



Security Quality Requirements Engineering (SQUARE) Improving requirements identification, analysis, specification, and management

Related Web Site

www.cert.org/sse/square.html

For More Information

To learn more, please contact
Nancy Mead
P: 412-268-5756
E: nrm@sei.cmu.edu
Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213-2612

For General Information

For information about the SEI
and its products and services,
contact Customer Relations
P: 412-268-5800
F: 412-268-5758
www.sei.cmu.edu

Related SEI Publications

www.sei.cmu.edu/publications

Mead, N.; Hough, E.; & Stehney, T. *Security Quality Requirements Engineering* (CMU/SEI-2005-TR-009). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2005.

Chen, P.; Dean, M.; Ojoko-Adams, D.; Osman, H.; Lopez, L.; & Xie, N. *Systems Quality Requirements Engineering (SQUARE) Methodology: Case Study on Asset Management System* (CMU/SEI-2004-SR-015). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2004.

Gordon, D.; Stehney, T.; Wattas, N.; & Yu, E. *System Quality Requirements Engineering (SQUARE): Case Study on Asset Management System, Phase II* (CMU/SEI-2005-SR-005). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2005.

Chung, L.; Hung, F.; Hough, E.; & Ojoko-Adams, D. *Security Quality Requirements Engineering (SQUARE): Case Study Phase III* (CMU/SEI-2006-SR-003). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2006.

Related Books

Allen, Julia H.; Barnum, Sean; Ellison, Robert J.; McGraw, Gary; & Mead, Nancy R. *Software Security Engineering: A Guide for Project Managers*. Boston, MA: Addison-Wesley, May 2008 (ISBN 032150917X).

Mouratidis, H. & Giorgini, P. *Integrating Security and Software Engineering*. Idea Group Publishing, www.idea-group.com, 2006.

© CERT is registered in the U.S. Patent and Trademark Office by Carnegie Mellon University. The Software Engineering Institute (SEI) is a federally funded research and development center sponsored by the U.S. Department of Defense and operated by Carnegie Mellon University.

© 2009 Carnegie Mellon University.
This work is supported by the Army Research Office through grant number DAAD19-02-1-0389 ("Perpetually Available and Secure Information Systems") to Carnegie Mellon University's CyLab.



SQUARE



Software Engineering Institute
Carnegie Mellon

Carnegie Mellon
CyLab
CONFIDENCE FOR A NETWORKED WORLD

The Problem

Requirements problems are the number one cause of why projects are significantly over budget and past schedule, have significantly reduced scope, deliver poor-quality applications, are little used once delivered, or are cancelled altogether. In particular, system quality requirements, such as security, are often poorly expressed and little analyzed, leading to inappropriate and incomplete system designs and implementations.

A recent study found returns on investment of 12 to 21 percent when security analysis and secure engineering practices are introduced early in the development cycle. The highest rate of return occurs when the analysis is performed during application design. Further, it is very difficult and expensive to significantly improve the security of an application after it is fielded in its operational environment.

The Project

The Security Quality Requirements Engineering (SQUARE) project is identifying and assessing processes and techniques to improve requirements identification, analysis, specification, and management. The project is also focusing on management issues associated with the development of good security requirements.

A framework for requirements engineering has been field tested in a series of client case studies, and the study results were published in three SEI reports (see “Related SEI Publications” at left). The baseline process is shown in the table on the reverse. Prototype tools have also been developed to support the process.

Benefits

Because many operational systems problems are traceable to requirements problems, we hope to enable the development of systems that are more secure and survivable by successfully using requirements engineering methods. In addition, we hope that this focus on security requirements will result in more predictable development activities and processes, as well as systems whose costs and schedules are more predictable.

2009 Plans

SQUARE will be extended to include privacy considerations and acquisition. A new more robust version of the SQUARE tool is under development in Carnegie Mellon University’s MSE Program. We plan to propose a workshop on security requirements engineering in conjunction with a major conference.

The SEI is seeking organizations in government, academia, and industry to participate in pilots and reviews of SQUARE practices and processes.

Security Quality Requirements Engineering (SQUARE)

Improving requirements identification, analysis, specification, and management

Security Requirements Elicitation and Analysis Process

| Number | Step | Input | Techniques | Participants | Output |
|--------|--|---|--|--|---|
| 1 | Agree on definitions | Candidate definitions from IEEE and other standards | Structured interviews, focus group | Stakeholders, requirements team | Agreed-to definitions |
| 2 | Identify assets and security goals | Definitions, candidate goals, business drivers, policies and procedures, examples | Facilitated work session, surveys, interviews | Stakeholders, requirements engineer | Assets and goals |
| 3 | Develop artifacts to support security requirements definition | Potential artifacts (e.g., scenarios, misuse cases, templates, forms) | Work session | Requirements engineer | Needed artifacts: scenarios, misuse cases, models, templates, forms |
| 4 | Perform risk assessment | Misuse cases, scenarios, security goals | Risk assessment method, analysis of anticipated risk against organizational risk tolerance, including threat analysis | Requirements engineer, risk expert, stakeholders | Risk assessment results |
| 5 | Select elicitation techniques | Goals, definitions, candidate techniques, expertise of stakeholders, organizational style, culture, level of security needed, cost benefit analysis, etc. | Work session | Requirements engineer | Selected elicitation techniques |
| 6 | Elicit security requirements | Artifacts, risk assessment results, selected techniques | Joint Application Development (JAD), interviews, surveys, model-based analysis, checklists, lists of reusable requirements types, document reviews | Stakeholders facilitated by requirements engineer | Initial cut at security requirements |
| 7 | Categorize requirements as to level (system, software, etc.) and whether they are requirements or other kinds of constraints | Initial requirements, architecture | Work session using a standard set of categories | Requirements engineer, other specialists as needed | Categorized requirements |
| 8 | Prioritize requirements | Categorized requirements and risk assessment results | Prioritization methods such as Triage, Win-Win | Stakeholders facilitated by requirements engineer | Prioritized requirements |
| 9 | Inspect requirements | Prioritized requirements, candidate formal inspection technique | Inspection methods such as Fagan, peer reviews | Inspection team | Initial selected requirements, documentation of decision-making process and rationale |