Security by Design with CMMI for Development, Version 1.3

An Application Guide for Improving Processes for Secure Products

Siemens AG
Corporate Technology
May 2013
# Table of Contents

**Executive Summary**  
3

**Abstract**  
6

1 **Introduction**  
7

**Background and Acknowledgements**  
Version 1.3  
7

**Purpose and Scope**  
8

**Relationships with CMMI and Security Standards**  
9

**Structure of the Security Extension**  
11

**Intended Audiences**  
11
   - Organizations Undertaking Process Improvement  
   - Members of Process Improvement Groups  
   - Security Experts  
   - Appraisers and Organizations Seeking to Benchmark Security Processes  
12

**Usage Scenarios and Use of CMMI-DEV Concepts**  
13
   - Usage Scenarios  
   - Use of CMMI Concepts  
   - CMMI Framework Interactions  
14

**Process Improvement Considerations**  
15

**Process Appraisal Considerations**  
15

2 **The Process Areas and the Generic Goals and Generic Practices**  
17

**Organizational Preparedness for Secure Development (OPSD)**  
18

**Security Management in Projects (SMP)**  
28

**Security Requirements and Technical Solution (SRTS)**  
41

**Security Verification and Validation (SVV)**  
51

**Generic Goals and Generic Practices**  
59

**Appendix A:** References  
72

**Appendix B:** Acronyms  
76

**Appendix C:** Glossary  
78

**Appendix D:** Relationship of Security by Design Practices to CMMI-DEV v1.3 Practices  
80
Executive Summary

Developing secure products requires security-specific techniques, skills, experience, and capabilities within an organization. The organization also requires appropriate processes that integrate these techniques and capabilities for a sustainable effort to develop secure products beyond trial-and-error. The objective of this application guide is to establish components of a process model to be used for creating and improving such processes. Four process areas are used in this guide to define characteristics for the development process. This includes two process areas for security aspects of engineering, one for managing security in projects, and one for organizational security topics. By implementing these process areas, the organization adapts the workflows, processes, and process assets, for example by introducing new or modifying existing process activities, milestones, roles, methods or tools. By these activities, the organization creates and acquires products that are secure by design. The structure of the guide is shown in Fig. 1.

Figure 1: Security by Design defines security characteristics for the development process

This application guide is a set of additional process areas for CMMI® (Capability Maturity Model® Integration) for Development, Version 1.3 (CMMI-DEV, V1.3) [SEI 2010 a]. These process areas integrate seamlessly into the Process Management, Project Management, and Engineering process area categories of CMMI-DEV. An overview of these security process areas is shown in Table 1.
More and more incidents occur that are caused by hacker attacks and data theft and have effects on important facilities. It is desired to develop products to withstand such attacks by design and not by chance. As a result, more and more security requirements are imposed by customers, industry standards, and governments (or other regulatory bodies).

This guide can be supportive in conjunction with product security standards that are also defining requirements on the development process.

<table>
<thead>
<tr>
<th>Security Process Area</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Organizational Preparedness for Secure Development</td>
<td>The purpose of Organizational Preparedness for Secure Development (OPSD) is to establish and maintain capabilities to develop secure products and react to reported vulnerabilities.</td>
</tr>
<tr>
<td>Security Management in Projects</td>
<td>The purpose of Security Management in Projects (SMP) is to establish, identify, plan, and manage security-related activities across the project lifecycle and to manage product security risks.</td>
</tr>
<tr>
<td>Security Requirements and Technical Solution</td>
<td>The purpose of Security Requirements and Technical Solution (SRTS) is to establish security requirements and a secure design and to ensure the implementation of a secure product.</td>
</tr>
<tr>
<td>Security Verification and Validation</td>
<td>The purpose of Security Verification and Validation (SVV) is to ensure that selected work products meet their specified security requirements and to demonstrate that the product or product component fulfills the security expectations when placed in its intended operational environment.</td>
</tr>
</tbody>
</table>

Table 1: Overview of the Security Process Areas.

An organization augmenting its processes and organizational structures to conform with these security process areas demonstrates that it has established product security for the entire development lifecycle. Instead of ad-hoc measures in the case of reported vulnerabilities, the whole development organization is capable of developing and purchasing products that are secure by definition and design. Relevant stakeholders (e.g., senior management, product management, development, commissioning) have gained awareness, understanding, and knowledge of product security.

Organizations using these security process areas to improve their development processes are better able to effectively address customer security requirements. By implementing these security process areas, security process assets and work products (e.g., security policy, secure coding guidelines, results
of security code reviews, security assessment reports) are created that help to provide security assurance for the customer.
Abstract

Best practices for security exist today but have not often been incorporated into a standard or model that would provide guidance to a broader development community. CMMI-DEV provides a framework within which security activities can take place but major guidance of security aspects are missing. Security by Design with CMMI for Development, Version 1.3 is an application guide for improving processes that lead to secure products.

This application guide was developed using similar methods as the +SAFE extension [SEI 2007 a] to CMMI-DEV, V1.2 [SEI 2006 a], which is meant to extend CMMI-DEV with safety attributes. The CMMI for Services (versions 1.2 and 1.3) also has an available application for security during service consisting of a single security management process area [Forrester et al 2009 and 2010]. The Security by Design application guide consists of four additional process areas to CMMI-DEV providing an explicit and focused basis for improving or appraising an organization’s capabilities for developing products with an adequate security level.

This application guide can be used in combination with CMMI-DEV, V1.3 or as an independent collection of goals and practices, if intended to address security topics in an organization that already uses mature or capable processes.
1 Introduction

Security by Design with CMMI for Development Version 1.3 is an application guide for CMMI for Development, Version 1.3 (CMMI-DEV, V1.3) and is intended to support the development and maintenance of secure products.

The application guide presents security-specific material for improving the capability of an organization to develop and maintain secure products.

Background and Acknowledgements

This application guide was developed by the Corporate Technology (CT) department at Siemens AG to enable the improvement and appraisals of an organization’s ability to develop secure products for the following reasons:

- Many development organizations address security in product development by some uncorrelated activities such as using coding guidelines or penetration testing in late phases. These isolated activities create undesirable results, leading to late or even missing detection of security vulnerabilities. The late detection of security vulnerabilities makes it often too expensive to correct defects since this usually requires a rework of the product architecture or design. As a consequence, ad hoc approaches often result in products deployed with severe security risks.

- Many companies are using CMMI-DEV and associated appraisal methods to improve and appraise the internal development processes. Although CMMI models provide a framework in which security engineering can take place, CMMI-DEV V1.3 references to “security” exist only in a few sentences in the informative material. Security can be interpreted, but major guidance is missing.

Version 1.3

The knowledge for Security by Design was obtained mainly from long-standing experience with security in product development and from various best-practice documents (for more details see section Relationships with CMMI-DEV and Security Standards). During the development of this guide, security experts and process consulting experts worked together. The results have been published [Fich 2011a], [Pan 2011a].
Siemens’ Corporate Technology (CT) department acknowledges the following contributing authors (in alphabetical order):

- Martin Baur (Siemens AG, Corporate Technology, Enterprise Processes Consulting)
- Irina Dietrich (Siemens AG, Corporate Technology, Enterprise Processes Consulting)
- Barbara Fichtinger (Siemens AG, Corporate Technology, Cyber Emergency Readiness Team)
- Peter Panholzer (Siemens AG, Corporate Technology, Cyber Emergency Readiness Team)
- Winfried Russwurm (Siemens AG, Corporate Technology, Enterprise Processes Consulting)

We thank the SEI and the CMMI Institute for supporting this project through valuable discussions and comments and for sharing their expertise with process improvement. In particular we want to thank the following people (in alphabetical order):

- Mary Beth Chrissis
- Jay Douglass
- Pat Kirwan
- Mike Konrad
- Mike Phillips

Moreover, we thank the following persons for their contributions during review (in alphabetical order): Doug Ashbaugh, Michael Barnett, Bradley J. Bittorf, Joachim Bauchrowitz, Leia R. Bowers, Mike Campo, Jeanine Courtney-Clark, Gerhard Fessler, Bernd Grobauer, Ronda Henning, Christian Hertneck, Kent A. Johnson, Richard LI, Klaus Lukas, Mukul Madan, Jorge Marquez, Francis Mayer, Nancy Mead, Winifred Menezes, Thomas Mosel, Michele Moss, Marcia A. Sommers, Ute Rosenbaum, Janne Uusilehto, Priyamvadha Vembar, Dorna Witkowski, Pieter van Zyl.

The approach used in the +SAFE extension [SEI 2007 a] created by the Defence Materiel Organisation, Australian Department of Defence has been emulated in this application guide. Also, some phrases from +SAFE have been used here.

**Purpose and Scope**

The purpose of Security by Design with CMMI for Development, Version 1.3 is to provide a structured framework for secure product development. It does not replace or modify material from an existing CMMI constellation.

The application guide uses most CMMI-DEV V1.3 model components. The security knowledge is structured into process
areas with specific goals and practices as well as generic goals and practices (see Figure 2).

Security by Design can generally be used in the same way as CMMI:
- As a framework for appraising and improving an organization’s capability in developing secure products.
- As a framework for appraising the capability of a supplier of security-critical products.

Relationships with CMMI and Security Standards

Numerous security standards and other material deal with the development of secure products. This section of the documents outlines how this material relates to Security by Design and will help the reader to understand the purpose and intention of the material and how it could be used with this application guide, or how it is different from this guide.

Security by Design consists of four additional process areas integrating seamlessly into the Process Management, Project Management, and Engineering process area categories of CMMI-DEV. Higher level cross-references to other process areas are also indicated, since other CMMI-DEV process areas could indirectly address security as a nonfunctional (quality) attribute of a product by the interpretations in the informative material.

Important characteristics of some best practice approaches (e.g., Microsoft Security Development Lifecycle [Mic 2011] or Comprehensive, Lightweight Application Security Process [Owa 2007]) are listed in the subpractices of Security by Design. To implement the specific and generic goals of this guide, those
best practice documents may provide additional implementation guidance.

Security by Design supports the fulfillment of requirements for the development process from e.g. Common Criteria for Information Technology Security Evaluation [CCMB 2009] and emerging industry standards such as the Product Development Requirements [ISA 201x] of the ISA-99 committee.

It is not the primary intent of Security by Design to be used as part of a product security assessment or compliance benchmark with requirements catalogues. Examples of industry or application specific requirements catalogues include, for example, the Cyber Security Standards for Critical Infrastructure Protection of the North American Electric Reliability Corporation, [NERC 2011 a] and [NERC 2011 b], Security Requirements for Vendors of the International Instrument User’s Association [WIB 2010], the European Privacy Seal [ULD 2010], or the PCI Data Security Standard of the Security Standards Council [SSC 2010]. However, products developed by an organization with a security enhanced development process are more likely and confident to comply with industry or application specific requirements since it can be ensured that the requirements are met efficiently by an early and comprehensive integration throughout the entire development lifecycle.

ISO/IEC 27001:2005 [ISO 2005] is a standard used for implementing an Information Security Management System (ISMS) within an organization ensuring an adequate level of information security. If the development organization is certified according to ISO 27001, the implementation might support specific practices of Security by Design targeting the secure work environment.

Moreover, this application guide suggests conducting a product security risk assessment to support the identification of product security requirements. The methodology for such a risk assessment as defined in ISO/IEC 27005:2011 [ISO 2011] might be useful during implementation.

Security by Design does not have any direct relationship to or overlap with the CERT Resilience Management Model (RMM), Version 1.0 [SEI 2010 c]. The main focus of RMM is organizational resilience and not the development of secure products. Products developed by an organization applying Security by Design have met some prerequisites to be used in an organization aligning its processes according to RMM.

The Systems Security Engineering Capability Maturity Model® (SSE-CMM), which became ISO/IEC 21827:2008 [ISO 2008] is a full process model for development, including security. Implementing in environments already using other models such as CMMI-DEV may contribute to the “multi-model quagmire”

+SAFE [SEI 2007] is a safety extension, developed by the Australian Defense Materiel Organization. It covers safety, but not security. This application guide has taken the approach of +SAFE, but addresses security content.

### Structure of Security by Design

The structure of the security extension is shown in Table 1.1.

<table>
<thead>
<tr>
<th>CMMI-DEV Process Area Category</th>
<th>Security Process Area</th>
<th>Specific Goals</th>
</tr>
</thead>
<tbody>
<tr>
<td>Process Management</td>
<td>Organizational Preparedness for Secure Development</td>
<td>SG1 Establish an Organizational Capability to Develop Secure Products</td>
</tr>
<tr>
<td>Project Management</td>
<td>Security Management in Projects</td>
<td>SG1 Prepare and Manage Project Activities for Security</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SG2 Manage Product Security Risks</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SG2 Implement the Secure Design</td>
</tr>
<tr>
<td>Engineering</td>
<td>Security Verification and Validation</td>
<td>SG1 Perform Security Verification</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SG2 Perform Security Validation</td>
</tr>
</tbody>
</table>

Table 1.1: Detailed Structure of the Security Application Guide

### Intended Audiences

Security by Design is intended to be used primarily by the following audiences:

- Organizations undertaking process improvement to improve the security activities in product development
- Members of process improvement groups supporting organizations to integrate security in the development processes
- Security experts who want to foster process integration of security activities
• SCAMPI lead appraisers and appraisal team members

Organizations Undertaking Process Improvement

For organizations undertaking process improvement efforts, Security by Design provides guidance on how to improve development processes in an organization to create secure products. Specific practices in each process area describe what is expected to satisfy the goals. The informative material provides guidance and examples on how to implement the practices.

An improvement program embracing security needs collaboration of process experts and security experts.

An organization can take better advantage of an effective implementation of Security by Design if it has already implemented processes based on CMMI-DEV. The security process areas found in this guide may also be integrated in a CMMI-DEV process improvement effort.

Members of Process Improvement Groups

Members of process improvement groups supporting organizational units for a better integration of security activities in the organization and its processes can use Security by Design for detailed guidance.

Security Experts

For improvement programs, security experts may develop appropriate improvement plans for security processes and also support during the implementation. The creation of related process assets (e.g., process definitions, activities, methods, tools) might be facilitated by security experts.

Security experts intending to integrate security activities sustainably in the organization and development process should have adequate process improvement know-how.

In appraisals, they may participate as appraisal team members focusing on security practices and work products. Security experts may also develop appropriate improvement recommendations in alignment with Security by Design.

Appraisers and Organizations Seeking to Benchmark Security Processes

Security by Design, together with CMMI-DEV, may serve as a reference model for conducting appraisals in an organizational unit to benchmark security capabilities of its development process and to identify areas for improvement.

Lead appraisers and team members using this security extension are trained in the SCAMPI method. In addition, security knowledge is very helpful to understand the context; therefore some additional training for security or experience in
security is highly recommended. See “Process Appraisal Considerations” for more information.

Usage Scenarios and Use of CMMI-DEV Concepts

Usage Scenarios
Security by Design can be used independently and in combination with CMMI-DEV process areas.

Usage in Combination with CMMI-DEV
For the full benefit of the Security by Design guide and effective security practices, an established process context and mature processes in an organization are indispensable. The CMMI-DEV process areas are an excellent basis for such processes. Therefore, it is recommended to either have a significant number of CMMI-DEV process areas already in place before applying security process areas, or implement CMMI-DEV process areas and security process areas concurrently.

Independent Usage
Security by Design can be used independently as a collection of goals and practices if it is intended to address security topics in an organization that already uses mature processes. Due to the relationship with CMMI-DEV, this guide assumes that users are familiar with the CMMI framework, terminology, and conventions, and are able to reference relevant sections of CMMI-DEV as required. The material in CMMI-DEV V1.3 Part 1 sections 1-5 and Part 3, The Appendices, should be used to understand the security extension.

Use of CMMI Concepts
Some key CMMI concepts need to be understood before using the Security by Design guide effectively.

Process Areas
The CMMI supports the improvement and appraisal of each of the process areas described. Processes can be improved through the achievement of specific and generic goals as proposed by CMMI. Security by Design provides guidance in specific goals and practices and generic goals and elaborated generic practices for each of its process areas. All process areas may be used to appraise organizational processes and identify strengths and weaknesses.

Conventions and Terms
CMMI defines a range of terminology, including terms specific to CMMI models, and other words that have a special meaning in CMMI models. These terms are defined in Parts 1 and 3 of
CMMI-DEV and are also used in Security by Design. Therefore they are not explained here again, nor repeated in the glossary.

Required, Expected, and Informative Content

Content in CMMI models is classified as either “required” (i.e., specific and generic goals), “expected” (i.e., specific and generic practices), or “informative” (e.g., subpractices, elaborations). The Security by Design application guide classifies content in the same way, and organizations may adopt practices, goals, and informative material found in this guide as in the CMMI models, using the same principles of guidance as provided by the CMMI models.

CMMI Framework Interactions

Part 1 of the CMMI-DEV model shows interactions and interdependencies among process areas in process area categories, and between process areas and capability levels. Security process areas are assigned to process area categories in CMMI-DEV in recognition of some of these relationships.

The Security by Design guide also creates some new relationships due to the design decision to structure the extension as a set of separate process areas:

- The process areas in Security by Design relate to and build on some process areas in CMMI-DEV. Figure 3 illustrates the relationships. See appendix D for a more detailed overview of the relationship of Security by Design practices to CMMI-DEV V1.3 practices.

  It is strongly recommended to adopt the Security by Design process areas together with the related process areas in CMMI-DEV, and also to adopt strongly interdependent process areas in CMMI-DEV (e.g., PMC together with PP, OPD together with OPF and IPM).

- The CMMI-DEV support process areas apply to and interrelate with the security process areas as they do with other CMMI-DEV process areas.

- The CMMI-DEV generic practices apply to and interrelate with the security process areas as they do with other CMMI-DEV process areas.
Process Improvement Considerations

Security processes are highly dependent on the quality of the implementation of most CMMI-DEV process areas. Organizations are strongly encouraged to use Security by Design for process improvement together with the CMMI-DEV as an integrated effort.

Process improvement efforts that integrate CMMI-DEV, CMMI-AQ, CMMI-SVC, and Security by Design will explore more synergies, spread process improvement across more processes than just development, and save costs by avoiding the “multi-model quagmire.”

Security by Design, as a process reference model, does not require the use of specific security standards. If an organization has selected such specific standards, or if these standards are imposed by a contract, law, or regulatory bodies, Security by Design is intended to accommodate the methods and techniques of the standard, including, where applicable, alternative practices.

Process Appraisal Considerations

Security by Design can be used with any of the appraisal methods applicable to CMMI (e.g., SCAMPI A Method Description Document [MDD] V1.3 [SEI 2011 a], or SCAMPI B/C v1.1 [SEI 2005 a] as appropriate). The security process areas in this application guide may be appraised as part of a broader appraisal that uses CMMI-DEV as a process reference model, or independently (i.e. not using other CMMI process areas) in the same way as any CMMI process area(s).

If the organizational unit creates products with a certain level of security and needs to improve security related processes, this application guide should be beneficial.
The development of secure or non-secure products may also be a sampling factor for an appraisal according to SCAMPI MDD V1.3. This may lead to the appraisal of basic units with and without security needs.

A mapping should be created prior to the appraisal that illustrates the relationships between the organization’s security terminology, and the terminology used in this extension.

As above, process improvement efforts that integrate CMMI-DEV, CMMI-ACQ, CMMI-SVC, and Security by Design will explore more synergies. For performing a SCAMPI A appraisal including multiple CMMI constellations, refer to the SCAMPI A V1.3 MDD [SEI 2011 a].

Currently, the results of a SCAMPI appraisal performed using only process areas cannot be formally reported to the CMMI Steward. The SAS (SEI Appraisal System) system and PARS (Published Appraisal Results List) cannot be used.

If one or more CMMI process areas (e.g. CMMI-DEV V1.3) are appraised using a SCAMPI method (e.g. SCAMPI A V1.3), one or more security process areas can be added using the continuous representation. If a maturity level is required, the rules for equivalent staging as in CMMI-DEV part 1 are applied. This means that security process areas are not included in any maturity level rating. Results for security process areas may be added in text fields that appear in SAS, ADS, and PARS.
2 The Process Areas and the Generic Goals and Generic Practices

This chapter consists of the process areas with the specific goals and practices and the generic goal interpretation for Security by Design.
ORGANIZATIONAL PREPAREDNESS FOR SECURE DEVELOPMENT (OPSD)
A Process Management Process Area

Purpose

The purpose of Organizational Preparedness for Secure Development (OPSD) is to establish and maintain capabilities to develop secure products and to react to reported vulnerabilities.

Introductory Notes

This process area builds on practices of the Organizational Process Definition process area (OPD) and the Organizational Training process area (OT) in CMMI-DEV V1.3. This process area adds security related practices.

The development of secure products needs organizational capabilities to be established. These capabilities will help management and staff maintaining a consistent full lifecycle perspective across all security relevant projects in the organization. Adequately addressing security requires additional activities, characteristics, and capabilities at the organizational level that go beyond what CMMI addresses, for example in OPD or OT process areas. These security activities are incorporated into the organization’s set of standard processes and process assets to ensure that they are an integral part of all work activities. Sponsorship and commitment from higher level management is visible at all levels of the organization. For the development of secure products, the work environment standards have special requirements, for example for tools and increased protection against unauthorized access. Communication and knowledge are also key elements of organizational preparedness. Therefore, special attention is directed to security awareness.

Vulnerability handling is of particular interest to product users and other stakeholders to be able to report security issues in an organized way.

Secure development process improvements are incorporated into the organization’s process improvement plan. Security-related process improvements are prioritized and implemented along with other organizational process improvements.

Related Process Areas

Refer to the Organizational Process Definition process area of CMMI-DEV V1.3 for more information about organizational process assets.
Refer to the Organizational Process Focus process area of CMMI-DEV V1.3 for more information about planning, implementing, and deploying organizational process improvements.

Refer to the Organizational Training process area of CMMI-DEV V1.3 for more information about identification and delivery of training.

Specific Goal and Practice Summary

SG 1 Establish an Organizational Capability to Develop Secure Products

| SP 1.1 | Obtain Management Commitment and Sponsorship for Security and Security Business Objectives |
| SP 1.2 | Establish Standard Processes and other Process Assets for Secure Development |
| SP 1.3 | Establish Awareness, Knowledge, and Skills for Product Security |
| SP 1.4 | Establish Secure Work Environment Standards |
| SP 1.5 | Establish Vulnerability Handling |

Specific Practices by Goal

SG 1 Establish an Organizational Capability to Develop Secure Products

An organizational capability to develop secure products and react to security vulnerabilities is established and maintained.

SP 1.1 Obtain Management Commitment and Sponsorship for Security and Security Business Objectives

Establish and maintain commitment, sponsorship, and involvement from higher level management for secure product development, security business objectives, and process needs.

Risks introduced by the development of insufficiently secure products have an extraordinary impact on the business of an organization. Secure product development presents an even higher challenge for the quality and adequacy of processes compared to development of non-secure products. Therefore sponsorship, commitment and involvement from higher level management are essential and need to be visible and incorporated in the day-to-day business. Security-related business objectives and process needs are established. Product security policies are one of the key instruments for managing clear expectations and translating business objectives into tangible directives to the organization. Sponsorship and commitment must be visible, for example in the definition of process needs and objectives, organizational objectives, resources, extraordinary attention, participation, involvement, and leadership. Commitment, sponsorship, and involvement also means that the generic practices are implemented in a proactive way (e.g., GP2.1, 2.2, 2.3, 2.4, 2.5, 2.8, 2.9, and GP2.10).
Example Work Products
1. Product security policy
2. Product security charter outlining management expectations and contributions regarding security
3. Business objectives, process needs, organizational objectives, and process improvement objectives related to security
4. Plans to involve higher management proactively in security related activities.
5. Organizational process needs for secure development

Subpractices
1. Obtain visible and proactive sponsorship of security activities.
   Security awareness is present with higher level management and is proactively communicated. Higher level management provides resources, budget, leadership, participation, and training.
3. Provide higher level management support for planning and executing security-related processes.
   This support includes ensuring that appropriate levels of sponsorship, commitment, and attention to security are at all levels of the organization.
4. Define escalation paths for security-related issues
   Not all decisions should be done by top level management. A mandate to make decisions should be clearly agreed and communicated to all levels.
   Refer to the Decision Analysis and Resolution process area of CMMI-DEV V1.3 for more information about methodical decision making.
5. Create and communicate business objectives, process needs, organizational objectives, and process improvement objectives related to security and the development of secure products to staff.
6. Ensure proactive participation by higher level management in key activities related to security. Priorities to address organizational process needs for improving security performance.
   Proactive participation in key activities includes participating in reviews of the status of security issues, threats, and risks. Proactive participation requires sustained, visible leadership in key security-related activities.
7. Sustain special attention and interest from higher level management in secure development and take immediate and high-priority action on security topics. Review, negotiate, and obtain commitment for business security-related objectives and associated process need priorities from relevant stakeholders.

Higher level management participates in reviews of process improvement activities, objective evaluations, security issues, risks, threats, vulnerabilities, and incidents (as they may occur anywhere in the product lifecycle). This requires attention not just during the project lifecycle, but over the product lifecycle. Identify process improvements to the organization’s processes and process assets to address prioritized security-related process needs.

SP 1.2 Establish Standard Processes and Process Assets for Secure Development

*Establish and maintain standard processes and related organizational process assets for secure product development.*

For secure development, different or additional standard processes and process assets are needed compared to the development of non-secure products. Standard processes and other organizational process assets that expressly address these needs are specifically designed for the development of secure products. An organizational process asset library holds these standard processes, methods, procedures, templates, measurements, and training material. Appropriate life-cycle models for secure product development (e.g., including appropriate milestones), work environment standards, tailoring guidelines, etc. are also available.

Examples of processes for which standard processes and process assets are created include the following:

- Product security risk management
- Security requirements engineering
- Security reviews (e.g., architecture reviews, code reviews, chip design reviews)
- Security testing
- Defect management
- Security validation
- Secure coding

Example Work Products

1. Organization’s set of standard processes for secure development

2. Tailoring guidelines and criteria for secure development and process assets used for secure development
3. Organizational measurement repository with common set of measures for secure development (e.g., number of externally reported vulnerabilities, number of security defects found with static code analyzer, common risk levels so that a common dashboard can compare relative risk across projects)

4. Additional or modified lifecycle models covering the needs of the development of secure products

5. Other organizational process assets for secure development (e.g., training material, templates, sample documents, checklists, etc.)

Subpractices

1. Initiate the creation of standard processes and process assets for secure development.

   Refer to the Organizational Process Focus process area of CMMI-DEV V1.3 for more information about setting the process needs.

2. Ensure that already existing process elements have a sufficient level of detail and usability to support their consistent application for the development of secure products.

   Standard processes for secure development require explicit attention and are distinguishable from other assets. Therefore, the existing standard processes and organizational process assets are analyzed to obtain information about their fitness for secure development.

3. Create the standard processes and organizational process assets for secure development.

   Process assets are comprised of tailoring guidelines and criteria, common measures for secure development, checklists, templates, lifecycle models, external security standards, etc. Ensure that a sufficient level of detail, usability, and support are obtained when creating above standard processes and process assets.
Examples of tailoring criteria for secure development include the following:

- Business case criticality
- Records of previous security vulnerabilities in the product
- Exposure of the product, e.g., internet accessible
- Sensitivity of handled data, e.g., personally identifiable data
- Criticality of product when placed in operational environment
- Maturity of technology
- Development language chosen

Examples of process assets for secure development include the following:

- Security requirements baseline
- Secure architecture and design standards
- Secure coding standards
- Enterprise policy that identifies explicit implementation guidance for the development of secure applications
- Enterprise policy that identifies vendor/supplier agreement and supply chain management procedures that “build security in.”

4. Ensure that the standard processes and process assets for secure development have well established and usable interfaces to other processes (e.g., development, manufacturing, service, enterprise risk management) in the organization.

    Secure development-related processes are appropriately integrated with other processes in the organization.

5. Ensure that the process assets for secure development satisfy the process needs and objectives of the organization and comply with applicable standards, models, regulations and laws, particularly as applied for secure development (e.g. PCI, ASCI SDSA).

SP 1.3 Establish Awareness, Knowledge, and Skills for Product Security

*Establish and maintain stakeholder awareness, knowledge, and skills for developing secure products.*

Refer to the Organizational Training process area of CMMI-DEV V1.3 for more information about developing knowledge and skills.

Numerous stakeholders are involved in security topics. This comprises staff concerned with development of secure products (e.g., engineers, project managers, quality assurance staff, configuration management staff), management at all levels, customers, end-users, etc. As a major and strategic element of preparing an organization for the development of secure products, all relevant stakeholders are aware of security topics, methods, techniques, issues, processes, etc. Awareness,
knowledge, and common vocabulary are created. Stakeholders receive training or other mechanisms to develop their awareness, knowledge, and skills to prepare them to perform their involvement and roles appropriately. Training to develop knowledge and skills is usually targeted at relevant stakeholders, including higher level management and those directly involved in the development of secure products to enable them to perform their security-related roles and responsibilities.

**Example Work Products**
1. Secure development training plans
2. Awareness plans
3. Security training material and material to create security awareness
4. Attendance records and certificates for security training

**Subpractices**
1. Identify topics that require security awareness and security training, for example secure coding trainings.
   Plan stakeholder certification for security topics (e.g., secure coding certifications for developers).
2. Identify the stakeholders for awareness and security training.
3. Relate awareness and security training levels or methods to the respective stakeholders and roles.
4. Create training and awareness material for security topics.
5. Define information paths and methods to convey security awareness information.
6. Deliver trainings for secure development, obtain training records, and determine effectiveness.
7. Deliver and implement awareness and training methods (e.g., workshops, brochures distribution, email distribution, web site with security material).
8. Update the training and awareness material as needed. Include updated and new standards, process models, laws, and knowledge.
9. Deliver refresher trainings and repeat training and awareness as needed and maintain certifications.

The need for this may be identified from the evaluation of training effectiveness.
Establish Secure Work Environment Standards

Establish and maintain work environment standards to enable secure development and to protect work products from unauthorized use.

Work environment standards for the development of secure products extend those for non-secure products. Secure work environment standards enable secure development, include security guidelines, and protect the work products created by development (e.g., source code, architecture and design, vulnerability information) from unauthorized access and use (e.g., manipulation, theft). Work environment standards include guidelines for tailoring to allow adaptation for the project’s work environment to meet specific needs. The work environment may also call for external standards for security (e.g., from regulatory bodies).

Example Work Products
1. Work environment standards for secure development
2. Guidelines for access rights, actual access, etc.
3. Information security policy

Subpractices
1. Create work environment standards that enable the development of secure products.

   This includes appropriate tools to create and test work products (e.g., certified compilers, static code analyzers), appropriate rooms and equipment (e.g., alarms, shielding, physical access control), protected work stations and servers (e.g., virus scanners, patch management, firewalls) and protected communication (e.g., encrypted transmission) between various sites and stakeholders.

2. Create information security policy and information security guidelines (e.g., classification of data, handling of sensitive data, password policies, change management, separation of duties, backup handling).

   Identify and use legal and other regulations and security standards (e.g. ISO/IEC 27001:2005 [ISO 2005]) where appropriate.

3. Establish special controls for the access to sensitive work products (e.g., password protected code repository, encryption) and guidelines for access control.

4. Evaluate security of work environment, methods and tools.

   Refer to the Configuration Management process area of CMMI-DEV V1.3 for more information about access rights for work products.

Establish Vulnerability Handling

Establish and maintain organizational vulnerability handling.
All stakeholders (e.g., engineers, end users, customers) can report vulnerabilities that are identified during operation using an established infrastructure. Reported vulnerabilities are assessed and fixed by the organization in a timely manner to prevent further damage to the organization and its customers. All reported vulnerabilities are stored in a dedicated repository. The product security risk list and the product security risk management plan are updated to ensure coverage of similar vulnerabilities by the product security risk management where appropriate. This is mainly an organizational task, since coordination across all projects, products, and lifecycles (product, project) is needed.

Vulnerability handling significantly expands what is usually seen to be within a development process. It takes into account that the product needs to be maintained for security issues even after its completion. Therefore, it is not just done during the life cycle of the development project, but operated through defined phases of the product lifecycle.

While this is handled within the processes for product development, another approach to handle it would be where the organization uses both CMMI-DEV and CMMI-SVC. The Incident Resolution and Prevention (IRP) process area may be used for vulnerability handling.

Refer to the Incident Resolution and Prevention process area of CMMI-SVC V1.3 for more information about effective resolution of service incidents and prevention of service incidents.

Example Work Products
1. Organizational standard processes and process assets for vulnerability handling
2. Vulnerability records and reports
3. Reports on solutions to vulnerabilities

Subpractices
1. Establish and maintain organizational processes and process assets for vulnerability handling.
   Analyze the available processes and activities for handling vulnerabilities.
Examples of process assets for vulnerability handling activities include the following:

- Security contact paths to external parties
- Contact to projects
- Internal communication channels
- Tracking of bug fix activities and problem resolution
- Creation of security advisory
- Security advisories from external sources (e.g. National Vulnerability Database, US-CERT alerts)

2. Create repositories to hold history and records of vulnerabilities, incidents, resolutions, etc. When reported, all details regarding the vulnerability are recorded and stored in the dedicated repository.

3. Establish evaluation criteria to determine vulnerability severity

4. Ensure that the process assets for vulnerability handling are appropriate for use across defined life cycle phases of the product.

   This includes lifecycle phases handled by organizational units other than development (e.g., service organization). They have working interfaces to and from other development process assets (e.g., to product security risk management, configuration management).

5. Establish an infrastructure that vulnerabilities can be reported to (e.g., email address, hotline, website).

6. Fix vulnerabilities by the organization in a timely manner to prevent further damage.

   This may be done by coordinating handling across projects or by creating projects to handle the vulnerability.

7. Manage the vulnerability report to closure.
SECURITY MANAGEMENT IN PROJECTS (SMP)

A Project Management Process Area

**Purpose**

The purpose of Security Management in Projects (SMP) is to establish, identify, plan, and manage security-related activities across the project lifecycle and to manage product security risks.

**Introductory Notes**

This process area builds on practices of the Project Planning (PP), Integrated Project Management (IPM) and Supplier Agreement Management (SAM) process areas in CMMI-DEV by adding security related practices. This includes defining, planning, and integrating security activities, as well as project specific security training. It also addresses security related considerations for managing suppliers. Underlying causes of security vulnerabilities are identified and resolved.

The identification, management, and mitigation of product security risks are planned. This is different from the Risk Management (RSKM) process area. RSKM focuses on risks that can have serious consequences for the project objectives, while the goal in this process area addresses product (not project) operation objectives when relying on the product in a defined environment.

**Related Process Areas**

Refer to the Integrated Project Management process area of CMMI-DEV V1.3 for more information about integrated plans in the project and for causes of selected issues.

Refer to the Project Planning process area of CMMI-DEV V1.3 for more information about the elements for planning a project and for project related training.

Refer to the Supplier Agreement Management process area of CMMI-DEV V1.3 for more information about selecting suppliers and related criteria.

**Specific Goal and Practice Summary**

SG 1 Prepare and Manage Project Activities for Security

SP 1.1 Establish the Integrated Project Plan for Security Projects

SP 1.2 Plan and Deliver Security Training

SP 1.3 Select Secure Supplier and Third Party Components

SP 1.4 Identify Underlying Causes of Vulnerabilities

SG 2 Manage Product Security Risks

SP 2.1 Establish Product Security Risk Management Plan
Specific Practices by Goal

SG 1 Prepare and Manage Project Activities for Security

*Identify, plan, and manage project activities to address security topics.*

This goal addresses practices that are performed in a secure project in addition to practices in non-secure projects.

In a project for secure products, security related activities are identified and planned. Organizational standard processes and other organizational process assets for the development of secure products are identified and tailored according to the tailoring guidelines to address the needs of the projects. Plans for the project addressing security activities are integrated with other plans. Each project may have needs for specific security related training that exceeds training needs for non-secure projects. The selection of suppliers and third party components receives a particular security dimension. In case of reported or detected vulnerabilities, the underlying causes are identified and analyzed.

*Refer to the Perform Security Validation specific goal of the Security Verification and Validation process area for more information about proactive identification of vulnerabilities.*

SP 1.1 Establish the Integrated Project Plan for Security Projects

*Establish and maintain plans for developing a secure product and integrate them with other project plans.*

The project’s defined secure development process is tailored from organizational process assets. The tailoring is performed according to guidelines and criteria that are particularly suitable for the development of secure products. The use of lessons learned and experience obtained by other projects in the organization is useful for defining a project’s process that integrates the activities needed for secure products. The result is a defined process with different, additional (for product security risks, see also subpractice 3) or modified activities, lifecycles and procedures compared to the development of non-secure products.

The processes for secure development are planned. The resulting plans are integrated in the project’s plan.

*Refer to the Organizational Preparedness for Secure Development process area for more information about creating process assets and organizational processes for secure development.*
**Example Work Products**

1. The project’s defined development process that is appropriate to develop secure products
2. Integrated project plan with security activities and plans
3. Project’s work environment that is appropriate for developing secure products

**Subpractices**

1. Determine from the organization’s set of standard processes and other process assets those that best fit the needs of the project for secure development.

2. Determine the lifecycle model that is the best fit for the development of the secure product by using selection criteria.

   If using agile lifecycles, security needs to be included (e.g., in the product backlogs and sprints).

3. Tailor the standard processes and other organizational process assets according to the tailoring guidelines to establish and document the project’s defined secure development process.

   The project’s defined process covers all of the security activities for the project.

   Examples of project activities for secure development include the following:
   - Security requirements development
   - Supplier management under consideration of security
   - Threat modeling
   - Security product risk management
   - Secure configuration according to standards
   - Security verification and validation

4. Use the organizational work environment standards for secure products and establish a work environment for the project.

5. Use other process assets for secure development from the organization’s process asset library as appropriate (e.g., lessons learned, use of the organizational measurement repository).

   This could also include security standards from various sources (e.g., internal, laws, regulatory bodies, enterprise security policy, and secure technical implementation guidance to include enterprise and national or international standards that apply to the technology involved in the product) as appropriate for the product to be developed.

6. Integrate plans for secure development with other project plans as needed.
8. Conduct peer reviews of the project's defined process for secure development.

   Refer to the Verification process area of CMMI-DEV V1.3 for more information about conducting peer reviews.

9. Revise the project's defined process for secure development as necessary.

   SP 1.2 Plan and Deliver Security Training

   Plan for security knowledge and skills needed to perform the project and deliver needed training.

   Refer to the Organizational Training process area of CMMI-DEV V1.3 for more information about the relation between strategic training approach and project training.

   Refer to the Establish Awareness, Knowledge, and Skills for Product Security specific practice of the Organizational Preparedness for Secure Development process area for more information about security trainings on an organizational level.

   Trainings regarding security awareness and security knowledge needed on an organizational level (e.g., for many projects and stakeholders) are developed and delivered as described in the Organizational Preparedness for Secure Development (OPSD) process area. Training must also address ramping staff up so that the necessary skills needed to build security into the product are addressed.

   This practice addresses the security knowledge and skills required specifically for a project. The knowledge needed may vary from project to project depending on technologies, application domain, risks, and threats. Therefore, each project identifies its needs. Identifying the project's needs is essential for the success of the project. The project's training needs are supplemented by the organizational training needs for security as described in the OPSD process area.

   The knowledge needs for the projects are addressed by relevant training or assigning trained resources etc. in a timely manner.

   Example Work Products
   1. Inventory of security related skill needs for the project
   2. Project training plan including security topics

   Subpractices
   1. Identify the security knowledge and skills needed for the project.
Examples of security skills for the project include the following:

- Knowledge of the threat landscape in the intended operational environment of the product
- Knowledge of product security risk assessment
- Knowledge of security risk elicitation techniques like abuse cases
- Technology-specific security knowledge, e.g., for web service security frameworks
- Knowledge of security architecture review methodologies
- Knowledge of secure coding practices
- Knowledge of source code analysis against industry and organization standards
- Cryptography knowledge
- Penetration testing skills
- Knowledge about specific security tooling, e.g., static code analyzer

2. Assess the security knowledge and skills that are available.
3. Select methods for providing needed knowledge and skills.
4. Incorporate selected methods and identified trainings and certifications in the project plan.
5. Deliver security training as planned
6. Track the delivery of security training against the plan
Select Secure Supplier and Third Party Components

Security is addressed when selecting suppliers and third party components for secure products.

If components from suppliers are integrated in the secure product or the product depends on services of suppliers (e.g. when hosted by a cloud provider), the selection of suppliers and components also involves criteria taking the security situation of the product to be developed into account. This practice applies to components to be integrated in the product that are manufactured or developed by suppliers, as well as for third party components (e.g., “as is” COTS, modified COTS or open source products), since those components may bear security risks as well. Use of a proven procurement language helps to define the security requirements for suppliers.

The organization ensures that the suppliers’ security standards meet the acquirer’s security expectations. Audits are coordinated with the supplier and conducted to ensure that those expectations are met. The audits may cover compliance to the acquirer’s security regulations as well as general process capability, use of security related standards, laws, process models, and best practices.

Example Work Products

1. Supplier agreements including documented security criteria
2. Supplier selection sheets with security criteria
3. Supplier management plan
4. Supplier contracts including security requirements
5. Supplier security audit reports
6. Action plans for mitigation of security audit findings

Subpractices

1. Identify the risks and threats that may be introduced by including supplier products or third party components into the product.
2. Establish and maintain security criteria for selecting suppliers and third party components.

Examples of security criteria for supplier selection include the following:

- Proven implementations of security related process models and standards such as Security by Design, ISO 27001
- Secure software development practices
- Security trainings and personnel certifications
- Information security management system (e.g., information security policies, data protection)
- Background checks
Examples of security criteria for selection of third party components include the following:

- Security certifications such as the Common Criteria
- Frequency of publications and other information about identified security issues in the component
- Availability of security updates, reaction time to the publication of security issues
- Results of security code analysis or security tests
- Defined and implemented product quality attributes supporting security
- Availability of security contacts, e.g., hotline, vulnerability handling
- Maintenance period, activity of the development community (for open source software)
- Provided security features
- Provided security documentation

3. Document the security criteria, e.g., in a supplier management plan.

   The role of the acquiring organization and the supplier security-related activities, as well as criteria, need to be specified in the supplier management plan and the supplier agreement.

4. Evaluate and select suppliers and third party components according to established security criteria.

   The established security criteria are applied in procurement activities to ensure the selection of appropriate suppliers and third party components and their adherence to the criteria.

   *Refer to the Decision Analysis and Resolution (DAR) process area of CMMI-DEV V1.3 for performing high-impact decisions.*

5. Plan and conduct supplier audits to ensure that appropriate security methods, standards, regulations, etc. are in place.

   Audits need to be agreed in the supplier agreement. The audit should be conducted in a timely manner (e.g. as part of the solicitation process) so that the results can be used to support the supplier selection for the particular project. The audit may be performed before the supplier selection or as part of the selection process. Additionally, regular audits should be conducted to monitor the supplier as appropriate.

6. Develop action plans to mitigate audit findings together with the supplier.

   Examples for mitigations are a security roadmap for the supplier or a decision to exclude a specific supplier or third party component.

**SP 1.4 Identify Underlying Causes of Vulnerabilities**

*Analyze and resolve underlying causes of security vulnerabilities.*
Security vulnerabilities are security relevant defects of a product or product component. Security vulnerabilities are discovered either in security validation activities such as a product security risk assessment or are reported to the organization by third parties (e.g., customers, security researchers).

An underlying cause of a vulnerability is a condition or event that contributes to the occurrence of one or more vulnerabilities.

The purpose is to obtain information to prevent recurrence in the same and other products.

Root cause analysis may be applied as well to identify the root cause in security processes that are selected for quantitative management and to improve process performance.

Refer to the Causal Analysis and Resolution process area of CMMI-DEV V1.3 for more information about identifying root causes of problems.

Refer to the Perform Security Validation specific goal of the Security Verification and Validation process area for more information about proactive identification of vulnerabilities.

Example Work Products
1. Analysis results from reported vulnerabilities
2. Action proposals and mitigation plans to address vulnerabilities
3. Closure reports of underlying causes

Subpractices
1. Involve the relevant stakeholders (e.g., security expert, architect, product management, user), and form a team as needed.
   The information obtained is made available to all relevant stakeholders in a timely manner to enable them to identify, research, prevent, and address similar issues.
2. Analyze issues and vulnerabilities if similar vulnerabilities exist in other products or product components.
3. Identify underlying causes of vulnerabilities.
4. Identify all affected work products.
5. Develop mitigation plans to close the vulnerability and to remove the underlying cause.
   Keep a strong focus on identifying and removing the underlying cause in a timely manner, as operators of the product might be at risk. Additionally, if the issue was reported by a person external to the organization, a vulnerability report might be published before security updates are published.
6. Manage the removal of underlying causes to closure.
7. Conduct root cause analysis (causal analysis) as needed.
8. Communicate results to relevant stakeholders in a timely manner to prevent further damage for this and other products.

9. Update security requirements to be included in future products.

10. Update and create process assets, the organizational processes, and the projects’ processes as needed to address underlying causes and root causes to prevent future occurrence.

While the project usually addresses the underlying causes of vulnerabilities, identification of underlying causes of vulnerabilities is not limited to the lifecycle of the development project, but may also be performed over defined phases of the product’s life cycle. Therefore it may become also an organizational task.

SG 2 Manage Product Security Risks

The product security risks are managed throughout the project.

The Risk Management process area covers risks that can have serious consequences for the project objectives (e.g., schedule, cost, functionality).

In contrast, this Product Security Risk Management goal addresses security threats to product (not project) operation objectives (e.g., risks to the confidentiality, integrity, or availability of data or product components) when relying on the product in a defined environment. The Product Security Risk Management is usually performed separately from RSKM. The results of the Product Security Risk Management may be an input (or category) for RSKM for the project (then to be handled as risks for the project objectives).

Product security risk assessments are conducted to identify mitigations that may add to already elicited requirements and to security activities defined in the project’s processes. The major elements of product security risk management are establishing a product security risk management plan, product security risk assessment and mitigation plans.

Product security risk management in the development project takes into account the threats the product will be exposed to during commissioning, operation and decommissioning.

For example, a product security risk assessment during development evaluates if physical attacks to the product in operation are of concern.

To be able to analyze operational risks, the intended operational environment is defined.

Refer to the Risk Management process area of CMMI-DEV V1.3 for more information about identifying and managing risks that have consequences for meeting the project objectives.
Establish Product Security Risk Management Plan

Establish and maintain a product security risk management plan.

The product security risk management plan sets the framework for performing the relevant activities. The most appropriate method (or technique) is determined. The method may also change over the project lifecycle. Together with the method (or integrated in the method), the security risk parameters to be used are defined.

Example Work Products
1. Product security risk management plan

Subpractices
1. Identify methods and tools to be used for product security risk assessment and mitigation

The most appropriate method (or technique) depends on the lifecycle phase of the project.

During early phases, e.g., requirements development, workshop based risk assessment helps to identify adequate security requirements. Methodologies like the Security Quality Requirements Engineering (SQUARE) Methodology [SEI 2005 b], which focuses on using product security risk assessment for eliciting requirements, can be applied here.

During development of architecture and design a more sophisticated approach is necessary to identify product specific security risks, e.g., Threat Modeling [SWI 2004].

The methods and tools are usually available in the process asset library and are determined by the project by tailoring. If not available in-house, the project selects external methods and tools or creates appropriate methods.

2. Define security risk parameters.

When the methods and tools are available in the process asset library, the security risk parameters are usually already defined. Projects that create methods also need to create the risk parameters.

The risk parameters define how security threats are evaluated regarding likelihood and impact levels; e.g., the likelihood of a threat is evaluated as very likely if no countermeasures are in place and the threat can be exploited via a public interface.

The product security risk management also defines which combination of likelihood and impact levels results in what risk levels and how the risks have to be treated in regard to defined levels, e.g., by development of a mitigation plan, required sign off for risk acceptance.


Security risks analyses are conducted in appropriate phases over the project lifecycle to cover security risks on varying levels of detail, e.g. during requirements development or design definition. Security risk
assessments should also be scheduled within the product lifecycle, e.g., as the product is already in operation.


Security risks to a product change as the intended operational environment changes, also with time. Therefore, the security risk assessments are updated regularly to cover, e.g., changes of the intended operational environment or newly discovered threats.

New risks might also be identified during security validation or vulnerability handling. In this case, the product security risks list is also updated.

5. Revise the product security risk management plan as necessary.

Refer to the Perform Security Validation specific goal of the Security Verification and Validation process area for more information about security risks identified during security validation.

---

**SP 2.2 Perform Product Security Risk Assessments**

**Identify, analyze, and evaluate product security risks.**

Security threats to a product vary depending on the operational environment (e.g., public use, internal use, interfaces). The involvement of the relevant stakeholders play an essential role in identifying risks, since their vision and experience give valuable input.

**Example Work Products**

1. Product security risk list

**Subpractices**

1. Involve relevant stakeholders (e.g., customer, architect, product management, system test, user, sales, security expert).

2. Identify the intended operational environment.

   Ensure that the definition of the intended operational environment is established and is still valid and current for the product. It might be necessary to evaluate risks for several intended operational environments. Updates for the operational requirement might be needed. Also specify which operational environment is out of scope.

   Refer to the Develop Customer Security Requirements specific practice of the Security Requirements and Technical Solution process area for more information about the operational environment.

3. Define scope of product components.

   The scope defines the product components for the analysis as well as the level of detail of the analysis. All components are covered at least in...
a high level analysis, including third party components and suppliers’ products.

4. Identify, analyze, and evaluate product security risks.

Security threats and risks are identified, analyzed, and evaluated by applying the defined security risks analysis method. Protection goals are derived from business goals or the results of a business impact analysis. An example for a protection goal is the confidentiality of personally identifiable information. A predefined list of threats can be used to support the identification of threats to the product. This list can be created from public sources and threats already identified in a project. Another source is the repository populated by the results of the vulnerability handling process. The identified security risks are documented in a product security risk list.

Examples of security threats include the following:

- Outsider is able to perform denial of service attack on system.
- Malicious user injects JavaScript code to hijack session of legitimate user.
- Internal user is able to escalate his privileges in the system due to architectural security flaws.
- Availability of system is affected due to a virus outbreak.
- Supplier may provide software configuration items that are malicious or of poor quality and insecure.

SP 2.3 Plan Risk Mitigation for Product Security

_Establish, maintain, and implement a plan for the mitigation of product security risks._

The mitigation of risks is planned in accordance with the product security risk management plan. In particular this means that for each identified security risk the handling is defined, planned, implemented, tracked, and documented in the product security risk mitigation plan.
Examples of handling security risks include the following (also depending on defined risk levels and risk strategy):

- Risk reduction: Risk controls are defined in the security risk mitigation plan to reduce the risk to the acceptable risk level defined in the product security risk management plan.
- Risk acceptance: It is not always possible to reduce security risks to the defined acceptable level. Procedures are defined for the acceptance of risks, e.g., management sign-off.
- Risk avoidance: Security risks can be avoided by changing the condition that leads to the security risks. This condition can be, for example, the business model, the specification of a product component (e.g., the integration of a specific technology like a web server) or the intended operational environment (e.g., physical access to the operational product).
- Risk transfer: Security risks can be shared with other parties. Examples are insurance companies or product operators, e.g., by transferring the risks of operating the product in an operational environment other than the documented intended operational environment.

Example Work Products

1. Product security risk mitigation plan

Subpractices

1. Based on the identified risks, plan mitigation and handling of the product security risks.

   In many instances, a risk is mitigated by defining product security requirements, since many risks arise from technical properties of the product (e.g., architecture). Therefore product security risk mitigations are an input for security requirements.

2. Use thresholds and risk levels as defined in the product security risk mitigation plan.

3. Implement the product security risk mitigation plan.

4. Align product security risks with risks that might have a consequence for the project objectives.

   Adding of security requirements (as a result of the product security risk mitigation) and other mitigations may provide also a risk for the project objectives, e.g., cost and schedule.

5. Monitor the product security risks regularly and when triggered by events (e.g., the identification of new vulnerability types, rising attacker attention to the product).

6. Track the implementation of the product security risk mitigation plan and re-evaluate the risks as appropriate.
SECURITY REQUIREMENTS AND TECHNICAL SOLUTION (SRTS)

An Engineering Process Area

Purpose

The purpose of Security Requirements and Technical Solution (SRTS) is to establish security requirements and a secure design and to ensure the implementation of a secure product.

Introductory Notes

This process area builds on practices of the Requirements Development (RD) and the Technical Solution (TS) process areas in CMMI-DEV V1.3. This process area adds security related practices.

For secure products it is essential to have security requirements. This goes beyond the general concept of quality attributes in CMMI. For example, if maintainability requirements are neglected, the product may still be used, but maintenance costs may be higher. If security requirements are neglected, the product may not be usable at all.

The development of security requirements can be combined with regular requirements development activities; ideally they should be integrated.

Based on these customer security requirements, product security and product component security requirements are derived and used for the implementation of the technical solution. This is not explicitly covered in Security by Design, but should be done in an integrated way by using the RD process area in CMMI.

To ensure a secure technical solution, security standards are developed and implemented to be used for a secure architecture, design, and implementation of security and non-security functionality. These development standards summarize proven security principles (e.g., least privilege, definition of trust boundaries) that have to be addressed for architecture and design. Adherence checks to the security standards are supported (e.g. by tools) and security issues are identified and handled in accordance with the product security risk management plan.

For technology selection, security criteria are integrated in the selection criteria. Security threats introduced by technologies selected for the product are evaluated in accordance with the product security risk management plan.
Standards for secure product configuration are developed for all product components (including third party components and all other supplies) to support a secure operation of the product. These standards are applied for product implementation, integration or installation.

For the implementation of the secure design, security standards for secure implementation (e.g., programming language specific coding rules such as forbidden application programming interfaces [APIs]) are established and implemented. Checking of the use is supported by manual or automated methods.

Security topics are distinctively added to the product support documentation to aid a secure integration, installation, configuration and operation of the product.

**Related Process Areas**

Refer to the Requirements Development process area of CMMI-DEV V1.3 for more information about defining and refining requirements.

Refer to the Security Management in Projects process area for more information about product security risk management.

Refer to the Technical Solution process area of CMMI-DEV V1.3 for more information about architecture, design, implementation, and product documentation.

**Specific Goal and Practice Summary**

SG 1 Develop Customer Security Requirements and Secure Architecture and Design

| SP 1.1 | Develop Customer Security Requirements |
| SP 1.2 | Design the Product According to Secure Architecture and Security Design Principles |
| SP 1.3 | Select Appropriate Technologies Using Security Criteria |
| SP 1.4 | Establish Standards for Secure Product Configuration |

SG 2 Implement the Secure Design

| SP 2.1 | Use Security Standards for Implementation |
| SP 2.2 | Add Security to the Product Support Documentation |

**Specific Practices by Goal**

**SG 1 Develop Customer Security Requirements and Secure Architecture and Design**

*Develop customer security requirements and design the product according to secure architecture and designs.*

For the development of security requirements, the intended operational environment is specified. This is also used for conducting the product security risk assessment. Moreover, the security needs of relevant stakeholders (e.g., customer, market, government) and the results of the product security risk
assessments are identified and specified as customer security requirements.

Security principles as part of security standards are identified and applied during the development of architecture and design to prevent security issues. These security principles describe activities (e.g., threat modeling) and constraints (e.g., definition of trust boundaries, server side authorization) to aid the development of a secure implementation. Criteria are established to ensure that security topics are considered during the selection of technologies.

### SP 1.1 Develop Customer Security Requirements

**Develop customer security requirements from stakeholder needs.**

The intended operational environment of the product or product components is the basis for conducting further security activities (e.g., conducting product security risk assessment). Customer security requirements are derived from the security needs of relevant internal (e.g., company, product management) and external (e.g., government, customers, competitors) stakeholders, as well as from the results of the product security risk assessment.

Refer to the Develop Product Requirements specific goal of the Requirements Development process area of CMMI-DEV V1.3 for more information about refining and elaborating customer requirements to develop product and product component requirements.

**Example Work Products**

1. Documentation of intended operational environment
2. Customer security requirements

**Subpractices**

1. Specify intended operational environment.

   Assumptions, constraints, and requirements about the intended operational environment are documented. The operational environment specification is the basis for further security considerations (e.g., during product security risk assessment, security documentation or security validation). The specification is not limited to security aspects but includes the whole operational environment. A considerable amount of the data required for an operational environment specification may already be covered by the definition of non-security related customer requirements.

   The specification of the intended operational environment includes items such as the following:

   - What is the criticality of the typical use cases (e.g., sensitivity of operational data, criticality of functions, dependence of other systems)?
2. Identify and document the security needs.

Security needs from various stakeholders are identified. Sources for security needs can either be internal or external of the organization. The needs shall not only focus on product security features but also include the support of a secure product lifecycle. The analysis of the intended operational environment helps to identify sources for security needs. The analysis of security needs is documented.

Examples of sources of security needs include the following:

- Country specific regulations and laws (e.g., personal data protection, export controls)
- Domain specific regulations and laws (e.g., Cyber Security Standards of the North American Electric Reliability Corporation [NERC 2011 a] and [NERC 2011 b], Health Insurance Portability and Accountability Act (HIPAA) [US 1996], Health Information Technology for Economic and Clinical Health Act (HITECH) [US 2009])
- Request for proposal (RFP) or feature requests
- Product security risk assessment
- Development of misuse and abuse cases
- Organization internal protection goals (e.g., intellectual property protection, licensing regulations)
- Security activities of competitors
- Analysis of existing or future threats by using vulnerability databases (e.g., Common Weakness Enumeration List, SANS TOP 25 Most Dangerous Software Errors)
- Security topics with high customer attention
- Analysis of security incidents in own or comparable systems
- Organizational security baselines and policies

3. Transform the security needs into customer security requirements.

The inputs from the various stakeholders are consolidated and conflicts are resolved resulting in a documented set of customer security requirements.
requirements. The results of the product security risk assessment are also included. The documentation of customer security requirements does not replace but extend the “regular” documentation of customer requirements.


SP 1.2 Design the Product According to Secure Architecture and Security Design Principles

**Develop a secure architecture and design for the product and product components according to security design principles.**

Security design principles for a secure architecture and design are identified and applied.

The security design principles apply to the product and to all product components, not only to components with identified security functionality. A security design issue in any part of the product or one of the product components might lead to security risks.

Example Work Products
1. Secure architecture standards
2. Security standards with security design principles
3. Security architecture documents
4. Security design documents

**Subpractices**
1. Establish and maintain secure architecture and design standards including security design principles.

   Identify, develop, or obtain the security design principles appropriate for architecture and design of the product.

   Standards may be used from the organizational process asset library. They may also be provided by external stakeholders or created by the project as needed.

   Security principles for architecture and design describe activities and constraints to aid in the development of a secure implementation. These principles either affect the complete design or focus on specific topics such as authentication, session management, encryption, input validation, or logging.

   The design standards are updated regularly to ensure the state of the art.
Examples of security design principles for architecture and design include the following:

- Least privilege principle
- Defense-in-depth principle
- Definition of trust boundaries
- Server side authorization
- Need-to-know principle
- Updatability

2. Adhere to secure architecture and design standards.

3. Identify security issues in architecture and design and plan mitigation.

   For development of architecture and design, methods to mitigate security issues and related risks are identified, evaluated, and applied. The evaluation and mitigation is performed in line with the product security risk management plan. This applies to the complete architecture and design, not only security parts.

Examples of methods to identify and mitigate security issues include the following:

- Security threat and risk assessment
- Threat modeling
- Attack surface reduction

Refer to the Manage Product Security Risks specific goal of the Security Management in Projects process area for more information about risk acceptance and risk mitigation techniques.

Refer to the Perform Security Validation specific goal of the Security Verification and Validation process area for more information about security validation.

4. Revise architecture and design as necessary.

SP 1.3 Select Appropriate Technologies Using Security Criteria

Select technologies using established and maintained security criteria.

Refer to the Project Planning process area of CMMI-DEV V1.3 for more information about project resources.

Refer to the Decision Analysis and Resolution process area of CMMI-DEV V1.3 for more information about formal decision making.

Technologies like protocols, platforms, or programming languages have an effect on product security. Criteria for the selection of technologies covering security aspects are established and applied during technology selection. Security threats introduced by selected technologies are evaluated in line with the product security risk management plan.
Example Work Products
1. Technology selection security evaluation criteria
2. Documentation of technology decisions

Subpractices
1. Establish and maintain security evaluation criteria for technology.

Examples for technologies and related security criteria:
- Programming language: built in security mechanism, needed effort to create secure code
- Protocols: encryption, integrity checks, replay protection, known weaknesses
- Cryptographic algorithms: supported key length, recommendation of institutions

2. Evaluate technology specific security threats.
   Various technologies introduce varying security threats. For example, many threats are specific for web technologies. These threats are evaluated for each alternative and used in the decision process.

3. Select technologies according to established criteria and identified threats and document decisions.
   Decisions are based on established selection criteria and the evaluation of the identified security threats in line with the product security risk management plan.

SP 1.4 Establish Standards for Secure Product Configuration

*Establish and maintain standards for secure product configuration and parameterization.*

While the Configuration Management (CM) process area in CMMI addresses the integrity of work products and baselines, this practice addresses the configuration of the product components during operation.

The secure default configurations of all product components are defined in a configuration standard. This includes products from suppliers and third party components as well as components the product relies on, e.g., the operating system in the case of software components. The benefit of the configuration standard is to enable the secure configuration of the product or product components during integration, installation, and operation. The product and product components are developed and tested to run on components that are configured according to the secure configuration standards.

Example Work Products
1. Secure product configuration standards
Subpractices

1. Obtain secure configuration guidelines for products from suppliers and third party components.

   For each product from a supplier and third party component, obtain guidelines from the vendor or other sources that describe configuration options and their security impact. This includes integrated components as well as components the product relies on, for example, the operating system in the case of software components. Standards may be used from the organizational process asset library. They may also be provided by external stakeholders or created by the project as needed.

2. Define and maintain secure configuration standards for product components.

   For all product components, specify a standard configuration that is secure per default.

   The standard configuration is updated regularly to ensure the state of the art.

   Examples of configuration options with an impact on security include the following:
   - Unneeded or rarely needed services and network ports should be deactivated per default to reduce the attack surface.
   - Unneeded default users accounts should be deleted to avoid their misuse.
   - Protocols should be configured in a secure manner. There might be a tradeoff between security and compatibility.

3. Review applicable secure configuration standards.

4. Apply secure configuration standards.

   Configuration of the components can be supported by tools.

SG 2 Implement the Secure Design

Secure products and product components, and associated security support documentation, are implemented.

Security implementation standards are established to ensure a secure use of technologies. These standards define technology specific aspects (e.g., usage of specific APIs, compiler options) and include items required by customers and norms. Security checks are supported to ensure that the standards are implemented. Security support documentation is created to aid in the secure installation, administration, and operation of the product.

SP 2.1 Use Security Standards for Implementation

Implement product components using security standards.

Example Work Products

1. Products developed using security standards
2. Peer review results containing security related criteria
3. Unit test results containing security related test aspects

Subpractices

1. Establish and maintain standards for secure implementation.

Standards for secure implementation define technology that is specific to security guidelines. Secure coding guidelines are a typical example. The standards include guidelines required by customers and norms. Sources for these standards can be identified during the elicitation of customer security requirements. Standards may be used from the organizational process asset library. They may also be provided by external stakeholders or created by the project, as needed.

The standards are updated regularly to ensure state-of-the-art practices.

Examples of secure implementation guideline topics include the following:

- Programming language specific contents such as forbidden APIs or secure memory management
- Technology specific contents such as how to avoid web technology specific security issues
- Compiler configuration and acceptable compiler warnings
- Application of additional tools to avoid exploitation of buffer overflows

2. Support the adherence to standards for secure implementation.

To support adherence, supportive tools can be used in the development environment, e.g. check tools or code analyzers providing immediate feedback during implementation.

3. Conduct security peer reviews of selected work products.

Plan and execute independent expert reviews of critical software configuration items based on analysis of business value of a component.

Refer to the Perform Security Verification specific goal in the Security Verification and Validation process area for more information about verification methods and procedures.

4. Perform security unit testing of the component in accordance with the project plans.

Examples of security test cases for unit testing include the following:

- Large user input is handled securely and does not lead to buffer overflows
- Resources cannot be accessed without proper authorization

Refer to the Perform Security Verification specific goal in the Security Verification and Validation process area for more information about verification methods and procedures and about verifying work products against their specified requirements.

5. Revise the product components in accordance with the project plans.
SP 2.2 Add Security to the Product Support Documentation

Develop and maintain documentation that supports a secure installation, administration and operation of the product.

To support a secure installation, administration, and operation of the product, the product support documentation provides security relevant information. This does not replace but extends development and maintenance of non-security related product support documentation.

Refer to the Develop Product Support Documentation specific practice in the Technical Solution process area of CMMI-DEV V1.3 for more information about developing and maintaining product support documentation.

Examples of topics in security product support documentation include the following:

- Explanation of security features and mechanisms (e.g., user administration, password policies, user roles and rights management, used ports)
- Secure configuration (e.g., deactivation of unused services, removal of default accounts, changing procedures for default passwords)
- Errors and warnings (e.g., security event log)
- Disaster recovery procedures
- Intended operational environment
- Remote connections for maintenance and how to securely handle them
- Procedures to report product defects

Example Work Products

SECURITY VERIFICATION AND VALIDATION (SVV)

An Engineering Process Area

Purpose

The purpose of Security Verification and Validation (SVV) is to ensure that selected work products meet their specified security requirements and to demonstrate that the product or product component fulfills the security expectations when placed in its intended operational environment.

Introductory Notes

This process area builds on practices of the Verification (VER) and the Validation process area (VAL) in CMMI-DEV V1.3. This process area adds security related practices.

Security verification and validation activities are conducted to identify security issues of the product or product components and to test the resistance against malicious attacks. In addition to verify that work products fulfill the defined security requirements, it is essential to use the attacker’s perspective as a source for the validation procedures to ensure the security of the product or product component when placed in its intended operational environment.

The results of the security verification are recorded and used as a basis for identifying corrective actions. The results of the security validation are analyzed and appropriate action is taken (e.g., a reevaluation according to the product security risk management plan and definition of mitigation plans). The implementation of the corrective actions is a prerequisite to save considerable cost of rework associated with troubleshooting problems or patch management. Security vulnerabilities in a productive environment (i.e., a product in operation) can also lead to substantial financial loss or liability problems if security incidents occur.

Security verification and validation do not replace but extend “regular” verification and validation activities.

Related Process Areas

Refer to the Security Management in Projects process area for more information about product security risk management.

Refer to the Validation process area of CMMI-DEV V1.3 for more information about validation.
Refer to the Verification process area of CMMI-DEV V1.3 for more information about verification.

**Specific Goal and Practice Summary**

<table>
<thead>
<tr>
<th>SG 1 Perform Security Verification</th>
</tr>
</thead>
<tbody>
<tr>
<td>SP 1.1 Prepare for Security Verification</td>
</tr>
<tr>
<td>SP 1.2 Perform Security Verification</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SG 2 Perform Security Validation</th>
</tr>
</thead>
<tbody>
<tr>
<td>SP 2.1 Prepare for Security Validation</td>
</tr>
<tr>
<td>SP 2.2 Perform Security Validation</td>
</tr>
</tbody>
</table>

**Specific Practices by Goal**

**SG 1 Perform Security Verification**

*Selected work products are verified against their specified security requirements.*

Security verification ensures that the selected work products meet the specified security requirements and security standards (e.g. for architecture, code, configuration). Procedures and criteria for security verification are defined and the verification environment is established. The results of the security verification are used as a basis for the identification of corrective actions.

**SP 1.1 Prepare for Security Verification**

*Select work products for security verification and establish and maintain environments, procedures, and criteria.*

Refer to the Prepare for Verification specific goal in the Verification process area of CMMI-DEV V1.3 for more information about preparing the work products for verification.

Identify work products for security verification. Define security test cases and prepare the security verification environment.

**Example Work Products**

1. Security verification procedures
2. Security verification criteria
3. List of security testing techniques and tools
4. Security verification environment
5. Procedures for independent review of software code by experts using static and dynamic analysis

**Subpractices**

1. Identify work products for security verification.
Selection of work products from all phases of the development lifecycle ensures an early and broad detection of security defects. The result of a product security risk assessment as described in the Security Management in Projects process area (SMP) can be used to select work products based on the respective risks. Security requirements affect security as well as non-security functionality of the components. Therefore, security verification also covers security functionality as well as non-security functionality to ensure correct implementation of security requirements for the whole product.

Examples of types of work products identified for security verification include the following:
- Architecture and design specification
- Software components
- Security product support documentation
- Installation and deployment documentation

2. Identify and define the verification methods to be used for each selected work product.

Security requirements and standards conformance is verified manually and using tools, for example, by performing unit testing, security peer reviews, tool based security code reviews, and static code analysis.

Refer to the Perform Peer Reviews specific goal in the Verification process area of CMMI-DEV V1.3 for more information about performing peer reviews.

3. Define procedures and criteria for security verification.

Verification procedures (e.g., test descriptions, cases) and criteria are defined to ensure that the work products meet their security requirements. This includes expected results, any tolerances allowed in observations, and other criteria (e.g., test end criteria) for satisfying the security requirements. Equipment and environmental components that are required to support security verification are identified.

Negative test cases have an extraordinary role in security testing. They help to identify security issues; e.g. for the requirement that “all non public interfaces must require authentication and authorization before granting access,” also test if all interfaces that are accessible without authentication are really meant to be public.

Examples of sources for security verification criteria include the following:
- Product and product component security requirements
- Organizational security policies
- Secure architecture and design standards
- Secure configuration standards
- Secure implementation standards
- Parameters for tradeoff between security and cost of testing
- Proposals and agreements

4. Establish and maintain environment for security verification.
The verification environment reflects the intended operational environment.

If operational data is used during verification, sensitive information (e.g., personal identifiable information, user specific passwords) is either removed, "pseudonymized," or "anonymized." This is not limited to security verification but includes also other verification activities.

As a side effect, security verification might adversely affect components or systems not in scope for verification. Therefore an adequate separation of the test environment for security verification is ensured.

Changes to work products for security verification purposes are tracked and placed under an appropriate level of configuration management.

*Refer to the Configuration Management process area of CMMI-DEV V1.3 for more information about configuration management.*

**SP 1.2 Perform Security Verification**

*Selected work products are verified against their specified security requirements and results are analyzed.*

Verifying products and work products incrementally promotes early detection of security issues. The results are analyzed and treatment is planned based on the security risk caused by the identified defects.

**Example Work Products**

1. Verification results
2. Verification reports
3. Analysis report
4. Defect reports
5. Change requests
6. Mitigation plan

**Subpractices**

1. Perform security verification of selected work products against their security requirements.
2. Record the results of security verification.
3. Analyze the results of the security verification activities and security risks caused.
4. Initiate corrective actions.

*Refer to the corrective action practices of the Project Monitoring and Control process area of CMMI-DEV V1.3 for information about implementing corrective action.*
SG 2 Perform Security Validation

The products or product components are validated to ensure that they fulfill the security expectations when placed in their intended operational environment.

In an operational environment, the product or product component is resistant against attacks when exposed to potential security threats. During security validation, the goal is to identify security vulnerabilities in the product or product component without explicitly testing against the defined security requirements. Procedures and criteria for security validation are defined and the validation environment is established. The results of the security validation are analyzed and evaluated in-line with the product security risk management plan to decide if corrective action is required.

SP 2.1 Prepare for Security Validation

Select products and product components for security validation and establish and maintain environments, procedures, and criteria.

Refer to the Prepare for Validation specific goal in the Validation process area of CMMI-DEV V1.3 for more information about preparing the products or product components for validation.

Products and product components for security validation are identified using criteria. Security procedures and criteria are defined and the validation environment is planned and prepared.

Example Work Products
1. Security validation procedures
2. Security validation criteria
3. List of security testing techniques and tools
4. Security validation environment

Subpractices
1. Identify products or product components for security validation.

The result of a product security risk assessment as described in the Security Management in Projects process area (SMP) can be used to focus on the highest risks. Security defects appear in security as well as non-security functionality of the components. Therefore, security validation covers security functionality as well as non-security functionality. Interfaces such as network services or file inputs are most likely to be attacked and are therefore covered by the validation.
Examples of types of products or product components identified for security validation include the following:

- Architecture and design
- Communication module
- Web application
- Database
- Encryption mechanisms
- Authentication and authorization mechanisms
- Session Handling
- Administrative interfaces
- Security product support documentation

2. Define procedures and criteria for security validation.

Security validation procedures and criteria are established to ensure that the product or product component fulfills the security expectations when placed in its intended operational environment and is not vulnerable to relevant commonly known attacks.

Refer to the Develop Customer Security Requirements specific practice from the Security Requirements and Technical Solution process area for more information about the specified intended operational environment.

Examples of sources for security validation criteria include the following:

- Product and product component security requirements
- Customer acceptance criteria
- Product security risk assessment
- Vulnerability and threat databases (e.g., Common Weakness Enumeration List)
- Standards for secure coding and enterprise security technical implementation guidance

Examples of types of test cases for security validation include the following:

- Check if security documentation can be implemented.
- Check if authentication mechanisms can be circumvented by brute force attacks.
- Check if users (authorized and unauthorized) can perform functions not intended.
- Check if denial-of-service attacks are successful.
- Check if code injection can be performed successfully (e.g., java script in web applications).
- Check if SQL statements can be manipulated (SQL injection)
- Check the current patch level of third party components.
- Check if product withstands malicious attack of a sophisticated attacker.

3. Identify the security validation techniques.
Security validation techniques are identified depending on the test case. In general, they can be classified in manual, automated, or a combination of the both.

Examples of types of security validation techniques include the following:

- Architectural risk assessment
- Threat and risk assessment
- Threat modeling
- Security code analysis both static and dynamic
- Automated vulnerability scanning
- Penetration testing
- Friendly hacking
- Fuzzy testing
- Re-play testing

Refer to the Perform Security Threat and Risk assessments specific practice of the Security Management in Projects process area for more information about performing a threat and risk assessment.

4. Identify and acquire security validation tools and equipment.

For the security validation, supportive tools and equipment are identified. Not all security validation tests can be executed automatically by tools only. In some cases, a manual execution with the optional usage of supportive tools is required or new testing tools suitable for the particular purpose might have to be developed.

Examples of types of security testing tools include the following:

- Data or network fuzzers
- Blackbox scanners
- Port scanners
- Vulnerability scanners
- Static software code analyzers
- Dynamic application security testing tools
- Network scanners
- Protocol analyzers

5. Prepare environment for security validation.

The validation environment reflects the intended operational environment.

If operational data is used during validation, sensitive information (e.g., personal identifiable information, user specific passwords) is either removed, pseudonymized or anonymized. This is not limited to security validation but includes other validation activities.

As a side effect, security validation might adversely affect components or systems not in scope for validation. Therefore an adequate separation of the test environment for security validation is ensured.
The environment is configured in a way to provide information that supports efficient testing, for example, configuration of detailed error messages. Note that this should be deactivated in the operational environment.

Changes to products or product components for validation purposes are placed under an appropriate level of configuration management.

Refer to the Configuration Management process area of CMMI-DEV V1.3 for more information about configuration management.

SP 2.2 Perform Security Validation

**Perform security validation on the selected products and product components.**

For security validation, it is beneficial to assure the independence of the security validation team, for example, by forming an independent internal security validation team or by hiring external security consulting companies.

Validation is also conducted on ways the system was not intended to be used, for example, manually creating malicious network packets instead of using the intended client. The validation results are analyzed and treatment is planned based on the security risk caused by the identified defects.

**Example Work Products**

1. Security assessment report
2. Security validation report
3. Factory acceptance test for security
4. Security validation issues
5. Security vulnerabilities
6. Security threats

**Subpractices**

1. Perform security validation of products and product components.
2. Record the results of security validation.
3. Analyze the results of the security validation activities and security risks caused.
4. Initiate corrective actions.

Refer to the corrective action practices of the Project Monitoring and Control process area of CMMI-DEV v1.3 for information about implementing corrective action.
This section lists all of the generic goals and generic practices as in the CMMI-DEV V1.3. The generic goals are organized in numerical order, GG 1 through GG 3, reflecting also the respective capability levels. The generic practices are also organized in numerical order under the generic goal they support. The informative material of the generic practices and generic goals is not repeated here, since it is applied as it is in the CMMI-DEV V1.3. For some generic practices, Security by Design has supplemented more informative material that can be used for all security process areas.

*Refer to the Generic Goals and Generic Practices section in Part 2 of the CMMI-DEV V1.3 for more information about each of the generic practices and goals.*

This section gives the generic practice elaborations for the security process areas.

**GG 1  Achieve Specific Goals**

*The specific goals of the process area are supported by the process of transforming identifiable input work products into identifiable output work products.*

**GP 1.1  Perform Specific Practices**

*Perform the specific practices of the process area to develop work products and provide services to achieve the specific goals of the process area.*

**GG 2  Institutionalize a Managed Process**

*The process is institutionalized as a managed process.*

**GP 2.1  Establish an Organizational Policy**

*Establish and maintain an organizational policy for planning and performing the process.*

**OPSD Elaboration**

The policy establishes organizational expectations for capabilities to develop secure products and react to product security incidents.

**SMP Elaboration**

This policy establishes organizational expectations for developing the project security plan, for suppliers and third party component selection, identification of underlying causes, and security risk management.
SRTS Elaboration

This policy establishes organizational expectations for collecting and eliciting stakeholder security needs and analyzing product security threats and risks. It also addresses security during development of design and implementation and expectations regarding the security of technologies applied and used for development.

SVV Elaboration

This policy establishes organizational expectations for selecting products, product components and work products for security verification and validation; establishing and maintaining environments, procedures, and criteria for security verification and validation; performing security verification and validation of selected products, product components, and work products; and analyzing the results.

GP 2.2 Plan the Process

Establish and maintain the plan for performing the process.

OPSD Elaboration

This plan for performing the organizational preparedness for secure development process can be part of (or referenced by) the organization’s process improvement plan.

SMP Elaboration

This plan for the Security Management in Projects integrates the security processes into integrated planning. It also plans for the product security risk management processes.

SRTS Elaboration

This plan for performing the security requirements development process and the secure technical solution process can be part of (or referenced by) the project plan as described in the Project Planning process area.

SVV Elaboration

This plan for performing the security verification and validation process can be part of (or referenced by) the project plan as described in the Project Planning process area.
GP 2.3 Provide Resources

*Provide adequate resources for performing the process, developing the work products, and providing the services of the process.*

For some critical projects and activities it may be required to use staff that has undergone appropriate security screening and other security checks.

**OPSD Elaboration**

A process group typically manages the organizational preparedness for secure development activities. This group typically is staffed by a core of professionals whose primary responsibility is coordinating organizational process improvement.

This group is supported by process owners and people with expertise in various disciplines such as the following:

- Engineering practices for secure development
- Security risk management
- Vulnerability handling
- Security standards and models
- Process know how for the development of secure products

**SMP Elaboration**

Special expertise in Security Management in Projects may include the following:

- Security experts
- Security trainers

Examples of other resources provided include the following tools:

- Threat modeling tools
- Security risk management tools

**SRSTS Elaboration**

Special expertise in the security domain, in operational environments of the products, in security needs, and in security requirements development may be required.
Examples of resources provided include the following tools:

- Security design analyzing tools
- Threat modeling tools
- Secure configuration tools
- Security code analyzers
- Secure code compilers

SVV Elaboration

Certain security verification and validation methods may require special tools, facilities, and training (e.g., security validation may require special security test equipment and people skilled in the use of the equipment).

Examples of resources provided include the following tools:

- Data or network fuzzers
- Blackbox scanners
- Port scanners
- Vulnerability scanners
- Network scanners
- Protocol analyzers

For security validation, it is recommended to ensure the independence of the security validation team. Examples for security validation teams include the following:

- Internal security validation teams
- External security teams specialized in “hacking techniques”

GP 2.4 Assign Responsibility

Assign responsibility and authority for performing the process, developing the work products, and providing the services of the process.

OPSD Elaboration

A process group typically manages the organizational preparedness for secure development activities. This group works with security subject matter experts and with higher level management.

SMP Elaboration

A security group or security responsible oversees the activities of security, in particular, product security risk management.

SRTS Elaboration

A security group or security responsible oversees the activities of security requirements development, secure architecture, design, and implementation activities (in particular the
operational environment and laws, regulations, and standards, as well as the architectural, product configuration, and implementation standards, etc.).

SVV Elaboration

A security group oversees the activities of secure verification and validation, in particular to ensure that the required security is achieved.

GP 2.5  Train People

Train the people performing or supporting the process as needed.

OPSD Elaboration

Examples of training topics include the following:

- Processes, process models and standards for secure development
- Process models for related topics, such as safety, organizational resiliency
- Developing work environment standards for secure development
- Security standards

SMP Elaboration

Examples of training topics include the following:

- Secure software lifecycle
- Security risk assessment method
- Threats

SRTS Elaboration

Examples of training topics include the following:

- Security requirements engineering
- Secure software development
- Common attack techniques
- Security standards
- Secure architecture development
- Secure design development
- Secure coding
- Secure configuration
- System hardening

SVV Elaboration

Examples of training topics include the following:

- Security testing tools
- Penetration testing
- Common attack techniques
### GP 2.6 Control Work Products

*Place selected work products of the process under appropriate levels of control.*

**OPSD Elaboration**

<table>
<thead>
<tr>
<th>Examples of work products placed under control include the following:</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Organization’s set of standard processes for secure development</td>
</tr>
<tr>
<td>- Definitions of the common set of product and process measures particularly for secure development</td>
</tr>
<tr>
<td>- Vulnerability handling process</td>
</tr>
<tr>
<td>- Training and awareness material for secure development</td>
</tr>
<tr>
<td>- Lessons learned from secure development</td>
</tr>
<tr>
<td>- Library with already known attack patterns</td>
</tr>
</tbody>
</table>

**SMP Elaboration**

<table>
<thead>
<tr>
<th>Examples of work products placed under control include the following:</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Project’s defined secure development process</td>
</tr>
<tr>
<td>- Product security risk management plan</td>
</tr>
<tr>
<td>- Product security risk mitigation plan</td>
</tr>
</tbody>
</table>

**SRTS Elaboration**

<table>
<thead>
<tr>
<th>Examples of work products placed under control include the following:</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Customer security requirements</td>
</tr>
<tr>
<td>- Security threat and risk assessment strategy</td>
</tr>
<tr>
<td>- Security threat and risk assessment documentation</td>
</tr>
<tr>
<td>- Secure architecture and design standards</td>
</tr>
<tr>
<td>- Security architecture documents</td>
</tr>
<tr>
<td>- Security design documents</td>
</tr>
<tr>
<td>- Secure technology selection criteria</td>
</tr>
<tr>
<td>- Documentation of secure technology decision</td>
</tr>
<tr>
<td>- Secure configuration standards</td>
</tr>
<tr>
<td>- Secure implementation standards</td>
</tr>
<tr>
<td>- Security documentation</td>
</tr>
</tbody>
</table>

**SVV Elaboration**

<table>
<thead>
<tr>
<th>Examples of work products placed under control include the following:</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Security verification procedures and criteria</td>
</tr>
<tr>
<td>- Security verification reports</td>
</tr>
<tr>
<td>- Security validation procedures and criteria</td>
</tr>
<tr>
<td>- Security validation reports</td>
</tr>
</tbody>
</table>
GP 2.7 Identify and Involve Relevant Stakeholders

Identify and involve the relevant stakeholders of the process as planned.

OPSD Elaboration

Key stakeholders in the OPSD process include the process group, higher level management, and security experts.

Examples of activities for stakeholder involvement include the following:
- Reviewing many organizational plans, assets, and other work products for impact on secure development
- Assessing the definitions of the common set of process and product measures, particularly for secure development
- Reviewing the work environment standards for secure development
- Contributing to lessons learned
- Identifying causes of any kind for security issues

SMP Elaboration

Examples of activities for stakeholder involvement include the following:
- Establishing integrated project security plan
- Establishing criteria for evaluating suppliers and third party components
- Definition of security risk parameters
- Performing product security risk assessment

SRTS Elaboration

Select relevant stakeholders from customers, end users, regulatory bodies, authorities, developers, producers, testers, suppliers, marketers, maintainers, disposal personnel, and others who may be affected by, or may affect, the security of the product as well as the security activities in the process. Note that security requirements may also vary by country, for example, requirements imposed by regulatory bodies. This may apply to the country of product origin and also to the countries where the product is sold.

Examples of activities for stakeholder involvement include the following:
- Eliciting, contributing, and delivering security requirements
- Reviewing the adequacy of security requirements
- Establishing product and product component security requirements
- Discussing security acceptance levels
- Defining secure architecture and design standards
- Identifying security issues in architecture and design
- Selecting appropriate technologies
- Defining secure configuration standards
Security by Design with CMMI for Development, Version 1.3

**SVV Elaboration**

Select relevant stakeholders from customers, end users, regulatory bodies, authorities, developers, producers, testers, suppliers, marketers, maintainers, disposal personnel, and others who may be affected by, or may affect, the security of the product as well as the security activities in the process.

Examples of activities for stakeholder involvement include the following:

- Selecting work products for security verification
- Establishing the security verification procedures and criteria
- Assessing security verification results and identifying corrective action
- Selecting products and product components for security validation
- Establishing the security validation procedures and criteria
- Attending verification and validation activities
- Reviewing results of security validation and resolving security issues
- Resolving security issues with the customers, end users and other stakeholders

**GP 2.8 Monitor and Control the Process**

*Monitor and control the process against the plan for performing the process and take appropriate corrective action.*

**OPSD Elaboration**

Examples of measures and work products used in monitoring and controlling include the following:

- Progress against security-related business objectives
- Percentage of projects using process elements of the organization’s set of standard processes for secure development
- Number of process elements of the organization’s set of standard processes for secure development
- Number of causes for security issues of any kind found in the products
- Timeliness of implementing process changes that affect the prevention of security issues
- Number and impact of security issues caused by unsecure working environment
- Number of people who received security awareness
- Number of people who received security training
- Progress toward achieving migration to a secure development process

**SMP Elaboration**

Examples of measures and work products used in monitoring and controlling include the following:
- Number of people who received security training in the project
- Number of findings in supplier audits
- Number of identified product security risks
- Number of mitigated product security risks

**SRTS Elaboration**

Examples of measures and work products used in monitoring and controlling include the following:

- Cost, schedule, and effort expended for rework caused by defects in security requirements
- Schedule for activities to develop a set of security requirements.
- Rate of changes in security requirements
- Cost, schedule, and effort expended for rework caused by design defects
- Percentage of security requirements addressed in the product or product component design
- Schedule for secure architecture, design and implementation activities

**SVV Elaboration**

Examples of measures and work products used in monitoring and controlling include the following:

- Number of security issues detected
- Security verification problem report status (i.e., how long each problem report has been open)
- Schedule for a specific security verification activity
- Number of security validation activities completed (planned versus actual)
- Security validation problem report aging (i.e., how long each problem report has been open)
- Schedule for a specific security validation activity

**GP 2.9 Objectively Evaluate Adherence**

Objectively evaluate adherence of the process and selected work products against the process description, standards, and procedures, and address noncompliance.

**OPSD Elaboration**

Examples of activities reviewed include the following:

- Establishing security-related business objectives
- Establishing organizational process assets for secure development
- Establishing work environment standards for secure development
- Performance of the lessons learned process
- Implementation of security awareness training
Examples of work products reviewed include the following:

- Organization’s set of standard processes for secure development
- Adherence of standard processes to applicable standards
- Organization’s measurement data for secure development
- Training and awareness material for security

SMP Elaboration

Examples of activities reviewed include the following:

- Tailoring of the standard processes for secure projects
- Supplier and third party component selection including security criteria
- Product security threat and risk assessment
- Identifying and evaluating product security risks
- Actions performed resulting from security risks being realized

Examples of work products reviewed include the following:

- The project’s defined secure development process
- Product security risk management plan
- Product security risk mitigation plan
- Project’s security training plan
- Supplier agreements
- Supplier and third party component selection plan

SRTS Elaboration

Examples of activities reviewed include the following:

- Collecting and eliciting stakeholder security needs
- Formulating product and product component security requirements
- Analyzing and validating product and product component security requirements
- Identify security issues in architecture and design and plan mitigation
- Select technologies according to established criteria and identified threats
- Define and maintain secure configuration standard for product components
- Creation and application of standards for secure implementation
- Develop security support documentation

Examples of work products reviewed include the following:

- Product security requirements
- Product component security requirements
- Standards for secure architecture and design
- Security architecture and design documents
- Technology selection criteria
- Documentation of technology decision
Secure implementation standards
Security support documentation

SVV Elaboration

Examples of activities reviewed include the following:
- Selecting the work products for security verification
- Establishing and maintaining security verification procedures, and criteria
- Performing security verification of work products
- Selecting the products and product components for security validation
- Establishing and maintaining security validation procedures, and criteria
- Performing security validation of products or product components

Examples of work products reviewed include the following:
- Security verification procedures and criteria
- Security verification reports
- Security validation procedures and criteria
- Security validation reports

GP 2.10 Review Status with Higher Level Management

Review the activities, status, and results of the process with higher level management and resolve issues.

OPSD Elaboration

These reviews are typically in the form of a briefing presented to the management by the process group, particularly focusing on the results and progress of process activities for secure development.

Examples of presentation topics include the following:
- Status of improvements for secure development topics being developed by process teams
- Status of progress toward achieving security-related business objectives

SMP Elaboration

Reviews of the product security risk status are held on a periodic and event-driven basis, with appropriate levels of management, to provide visibility into the potential for project risk exposure and appropriate corrective action.

SRTS Elaboration

Reviews are held with appropriate levels of management to provide visibility into the status of, the attention for and quality of the security requirements.

SVV Elaboration

Reviews are held with appropriate levels of management to provide visibility into meeting the security requirements and the
status of the secure product behavior in the intended environment.

<table>
<thead>
<tr>
<th>GG 3</th>
<th>Institutionalize a Defined Process</th>
</tr>
</thead>
<tbody>
<tr>
<td>GP 3.1</td>
<td>Establish a Defined Process</td>
</tr>
<tr>
<td><strong>Establish and maintain the description of a defined process.</strong></td>
<td></td>
</tr>
<tr>
<td>GP 3.2</td>
<td>Collect Process-Related Experiences</td>
</tr>
<tr>
<td><strong>Collect process-related experiences derived from planning and performing the process to support the future use and improvement of the organization’s processes and process assets.</strong></td>
<td></td>
</tr>
</tbody>
</table>

Elaboration for all security PAs

Process-related experiences are obtained from discussions, publications in media (e.g. print, internet), and by applying processes and the implementation and use of tools and methods. Those assets and other security related knowledge (e.g. vulnerabilities, incident resolution approaches, usefulness, effectiveness, and process impact) are obtained in a regular and timely manner and are made available in a timely and proactive manner to all other projects because speed and timeliness are essential due to the nature of the topics. Some process assets (e.g., created by a project) for secure development (e.g., authentication library, central input validation component and code samples) may require approval before used by other projects in order to raise security, efficiency and maintainability. All lessons learned, exemplary work products, results from appraisals and process related experience, etc. are recorded and proactively shared with other projects.

**OPSD Elaboration**

Examples of process-related experiences include the following:

- Submission of lessons learned for secure development to the organization's process asset library
- Submission of measurement data to the organization’s measurement repository (for secure development)
- Requests for additional and improved processes for secure development
- Experiences from performing security awareness and security training

**SMP Elaboration**

Examples of process-related experiences include the following:

- Experiences from performing security trainings
- Discovered threats
• Reusable product security risk mitigations
• Data from the project that is suitable to help other projects, for example, security lessons learned, security-related effort and cost, measurements, etc.

SRTS Elaboration

Examples of process-related experiences include the following:
• Number of security requirements introduced at each phase of the project lifecycle
• Results of applying new security methods and tools
• Developed secure architecture and design standards
• Developed evaluation criteria for technology
• Results of technology security analysis
• Developed secure configuration standards
• Developed standards for secure implementation

SVV Elaboration

Examples of process-related experiences include the following:
• Number of product security issues found through security verification per development phase
• Security verification report
• Number of product security issues found through security validation per development phase
• Security validation report
Appendix A: References


[Fich 2011 a] Fichtinger, Barbara; Russwurm, Winfried; Panholzer, Peter; Process Models and Implementation Examples for the Development of Secure Products presented at SEPG Europe 2011 conference, contribution #1780


[ISA 201x] ISA-99.03.04 / IEC 62443-3-4, Product development requirements, ISA99 Committee, under development, http://isa99.isa.org

[ISCI 2010] Embedded Device Security Assurance, ISASecure Certification Scheme, Version 1.1, EDSA-100, ISA Security Compliance Institute, June 2010


[Pan 2011 a] Panholzer, Peter; Russwurm, Winfried; Fichtinger, Barbara. How to Develop Secure Products Using CMMI and Some “Extras” presented at SEPG North America 2011 conference, contribution #1740


[US 2009] Health Information Technology for Economic and Clinical Health Act (HITECH Act), enacted as part of the American Recovery and Reinvestment Act, 2009

## Appendix B: Acronyms

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADS</td>
<td>Appraisal Disclosure Statement</td>
</tr>
<tr>
<td>API</td>
<td>Application Programming Interface</td>
</tr>
<tr>
<td>CMM</td>
<td>Capability Maturity Model</td>
</tr>
<tr>
<td>CMMI</td>
<td>Capability Maturity Model Integration</td>
</tr>
<tr>
<td>CMMI-DEV</td>
<td>CMMI for Development</td>
</tr>
<tr>
<td>CMMI-SVC</td>
<td>CMMI for Services</td>
</tr>
<tr>
<td>COTS</td>
<td>Commercial Off the Shelf</td>
</tr>
<tr>
<td>CT</td>
<td>Corporate Technology</td>
</tr>
<tr>
<td>DAR</td>
<td>Decision Analysis and Resolution (process area)</td>
</tr>
<tr>
<td>GG</td>
<td>Generic Goal</td>
</tr>
<tr>
<td>GP</td>
<td>Generic Practice</td>
</tr>
<tr>
<td>IEEE</td>
<td>Institute of Electrical and Electronics Engineers</td>
</tr>
<tr>
<td>IRP</td>
<td>Incident Resolution and Prevention (process area in CMMI-SVC)</td>
</tr>
<tr>
<td>ISMS</td>
<td>Information Security Management System</td>
</tr>
<tr>
<td>ISO</td>
<td>International Organization for Standardization</td>
</tr>
<tr>
<td>ISO/IEC</td>
<td>International Organization for Standardization and International Electrotechnical Commission</td>
</tr>
<tr>
<td>ML</td>
<td>Maturity Level</td>
</tr>
<tr>
<td>OPD</td>
<td>Organizational Process Definition (process area)</td>
</tr>
<tr>
<td>OPSD</td>
<td>Organizational Preparedness for Secure Development (process area)</td>
</tr>
<tr>
<td>OT</td>
<td>Organizational Training (process area)</td>
</tr>
<tr>
<td>PA</td>
<td>Process Area</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------</td>
</tr>
<tr>
<td>PARS</td>
<td>Published Appraisal Results</td>
</tr>
<tr>
<td>PI</td>
<td>Product Integration (process area)</td>
</tr>
<tr>
<td>PMC</td>
<td>Project Monitoring and Control (process area)</td>
</tr>
<tr>
<td>PP</td>
<td>Project Planning (process area)</td>
</tr>
<tr>
<td>PPQA</td>
<td>Process and Product Quality Assurance (process area)</td>
</tr>
<tr>
<td>QA</td>
<td>Quality Assurance</td>
</tr>
<tr>
<td>RD</td>
<td>Requirements Development (process area)</td>
</tr>
<tr>
<td>REQM</td>
<td>Requirements Management (process area)</td>
</tr>
<tr>
<td>RSKM</td>
<td>Risk Management (process area)</td>
</tr>
<tr>
<td>SAM</td>
<td>Supplier Agreement Management (process area)</td>
</tr>
<tr>
<td>SAS</td>
<td>SCAMPI Appraisal System</td>
</tr>
<tr>
<td>SCAMPI</td>
<td>Standard CMMI Appraisal Method for Process Improvement</td>
</tr>
<tr>
<td>SD</td>
<td>Service Delivery (process area in CMMI-SVC)</td>
</tr>
<tr>
<td>SEI</td>
<td>Software Engineering Institute</td>
</tr>
<tr>
<td>SG</td>
<td>Specific Goal</td>
</tr>
<tr>
<td>SMP</td>
<td>Security Management in Projects (process area)</td>
</tr>
<tr>
<td>SP</td>
<td>Specific Practice</td>
</tr>
<tr>
<td>SRTS</td>
<td>Security Requirements and Technical Solution (process area)</td>
</tr>
<tr>
<td>SVC</td>
<td>Services (as used for CMMI-SVC)</td>
</tr>
<tr>
<td>SVV</td>
<td>Security Verification and Validation (process area)</td>
</tr>
<tr>
<td>TS</td>
<td>Technical Solution (process area)</td>
</tr>
<tr>
<td>VAL</td>
<td>Validation (process area)</td>
</tr>
<tr>
<td>VER</td>
<td>Verification (process area)</td>
</tr>
</tbody>
</table>
## Appendix C: Glossary

The glossary defines security terms exclusively used in the Security by Design application guide. Terms that are already defined in the glossary of CMMI-DEV V1.3 [SEI2010] or in SCAMPI A MDD V1.3 [SEI 2011 a] are not repeated here. Refer to the glossary sections of above documents for those terms.

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Acceptance criteria</strong></td>
<td>The criteria that a deliverable must satisfy to be accepted by a user, customer, or other authorized entity.</td>
</tr>
<tr>
<td><strong>Anonymization</strong></td>
<td>The act of replacing personally identifiable information with generic identifiers.</td>
</tr>
<tr>
<td><strong>Information security</strong></td>
<td>The protection of information and information systems from unauthorized access, use, disclosure, disruption, modification, or destruction in order to provide confidentiality, integrity, and availability. [IEC 2008b]</td>
</tr>
<tr>
<td><strong>Intended operational environment</strong></td>
<td>The intended operational environment is a description of the future usage environment of the developed product. It includes items such as network integration, users, physical environment, etc.</td>
</tr>
<tr>
<td><strong>Pseudonymization</strong></td>
<td>The act of replacing personally identifiable information with generic identifiers.</td>
</tr>
<tr>
<td><strong>Secure work environment</strong></td>
<td>A secure work environment provides tools necessary for secure development and protects the work products created by secure development (e.g., source code, architecture, and design) from unauthorized access and use (e.g., manipulation, theft).</td>
</tr>
<tr>
<td><strong>Security advisory</strong></td>
<td>Message provided from a development organization to its customers to inform about known security issues in a product or product component usually including information about the severity and recommended reactive actions.</td>
</tr>
<tr>
<td><strong>Security functionality</strong></td>
<td>Functionality of a product or product component with the main purpose of protecting against security threats.</td>
</tr>
<tr>
<td><strong>Security incident</strong></td>
<td>Adverse event in a system or network, or the threat of the occurrence of such an event [ISA 2009].</td>
</tr>
<tr>
<td><strong>Security issues</strong></td>
<td>Any circumstance in a product, product component, process, etc. that violates security.</td>
</tr>
<tr>
<td><strong>Security need</strong></td>
<td>A security need is a certain interest of a relevant stakeholder.</td>
</tr>
<tr>
<td><strong>Security principles</strong></td>
<td>Security principles for architecture and design describe activities and constraints to aid the development of a secure implementation. These principles either affect the whole design or focus on specific topics like authentication, session management, encryption, input validation, or logging.</td>
</tr>
<tr>
<td><strong>Security requirements</strong></td>
<td>Security related functional or non-functional requirements to products or product-components.</td>
</tr>
<tr>
<td>---------------------------</td>
<td>---------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>Security risk</strong></td>
<td>Expectation of loss expressed as the probability that a particular threat will exploit a particular vulnerability with a particular consequence [ISA 2009]</td>
</tr>
<tr>
<td><strong>Security risk assessment</strong></td>
<td>Overall process of risk identification, risk analysis and risk evaluation [ISO 2005]</td>
</tr>
<tr>
<td><strong>Security risk management</strong></td>
<td>The act or practice of dealing with risk. It includes planning for risk, assessing (identifying and analyzing) risk areas, developing risk handling options, monitoring risks to determine how risks have changed, and documenting the overall risk management program [IEC 2008a]</td>
</tr>
<tr>
<td><strong>Security standards</strong></td>
<td>Security standards are publicly available documents describing security best practices in engineering or management.</td>
</tr>
<tr>
<td><strong>Threat</strong></td>
<td>Potential for violation of security, which exists when there is a circumstance, capability, action, or event that could breach security and cause harm [ISA 2009]</td>
</tr>
<tr>
<td><strong>Trust boundary</strong></td>
<td>Boundary between systems or components with different levels of trust.</td>
</tr>
<tr>
<td><strong>Vulnerability</strong></td>
<td>Flaw or weakness in a system's design or implementation, or operation and management that could be exploited to violate the system's integrity or security policy [ISA 2009]</td>
</tr>
<tr>
<td><strong>Vulnerability handling</strong></td>
<td>The act or practice of dealing with vulnerabilities. It includes establishing contacts for reporting vulnerabilities and the coordinated correction of vulnerabilities.</td>
</tr>
</tbody>
</table>
### Appendix D: Relationship of Security by Design Practices to CMMI-DEV V1.3 Practices

<table>
<thead>
<tr>
<th>Specific Practices</th>
<th>REQM</th>
<th>PP</th>
<th>PMC</th>
<th>SAM</th>
<th>MA</th>
<th>PPQA</th>
<th>CM</th>
<th>RD</th>
<th>TS</th>
<th>PI</th>
<th>VER</th>
<th>VAL</th>
<th>RSKM</th>
<th>IPM</th>
<th>DAR</th>
<th>OPF</th>
<th>OPD</th>
<th>OT</th>
</tr>
</thead>
<tbody>
<tr>
<td>SP 1.1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SP 1.2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SP 1.3</td>
<td>SMP SP 1.3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SMP SP 1.1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SP 1.4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SW SP 1.1</td>
<td></td>
<td>SMP SP 1.1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SP 1.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SW SP 1.1</td>
<td></td>
<td>SMP SP 1.2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SP 1.6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>OPS SP 1.2</td>
<td></td>
<td>SMP SP 1.2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SP 1.7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>OPS SP 1.2</td>
<td></td>
<td>SMP SP 1.3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SP 2.1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>OPS SP 1.3</td>
<td></td>
<td>SMP SP 1.3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SP 2.2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>OPS SP 1.3</td>
<td></td>
<td>SMP SP 1.3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SP 2.3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SMP SP 2.3</td>
<td></td>
<td>SMP SP 1.3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SP 2.4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SMP SP 2.2</td>
<td></td>
<td>SMP SP 1.3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SP 2.5</td>
<td>SMP SP 1.2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SMP SP 1.2</td>
<td></td>
<td>SMP SP 1.3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SP 2.6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SMP SP 2.2</td>
<td></td>
<td>SMP SP 1.3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SP 2.7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SMP SP 2.2</td>
<td></td>
<td>SMP SP 1.3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SP 3.1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SMP SP 2.3</td>
<td></td>
<td>SMP SP 1.2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SP 3.2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SMP SP 2.2</td>
<td></td>
<td>SMP SP 1.2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SP 3.3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SMP SP 2.2</td>
<td></td>
<td>SMP SP 1.2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SP 3.4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SMP SP 2.2</td>
<td></td>
<td>SMP SP 1.2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SP 3.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>SMP SP 2.2</td>
<td></td>
<td>SMP SP 1.2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

No relationship to CMMI-DEV: OPS SP 1.5, SMP SP 1.4, SMP SG 2, SRT SP 1.3, SRT SP 1.4