ISSN 2250-3153

Software Development Life Cycle Processes with Secure

Ashok Kumar Gottipalla, N.M.S.Desai, M.Sudhakar Reddy

Uppal Hyderabad Ranga Reddy (Dt) Pin code: 500039

Abstract- It is to be to present the information about existing processes, standards, life cycle models, frameworks, and methodologies that support or could support secure software development. This includes software engineering process group (SEPG) members, software developers, and managers seeing information about existing software development life cycle (SDLC) processes that address security.

Index Terms- SDLC processes, security Risk Identification, security engineering activities.

I. INTRODUCTION

The purpose of is to collect and present overview information about existing processes, standards, life cycle models, frameworks, and methodologies that support or could support secure software development. Where applicable and possible, some evaluation or judgment may be provided for particular life cycle models, processes, frameworks, and methodologies.

The target for this includes software engineering process group (SEPG) members who want to integrate security into their standard software development processes. It is also relevant for developers and managers looking for information on existing software development life cycle (SDLC) processes that address security. Technology or content areas described include existing frameworks and standards such as the Capability Maturity Model® Integration (CMMI®) framework, the FAA-iCMM, the Trusted CMM/Trusted Software Methodology (T-CMM/TSM), the Systems Security Engineering Capability Maturity Model (SSE-CMM), in addition to existing processes such as the Microsoft Trustworthy Computing Software Development Lifecycle, the Team Software Process SM for Secure Software Development (TSPSM-Secure), Correctness by Construction, Agile Methods, and the Common Criteria.

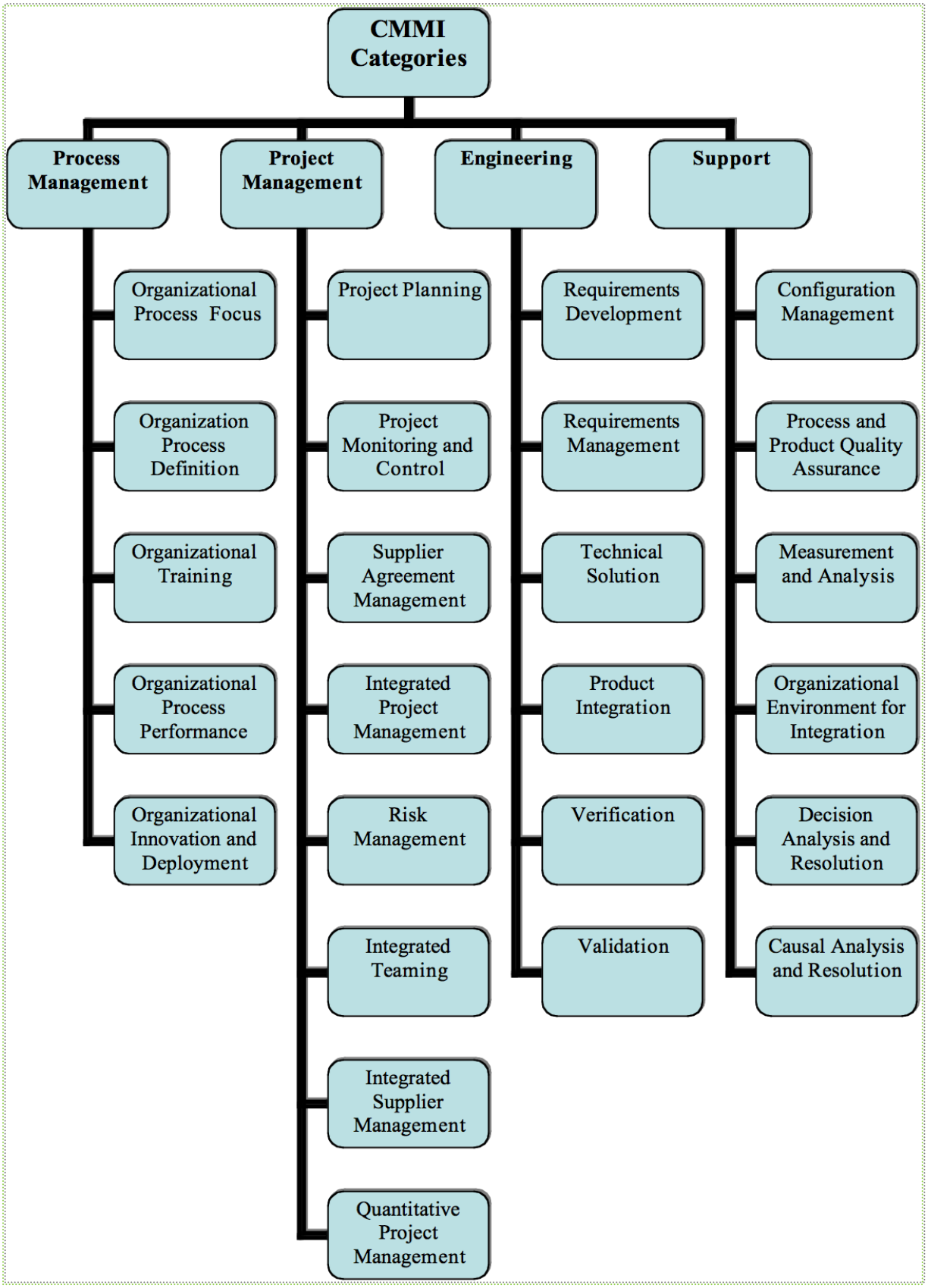
Capability Maturity Models (CMMs)

Capability Maturity Models provide a reference model of mature practices for a specified engineering discipline. An organization can compare their practices to the model to identify potential areas for improvement. The CMMs provide goal-level definitions for and key attributes of specific processes (software engineering, systems engineering, security engineering), but do not generally provide operational guidance for performing the work. In other words, they don't define processes, they define process characteristics; they define the what, but not the how:

"CMM-based evaluations are not meant to replace product evaluation or system certification. Rather, organizational evaluations are meant to focus process improvement efforts on weaknesses identified in particular process areas"

Capability Maturity Model Integration (CMMI)

The Capability Maturity Model Integration (CMMI) framework helps organizations increase the maturity of their processes to improve long-term business performance. The CMMI provides the latest best practices for product and service development, maintenance, and acquisition, including mechanisms to help organizations improve their processes and provides criteria for evaluating process capability and process maturity. Improvement areas covered by this model include systems engineering, software engineering, integrated product and process development, supplier sourcing, and acquisition. The CMMI has been in use for more than three years and will eventually replace its predecessor, the Capability Maturity Model for Software (SW-CMM), which has been in use since the mid-1980s. As of June 2005, the Software Engineering Institute (SEI) reports that 782 organizations and 3250 projects have reported results from CMMI-based appraisals [SEI 05a]. Beginning in 1987 through June 2005, 2,859 organizations and 15,634 projects have reported results from SW-CMM-based appraisals and assessments [SEI05b]. The CMMI addresses four categories for process improvement and evaluation. Each category includes several Process Areas. As shown in Figure 1, the CMMI addresses project management, supplier management, organization-level process improvement as well as training, quality assurance, measurement, and engineering practices. However, it does not specifically address the four areas mentioned earlier (security risk management, security engineering practices, security assurance, and project/organizational processes for security), although it is not unreasonable to assume that each of these are special cases of practices already addressed by the CMMI.


CMMI Categories
Process Management
Project Management
Engineering
Support
Organizational Process Focus
Organization Process Definition
Organizational Training
Organizational Process Performance
Organizational Innovation and Deployment
Project Planning
Project Monitoring and Control
Supplier Agreement Management
Integrated Project Management
Risk Management
Integrated Teaming
Integrated Supplier Management
Quantitative Project Management
Requirements Development
Requirements Management
Technical Solution
Product Integration
Verification
Validation
Configuration Management
Process and Product Quality Assurance
Measurement and Analysis
Organizational Environment for Integration
Decision Analysis and Resolution
Causal Analysis and Resolution

.

II. Conclusion

Other key standards and methods that apply to developing secure software but have not been summarized in this paper include

- ISO/IEC 15288 for System Life Cycle Processes,
- ISO/IEC 12207 for Software Life Cycle Processes
- ISO/IEC 15026 for System and Software Integrity Levels
- Clean room Software Engineering.

This Research paper demonstrates that although there are several processes and methodologies that could support secure software development, very few are designed specifically to address software security from the ground up. The notable exceptions are Microsoft's Trustworthy Computing SDL and the SSE-CMM. As software security becomes a more important issue in an increasingly networked world, more processes that explicitly address the four focus areas identified in this paper (security engineering activities, security assurance activities, security organizational and project management activities, and security risk identification and management activities) should achieve visibility.

References

[1] The Agile Alliance. Manifesto for Agile Software Development. http://agilemanifesto.org (2001).

[2] Beznosov, Konstantin. eXtreme Security Engineering: On Employing XP Practices to Achieve 'Good Enough Security' without Defining It. http://konstantin.beznosov.net/professional/papers /eXtreme_Security_Engineering.html (2003).

[3] Beznosov, Konstantin & Kruchten, Philippe. Towards Agile Security Assurance. http://konstantin.beznosov.net/professional/papers /Towards_Agile_Security_Assurance.html (2004).

[4] Common Criteria. http://www.commoncriteriaportal.org/ (2005).

[5] Common Vulnerabilities and Exposures. http://www.cve.mitre.org/ (2005).

[6] The Federal Aviation Administration Integrated Capability Maturity Model® (FAA-iCMM®), Version 2.0. Washington, DC: Federal Aviation Administration, September 2001. http://www.faa.gov/aio/common/documents/iCMM /FAA-iCMMv2.htm

[7] Lipner, Steve & Howard, Michael. The Trustworthy Computing Security Development Lifecycle. http://msdn.microsoft.com/security/default.aspx?pull=/library /en-us/dnsecure/html/sdl.asp (2005).

[8] Microsoft. Microsoft Security Advisories. http://www.microsoft.com/technet/security/advisory/default.mspx (2005).

[9] Software Engineering Institute. Process Maturity Profile: CMMI v1.1 SCAMPI v1.1 Class A Appraisal Results, 2005 Mid-Year Update. Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, September 2005. http://www.sei.cmu.edu /appraisal-program/profile/pdf/CMMI/2005sepCMMI.pdf

[10] Wäyrynen, J.; Bodén, M.; & Boström, G. "Security Engineering and eXtreme Programming: an Impossible Marriage?" Proceedings of XP/Agile Universe 2004: 4th Conference on Extreme Programming and Agile Methods, Calgary, Canada, August 15–18, 2004. Germany: Springer-Verlag, 2004.

Authors

First Author – Ashok Kumar Gottipalla, Hno:-3-107/1 Behind Sahithya High school Mallikarjuna Nagar Uppal Depot Uppal Hyderabad Ranga Reddy (Dt) Pin code: 500039, 4) Phone number: +91 80962 80964, 5) Email address: ashokkumar.wishes@gmail.com
Second Author – N.M.S.Desai
Third Author – M.Sudhakar Reddy