to grave, which is about:

- being able to describe the needs of the organization in a form that someone can build on
- getting a contract that reflects those needs
- establishing a win-win relationship
- ensuring an effective change management process within the buying organization to make sure the new system can be successfully deployed
- eventual disposal or termination of the new system when it is no longer useful or has been replaced

There are many opportunities for challenges and mistakes along the way such as:

- translating operational needs and threats to be addressed into accurate requirements
- making sure developer interpretations and assumptions are correct
- orchestrating multiple contractors, suppliers, and vendors

Managing an outside relationship with another organization can be much more challenging than managing folks within your own shop.

PART 2: ACQUIRING SOFTWARE WITH SECURITY IN MIND

Addressing Security as a Software Acquisition Requirement

To identify security requirements, ask some key questions, such as:

- What is the operational need (physical security, data security, etc.)?
- What type of environment will the new system need to integrate with?
- What is the developers' ability to build high quality software that is free of vulnerabilities?
- Are there second or third tier vendors that will be participating (suppliers to suppliers, sub-sub contractors)? If so, how do you know that the licensing is correct?
- How will you know that the system functions as intended when delivered (acceptance criteria)?
- Does the developer have the required knowledge, skills, and abilities? This needs to be part of the selection process.
- Have you sufficiently addressed quality assurance, systems assurance, and software assurance during the development process?
- Has the developer done this kind of job before? Are there examples of past performance?

Useful Approaches for Managing Secure Software Acquisition

Consider the following:

- operational threat analysis and being able to translate anticipated threats into contractual requirements
- SEI's Quality Attribute Workshop can be helpful in understanding how to specify quality attributes such as performance, reliability, dependability, and security.
- defining use cases and scenarios that exercise and reveal the extent of quality attributes

An Example Scenario

Consider a handheld system used for situational awareness by a soldier in the field – getting an accurate picture of the battlefield. Questions to consider include:

- How do you make sure the data the soldier is displaying is fresh, current, and up-to-date?
- How are you going to provide updates to the software on the handheld device?
- What precautions do you need to take if the handheld device falls into the hands of the enemy? How do you protect its information from being used against you?

Capture these types of requirements in your statement of work and your technical requirements specification. Make sure to adequately address quality (non-functional) attributes.

Statements such as "the system must be secure" or "the system must inter-operate" are not sufficient. Scenarios can be used to help describe expected outcomes and under defined conditions.

Monitoring and Review – for Both the Developer and the Acquirer

Make sure to conduct regular product reviews and technical reviews.

Ensure that the developers' process includes solid coding standards and practices, and a high degree of quality assurance to make sure that the system is as free of defects and vulnerabilities as possible.

In other words, focus on both product and process.

Acquirers often overlook what vulnerabilities they may introduce into the development process. For example, purchasing organizations need to protect design information and documentation that is entrusted to their care to make sure it does not get into unauthorized hands.

Resources

CERT Secure Coding Practices

CERT Insider Threat Practices, including those that specifically address the software development life cycle.

Department of Homeland Security's Build Security In (BSI) web site

BSI Acquisition content area

Creel, Rita. Assuring Software Systems Security: Life Cycle Considerations for Government Acquisitions." Software Engineering Institute, Carnegie Mellon University, June 2007.

SEI's Acquisition Support Program