Software Engineering Institute
Carnegie Mellon University

# Security Engineering Risk Analysis (SERA)

## Designing Security into Software-Reliant Systems

Software is a growing component of modern business- and mission-critical systems. As organizations become more dependent on software, security-related risks to their organizational missions are also increasing. Traditional security-engineering approaches rely on addressing security risks during the operation and maintenance of software-reliant systems. However, the costs required to control security risks increase significantly when organizations wait until systems are deployed to address those risks. It is more cost effective to address software security risks as early in the lifecycle as possible.

As a result, researchers from the CERT® Division at Carnegie Mellon® University's Software Engineering Institute (SEI) have started investigating early lifecycle security risk analysis. Initial research suggests that applying traditional security risk-analysis methods earlier in the lifecycle will not solve the problem because those methods cannot handle the inherent complexity of modern cybersecurity attacks.

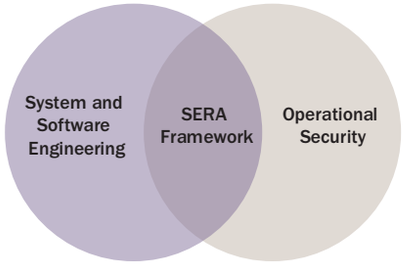
Traditional methods are based on a simple, linear view of risk that assumes a single threat actor exploits a single vulnerability in a single system to cause an adverse consequence. In reality, multiple actors exploit multiple vulnerabilities in multiple systems as part of a complex chain of events. Traditional methods are often unable to analyze the complex cybersecurity attacks effectively. New approaches are needed.

SEI researchers developed the Security Engineering Risk Analysis (SERA) Framework, a security risk-analysis approach, to advance the state-of-the-practice. The SERA Framework incorporates two important technical perspectives: (1) system and software engineering and (2) operational security. The framework requires system and software engineers to consider operational security risks early in the lifecycle. This approach blends multiple technical disciplines to define an engineering-oriented risk-analysis practice consistent with the NIST Risk Management Framework (RMF).

### Minimizing Design Weaknesses

Software assurance is defined as a level of confidence that software functions as intended and is free of vulnerabilities, either intentionally or unintentionally designed or inserted as part of the software, throughout the life cycle. The pursuit of software assurance is a goal that must be translated into practical methods that acquirers, designers, and developers can apply throughout the acquisition-and-development life cycle. The SERA Framework is designed to meet this need.

Operational security vulnerabilities have three main causes: (1) design weaknesses, (2) implementation/coding errors, and (3) system configuration errors. The SERA Framework is focused on addressing design weaknesses. Correcting these weaknesses as early as possible is especially important. Remediation normally requires extensive redesign of the system, which is costly and often proves to be impractical after deployment. As a result, software-reliant systems with design weaknesses are often allowed to operate under a high degree of residual security risk, putting their associated operational missions in jeopardy.



## SERA Framework

The SERA Framework is designed to (1) address cybersecurity risks in complex cyber-physical systems; (2) identify security requirements and acquisition needs early in the life cycle; (3) consider risk factors in system-of-systems environments and the software supply chain; and (4) be consistent with various guidelines, specifications, and laws (e.g., DoD Instruction 8510.01 and NIST 800-37). The SERA Framework comprises the following four tasks:

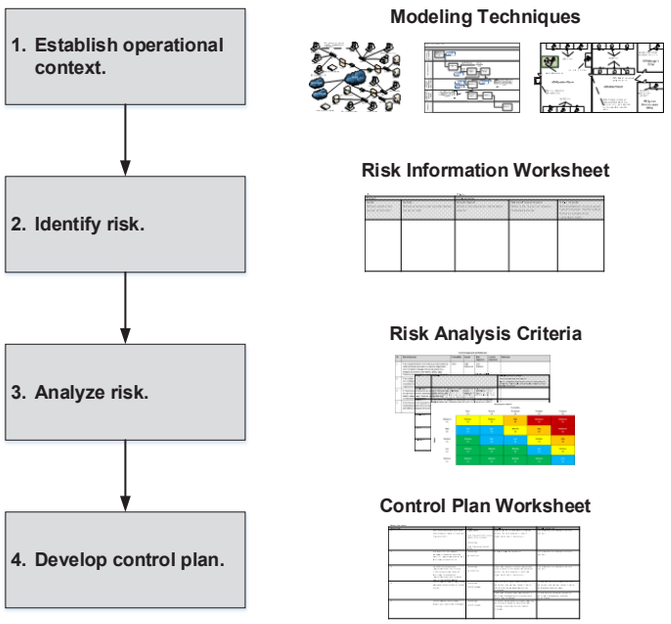
Task 1: Establish operational context—uses modeling techniques to define the operational context for the analysis

Task 2: Identify risk—transforms security concerns into a distinct, tangible security risk scenarios that can be described and measured

Task 3: Analyze risk—evaluates each security risk scenario in relation to predefined criteria to determine its probability, impact, and risk exposure.

Task 4: Develop control plan—establishes a plan for controlling a selected set of security risk scenarios

The SERA Framework can be self-applied or facilitated with an external team of three to five people.




## Key Differentiators

The SERA Framework incorporates three key design features that differentiate it from other security risk assessments. The first is the use of operational models. People conducting traditional security-risk assessments rely on their tacit understanding of the operational context in which a software-reliant system must operate. SEI experience indicates that tacit assumptions are often incorrect or incomplete, which adversely affects the results of a security risk analysis. The SERA Framework uses operational models to describe a system's operational context explicitly and establish a baseline of operational performance to inform the risk identification and analysis.

The second feature is the semantic structure used to document security risks. Most traditional assessments rely on linear, simplistic formats for recording risks (e.g., if-then statements). These basic structures are unable to capture the complexities and nuances of modern cybersecurity attacks. To address this deficiency, the SERA Framework uses scenarios to document a cybersecurity risk. A security risk scenario conveys information describing how one or more threat actors can exploit multiple vulnerabilities in multiple systems to cause adverse consequences for stakeholders.

The third feature is the shared view of a system, its operational context, and its associated security risks. This shared view is presented in a format that is easily understood by multiple stakeholders, including system and software engineers, security experts, and program managers. As a result, complex security risks can be evaluated effectively and then prioritized based on their impact to the operational mission of the system.

## For More Information

http://www.cert.org/cybersecurity-engineering/research/security-engineering-risk-analysis.cfm

## About

For more than 25 years, the CERT Division of the Software Engineering Institute has been a leader in cybersecurity. Originally focused on incident response, the division has expanded into areas of network situational awareness, malicious code analysis, secure coding, resilience management, insider threat, digital intelligence and investigation, and workforce development.



## Contact Us

Software Engineering Institute
4500 Fifth Avenue, Pittsburgh, PA 15213-2612

**Phone:** 412.268.5800 | 888.201.4479
**Web:** www.sei.cmu.edu | www.cert.org
**Email:** info@sei.cmu.edu