Engineering
Safety- and Security-Related
Requirements for
Software-Intensive Systems

6th International Workshop on Software
Engineering for Secure Systems (SESS’10)
Workshop at the 32nd ICSE Conference, Cape
Town, South Africa

Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA  15213

Donald Firesmith
2 May 2010
This work was created in the performance of Federal Government Contract Number FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center. The Government of the United States has a royalty-free government-purpose license to use, duplicate, or disclose the work, in whole or in part and in any manner, and to have or permit others to do so, for government purposes pursuant to the copyright license under the clause at 252.227-7013.

This Presentation may be reproduced in its entirety, without modification, and freely distributed in written or electronic form without requesting formal permission. Permission is required for any other use. Requests for permission should be directed to the Software Engineering Institute at permission@sei.cmu.edu.

NO WARRANTY

THIS MATERIAL OF CARNEGIE MELLON UNIVERSITY AND ITS SOFTWARE ENGINEERING INSTITUTE IS FURNISHED ON AN “AS-IS” BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.
Contents

Three Disciplines
Challenges
Fundamental Concepts
Types of Safety- and Security-related Requirements
Consistent Common Processes
  • Safety and Security Processes Driving Requirements Process
Conclusion
Three Disciplines:
Requirements, Safety, and Security Engineering
Three Related Disciplines

Safety Engineering

the engineering discipline within systems engineering concerned with lowering the risk of *unintentional unauthorized* harm to valuable assets to a level that is acceptable to the system’s stakeholders by preventing, detecting, and reacting to such harm, mishaps (i.e., accidents and incidents), hazards, vulnerabilities, and safety risks

Security Engineering

the engineering discipline within systems engineering concerned with lowering the risk of *intentional unauthorized* harm to valuable assets to a level that is acceptable to the system’s stakeholders by preventing, detecting, and reacting to such harm, misuses (i.e., attacks and incidents), threats, vulnerabilities, and security risks

Requirements Engineering

the engineering discipline within systems/software engineering concerned with identifying, analyzing, reusing, specifying, managing, verifying, and validating goals and requirements (including safety- and security-related requirements)
Challenges: Combining Requirements, Safety, and Security Engineering
Challenges

Requirements engineering, safety engineering, and security engineering have different:

- Communities
- Disciplines with different training, books, journals, and conferences
- Professions with different job titles
- Fundamental underlying concepts and terminologies
- Tasks, techniques, and tools

Safety and security engineering are:

- Typically treated as secondary specialty engineering disciplines
- Performed separately from, largely independently of, and lagging behind the primary engineering workflow: (requirements, architecture, design, implementation, integration, testing, deployment, sustainment)
Challenges

Current separate methods for performing requirements, safety, and security engineering are inefficient and ineffective.

Separation of requirements engineering, safety engineering, and security engineering:

- Causes *poor* safety- and security-related requirements that are often:
  - Vague, unverifiable, unfeasible, architectural and design constraints
  - Capabilities or goals rather than requirements
  - Inadequate and too late to drive architecture and testing
- Makes it unnecessarily harder to achieve certification and accreditation
Challenges

Poor requirements are a primary cause of more than half of all project failures (defined in terms of):

- Major cost overruns
- Major schedule overruns
- Major functionality Not delivered
- Large number of defects delivered
- Delivered systems that are never used

Poor requirements are one major root cause of many (or most) accidents involving software-intensive systems.

Most mandated security “requirements” are actually constraints such as:

- Security functions or subsystems
- Industry “best practices”
Challenges

How safe and secure is safe and secure *enough*?

Situation cries out for process improvement:

- Better consistency between safety and security engineering
  - More consistent concepts and terminology
  - Reuse of techniques across disciplines
  - Less unnecessary overlap and avoidance of redundant work
- Better collaboration:
  - Between safety and security engineering
  - With requirements engineering
- Better safety- and security-related requirements
Fundamental Concepts:
A Foundation for Understanding
Quality Model

Architectural Components

System

defines the meaning of the quality of a

Quality Model

defines the meaning of a specific type of quality of a

Quality Characteristics

Quality Attributes

are measured along

Quality Measurement Scales

measure quality along

Quality Measurement Methods

are measured using

Internal Quality Characteristics

External Quality Characteristics
Defensibility Quality Attributes

- Occurrence of Unauthorized Harm
- Occurrence of Abuse (Mishap, Misuse, or Incident)
- Existence of External Abuser
- Existence of Internal Vulnerability
- Existence of Danger (Hazard or Threat)
- Existence of Defensibility Risk

Defensibility Attribute

Problem Type
- Problem Prevention
- Problem Detection
- Problem Reaction
- Problem Adaptation

Solution Type
- Harm Arrest
- Mitigation
- Recovery
- Analysis
- Counterattack (Security and Survivability)

Robustness

Safety

Security

Survivability

Defensibility

Quality Characteristic

Quality Attribute

Quality Model

measures quality along a

is measured along a

defines the meaning of the quality of a

System

Quality Measurement Scale

Quality Measurement Method

Defensibility Attribute

Engineering Safety- & Security-Related Requirements
Donald Firesmith, 2 May 2010
© 2010 Carnegie Mellon University

Software Engineering Institute | Carnegie Mellon
Defensibility

the quality characteristic capturing the degree to which the system:

• Properly prevents, detects, reacts to, and adapts to:
  — Unintended and unauthorized harm to valuable assets due to the occurrence of
  — Abuses enabled by the existence of
  — Dangers
• Has defensibility risks that are acceptably low to its stakeholders

Safety and security are defined in a similar manner by replacing:

• Abuse with either mishap (safety) or misuse (security)
• Danger with either hazard (safety) or threat (security)
• Defensibility risks with safety risks and security risks
Unauthorized Harm to Valuable Assets

Stakeholders

have an interest in the

System

must defend

Value

Unauthorized Harm

may occur to

Valuable Assets

People

Human Beings

Roles Played

Organizations

Development

Owner

Supplier

User

Property

Tangible Property

Intangible Property

Environment

Private Property

Public Property

Commercial Property

Services
Types of Harm

- Unintentional (Accidental) Harm
- Attacker-Caused (Malicious) Harm

Authorized Harm
- Unauthorized Harm

Valuable Assets may occur to Harm
- Direct Harm
- Indirect Harm

Harm to People
- Death
- Injury
- Illness
- Kidnap
- Corruption (bribery or extortion)
- Hardship

Harm to Organizations
- Bankruptcy
- Lost Market Share
- Lost Profits
- Loss of Reputation

Harm to Property
- Destruction
- Damage
- Corruption
- Theft
- Unauthorized Access
- Unauthorized Disclosure

Harm to the Environment
- Destruction
- Damage
- Loss of Use

Harm to a Service
- Corruption
- Unauthorized Usage (Theft)
- Accidental Loss of Service
- Denial of Service (DOS)
- Repudiation of Transaction

- e.g., caused to enemy forces by weapons systems

© 2010 Carnegie Mellon University
Types of Abuses

- Abuses
  - Defensibility Events
  - Mishaps (Safety)
    - Accidents
    - Safety Incidents
      - Unauthorized Harm
  - Misuses (Security)
    - Successful Civilian Attacks
    - Security Incidents
      - Unsuccessful Attacks
      - Probes
  - Survivability Abuses
    - Military Attacks
    - Survivability Incidents

cause

- Accidents
cause

- Safety Incidents
cause

- Successful Civilian Attacks
cause

- Security Incidents
cause

- Unsuccessful Attacks
cause

- Probes
cause
Vulnerabilities

- Defenses
  - Dangers
    - are partially defined in terms of the existence of system-internal
  - eliminate or mitigate

- Abusers
  - Abusers
    - exploit
      - typically cause
        - Nonmalicious Abusers
        - Malicious Abusers
          - desire

- Abuses
  - may cause

- Unauthorized Harm
  - may cause
    - Stakeholders
      - have an interest in the
        - have
          - Stakeholder Needs
            - must meet
              - System
                - must defend
                  - may occur to

- Valuable Assets
  - value
    - Stakeholder Needs
      - have
        - have
          - System
            - define types of ‘quality’ of the
              - Quality Factors
                - Defensibility
Types of Abusers

- **System Maintainer**
- **Manager**
- **System Developer**
- **System Operator**
- **User**

**Unintentional Human Abuser**

**Non-malicious External System**

**Aspect of the Natural Environment**

- **Cracker** (malicious hacker)
- **Professional Criminal**
- **Terrorist**

**System-Internal Condition**

**System-External Condition**

- **Vulnerability**

**Unintentional Abuser (Safety)**

**Intentional Abuser (Security)**

- **Malware**
- **Defensibility Event**

**Abuse**

- **Safety Abuse**
- **Security Abuse**

**Abuser**

- **Danger**
- **Hazard (Safety)**
- **Threat (Security)**

**System Condition**

- **Malware System**
- **Malware**
- **Software Malware**
- **Hardware Malware**

**Danger**

- **Virus**
- **Worm**
- **Spyware**
- **Adware**
- **Zombie**
- **Zombie Army**
- **Ransomware**
- **Scareware**

- **Malicious**

**External System**

**Foreign Intelligence Agency**

**Attacker**

- **Industrial Spy**

**Event**

- **Accident (Safety)**
- **Attack (Security)**
- **Safety Incident**
- **Security Incident**
- **Defensibility Event**

**Engineerings Safety & Security-Related Requirements**

Donald Firesmith, 2 May 2010

© 2010 Carnegie Mellon University
Dangers

Defensibility
Risks

Dangers

is the expected amount of

Defensibility

Risks

can be estimated using the probability of

Dangers

are partially defined in terms of the existence of system-external

Abusers

typically cause

Malicious
Abusers

exploit

desire

Nonmalicious
Abusers

Vulnerabilities

may enable the occurrence of

Abuses

may cause

quality of the

Stakeholders

exist in the

System

Unauthorized
Harm

define types of

Stakeholder
Needs

must meet

may occur to

Value

Stakeholders

have an interest in the

have

System

must defend

may defend

Valuable Assets

value

Stakeholder
Needs

must meet

have

Firesmith

May 2010

© 2010 Carnegie Mellon University
Risk in terms of Software Degree of Control

- Dangers may result in Abuses, which may cause Unauthorized Harm.
- Unauthorized Harm may occur to Valuable Assets.

- Defensibility Risk is due to Software Degree of Control.
- Software Degree of Control is estimated in terms of Harm Severity.

Risk categorizes amount of Abuses, Unauthorized Harm, Valuable Assets, and Dangers. Harm Severity is due to the software’s control over occurrence of Abuses.
Safety- and Security-Related Requirements
Types of Safety- and Security-Related Requirements

Too often only a Single Type of Requirements is considered.

Not just:

• Specific Types of Non-Functional Requirements (NFRs):
  – Safety and Security Requirements are Quality Requirements are NFRs
• Safety- and Security-Significant Functional, Data, and Interface Requirements
• Architecture and Design Constraints
• Safety and Security Functions/Subsystems
• Software Requirements
• Constraints on Functional Requirements

Reason for Presentation Title

Safety- and Security-Related Requirements for Software-Intensive Systems
Types of Defensibility-Related Requirements

- Safety Requirements
  - Safety-Significant Requirements
  - Safety-Related Requirements
  - System Requirements
- Security Requirements
  - Security-Significant Requirements
  - Security-Related Requirements
- Defensibility Requirements
  - Defensibility-Significant Requirements
  - Defensibility-Related Requirements
  - Defensibility-Constraints
- System Requirements
- Defensibility-Related Requirements
- Safety-Related Requirements
- Security-Related Requirements
- Safety-Significant Requirements
- Security-Significant Requirements
- Defensibility-Significant Requirements
- Safety-Related Requirements
- Security-Related Requirements
Four Types of Defensibility-Related Requirements

1. Intolerable Risk Requirements \( S AL = 4 \)
2. High Risk Requirements \( S AL = 3 \)
3. Moderate Risk Requirements \( S AL = 2 \)
4. Low Risk Requirements \( S AL = 1 \)

Safety/Security Assurance Level (SAL)

Defensibility Function / Subsystem Requirements

Defensibility Requirements

Functional Requirements

Quality Requirements

Data Requirements

Interface Requirements

Constraints

Defensibility - Independent Requirements \( S AL = 0 \)

Defensibility - Significant Requirements \( S AL = 1 - 4 \)

System Requirements

Primary Mission Requirements

Supporting Requirements

Safety/Security Requirements

Engineering Safety- & Security-Related Requirements
Donald Firesmith, 2 May 2010
© 2010 Carnegie Mellon University
Example Safety- and Security-Related Requirements (Actually Goals – Requirements are more specific)

Safety / Security Requirement:

“When in mode V, the system shall limit the occurrence of accidental harm of type W to valuable assets of type X to an average rate of no more than Y asset value per Z time duration.”

“When in mode X, the system shall detect misuses of type Y an average of at least Z percent of the time.”

Safety / Security Significant Requirement

“The system shall automatically transport passengers between stations.”

“The system shall enable users to update their personal information.”

Safety / Security Function / Subsystem Requirement

“The system shall include a fire detection and suppression subsystem.”

“The system shall support the encryption/decryption of sensitive data.”

Safety / Security Constraint

“The system shall not contain any of the hazardous materials in Table X.”

“The system shall use passwords for user authentication.”
Common Process:  
A Basis for Effective Collaboration
Overall Defensibility Engineering Method

Defensibility Program Planning → Defensibility Analysis

Defensibility Policy Development → Compliance Assessment

Defensibility Monitoring → Abuse Investigation

Defensibility Certification and Accreditation

© 2010 Carnegie Mellon University
Defensibility Analysis

Defensibility Analysis performs Stakeholder Analysis, Asset Analysis, Abuse Analysis, System Analysis, and Vulnerability Analysis. These analyses collaborate with the Requirements Team to develop Defensibility-Work Products.

The Defensibility-Related Requirements are developed by the Requirements Team and are validated by the Security-Related Requirements. The Defensibility Analysis also performs Danger Analysis, Risk Analysis, Significance Analysis, and Defense Analysis.

The Defensibility Analysis collaborates with the Safety Team and the Security Team, and participates in Stakeholder Analysis, Asset Analysis, Abuse Analysis, System Analysis, and Vulnerability Analysis.

The Defensibility Analysis supports the Safety Team, the Security Team, the Subject Matter Experts, and the Stakeholders.

Safety and Security Certification Repositories are used to store the results of the Defensibility Analysis.
Asset Analysis

Safety and Security Engineering

Requirements Engineering

Asset Requirements
- Asset Prevention Requirements
- Asset Detection Requirements
- Asset Reaction Requirements
- Asset Safety Requirements
- Asset Security Requirements

Asset Analysis

Project Documentation (RFP, Contract, ConOps)
- Generic / Reusable Asset Tables
- Standard / Reusable Asset Value and Harm Severity Categories
- Generic / Reusable Asset Value and Harm Tables
- Standard / Reusable Asset-Harm Goals

Stakeholders
- provide input during

Safety Team
- collaborates with

Security Team
- provides input during

Requirements Team
- performs

Preparation
- Standard / Reusable Asset-Harm Requirements

Asset Identification
- Asset Stakeholder Table
- Asset Usage Table
- Asset Value and Harm Table
- Asset-Harm Goals

Asset Modeling
- Asset Table

Asset Goal Identification
- Safety and Security Certification Repositories

Safety and Security Engineering

Subject Matter Experts

Safety Team

Security Team

Stakeholders
Abuse (Misuse and Mishap) Analysis

**Requirements Engineering**

- Abuse Goals
- Abuse Cases
- Abuse Trees
- Abuse Identification
- Preparation

**Safety and Security Engineering**

- Abuse Goals
- Abuse Cases
- Abuse Trees
- Abuse Identification
- Preparation

**Abuse Analysis**

- Safety Team
- Security Team
- Stakeholders
- Subject Matter Experts

**Requirements Team**

- Abuse Prevention Requirements
- Abuse Detection Requirements
- Abuse Reaction Requirements
- Abuse (Misuse) Security Requirements
- Abuse (Mishap) Safety Requirements

**Requirements Development**

- Abuse Goals
- Abuse Cases
- Abuse Trees

**Requirements Validation**

- Support
- Perform

**Abuse Requirements**

- Safety Team
- Security Team
- Subject Matter Experts
- Stakeholders

**Generic Reusable Work Products:**
- Abuse Lists & Tables
- Abuse Cases & Trees
- Abuse Goals
- Enterprise and Reference Architectures

**Project-Specific Work Products:**
- RFP and Contract
- ConOps
- Rqmts Repository
- Architecture

**Requirements & Architecture Work Products:**
- RFP and Contract
- ConOps
- Rqmts Repository
- Architecture

**Safety and Security Certification Repositories**

**Abuse**

- Analysis
- Goals
- Identification
- Identification
- Prevention
- Detection
- Reaction
- Certification
Vulnerability Analysis

Architecture, Designers, and Implementers

Quality Engineers, Testers, and Maintainers

Vulnerability Analysis

Concept of Operations (ConOps)

Requirements, Glossaries, and Domain Models

System Architectural Representations

Abuse Table

Abuse Cases

Abuse Trees

Safety Team

collaborates with

Security Team

Safety and Security Engineering

Preparation

Vulnerability Identification

Vulnerability Modeling

Vulnerability Goal Identification

Vulnerability List

Vulnerability and Abuse Table

Vulnerability and Defense Table

Vulnerability Goals

Requirements Development

Requirements Validation

Stakeholders

Safety Team

Security Team

Subject Matter Experts

Safety and Security Certification Repository

Generic / Reusable Vulnerability Requirements and Constraints

Requirements Team

Vulnerability Goals

Support

Perform

Safety Vulnerability Requirements

Security Vulnerability Requirements

Vulnerability Constraints

Safety Vulnerability Constraints

Security Vulnerability Constraints

Abuse Cases

Abuse Trees
Abuser Analysis

Subject Matter Experts

Safety Team

Security Team

collaborates with

Stakeholders

provide input during

Project Documentation (RFP, Contract, ConOps)

Generic / Reusable Abuser Lists

Generic / Reusable Abuser Profiles

Generic / Reusable Abuser-Related Goals

Abuser Analysis

performs

Preparation

Abuser Identification

Abuser Modeling

Abuser Goal Development

Safety and Security Certification Repositories

Abuser List

Asset Abuser Table

Abuser Profiles

Abuser Abuse Asset Harm Table

Abuser-Related Goals

Requirements Development

Requirements Validation

Support

Safety and Security Engineering

Safety Team

Security Team

Subject Matter Experts

Stakeholders

Abuser Protection Requirements

Abuser Detection Requirements

Abuser Reaction Requirements

Safety Abuser Requirements

Security Abuser Requirements

Requirements Engineering
Danger Analysis

Subject Matter Experts

Stakeholders

provide input during

perform

Safety and Security Documentation

Other System Documentation

Non-System Documentation

Generic / Reusable Danger Lists

Generic / Reusable Danger Profiles

Generic / Reusable Danger Likelihoods

Safety and Security Certification Repositories

Danger Analysis

Danger Modeling

Danger Goal Identification

Danger Likelihood Analysis

Cause Analysis

Root Cause(s) Analysis

Common Cause Analysis

Danger Effects Analysis

Danger Profiling

Danger (Hazard & Threat) List

Danger Profiles

Danger Cause & Effects Diagrams

FMECA Tables

Danger Goals

Requirements Development

Requirements Validation

Requirements Engineering

Danger Protection Requirements

Danger Detection Requirements

Danger Reaction Requirements

Safety Hazards Requirements

Security Threats Requirements

Safety Team

Security Team

Subject Matter Experts

Requirements Team

Stakeholders

Engineering Safety- & Security-Related Requirements
Donald Firesmith, 2 May 2010
© 2010 Carnegie Mellon University
Defensibility Risk Analysis

Safety and Security Engineering

Risk Analysis

Stakeholders
- Subject Matter Experts
- Security Team
- Safety Team

Requirements Team
- Risk Protection Requirements
- Risk Detection Requirements
- Risk Reaction Requirements
- Safety Risk Requirements
- Security Risk Requirements

Risk Analysis performs
- Preparation
- Risk Identification
- Risk Modeling
- Risk Goal Identification
- Defensibility Risk Goals

Risk Lists
- Requirements Development
- Requirements Validation

Generic / Reusable Risk Tables
- Abuse Table
- Abuse Trees
- Abuse Cases
- Danger Profiles
- Danger Cause and Effects Diagrams

Safety and Security Certification Repositories

Support

Validation

Risk Detection Requirements

Risk Reaction Requirements

Safety Risk Requirements

Security Risk Requirements
Defensibility Significance Analysis

- Subject Matter Experts
- Stakeholders
- Safety Team
  - collaborates with
  - provides input during
- Security Team
  - provides input during
- Safety and Security Goals
- Project-Specific Safety and Security Assurance Level (SAL) Definitions
- Project-Specific Safety and Security Evidence Assurance Level (SEAL) Definitions
- Safety and Security Certification Repositories
- Significance Analysis
  - performs
- Requirements Engineering
- Requirements Team
  - performs
- Requirements Identification
- Requirements Analysis
- Requirements Repository
- SEAL Categorization
- SEAL Definition
- SEAL Allocation
- Architecture Representations
- Architecture Verification
- Architecture Team
- Stakeholders
  - collaborates in the performance of
- Subject Matter Experts
- Safety Team
- Security Team
- Safety and Security Certification Engineering
- Architecture Engineering
- Software Engineering Institute
- Carnegie Mellon
Defense Analysis

Defense Analysis performs:
- Defense Type Identification
- Defense Functionality Identification
- Defense Selection
- Defense Adequacy Analysis
- Countermeasure and Safeguard Type Lists
- List of Defense Functions / Subsystems
- Requirements Identification
- Market Research
- Vendor Trade Studies
- Countermeasure and Safeguard Selection Reports
- Requirements Analysis
- Requirements Validation

Collaborators:
- Safety and Security Engineering
- Requirements Team
- Architecture Team
- Stakeholders
- Subject Matter Experts
- Safety Team
- Security Team

Repositories:
- Safety Function/Subsystem Requirements
- Security Function/Subsystem Requirements
- Defense Function/Subsystem Requirements
- Defense Constraints
- Security Constraints

Tasks:
- Defense Adequacy Analysis
- Countermeasure and Safeguard Selection Reports
- Requirements Validation
- Requirements Team

Subject Matter Experts collaborate with:
- Safety Team
- Security Team

Stakeholders provide input during:
- Safety and Security Requirements
- Generic / Reusable Safeguard and Countermeasure Lists
- Standard Defense Functionality and Constraint Requirements
- Safety and Security Risks

Safety and Security Certification Repositories
Conclusion:

*Process Improvement Recommendations*
Process Improvement Recommendations


Better Integrate Safety and Security Processes:

- Concepts and Terminology
- Techniques and Work Products
- Provide Cross Training

Better Integrate Safety and Security Processes with Requirements Process:

- Early during Development Cycle
- Clearly define Team Responsibilities
- Provide Cross Training

Develop all types of Safety- and Security-related Requirements.

Ensure that these Requirements have proper Properties.
Any Questions?

For more information, contact:

Donald Firesmith
Acquisition Support Program (ASP)
Software Engineering Institute (SEI)
Pittsburgh, Pennsylvania, USA 15213
412-268-6874
dgf@sei.cmu.edu