Software Engineering Institute
Carnegie Mellon University

SQUARE FREQUENTLY ASKED QUESTIONS

What is the main objective of the SQUARE process?

The main objective of SQUARE is to elicit and document security requirements.

What are the common approaches followed for developing high-level and detailed requirements using SQUARE?

Approach 1 is to think about classes of requirements. For example, for access control, you would consider whether there are assets that need to be protected with access control and how you would go about implementing it.

You could use a document such as The Open Web Application Security Project's (OWASP's) Development Guide[1] to get started. A list of candidate requirements areas can be found in the "Security Requirement Areas" section starting on page F-12 of Software Assurance in Acquisition: Mitigating Risks to the Enterprise.[2]

Approach 2 is to use scenarios. If you have normal user scenarios and intruder scenarios, it is pretty easy to understand what the security requirements should be. You need to understand the architecture in order to use this approach, however.

In both approaches it should be relatively easy to think about what is needed to block the threat or intrusion or what is needed to protect the assets. You should do this informally and not worry about writing requirements, statements, or use cases initially.

You might find that certain high-level requirements have mismatches that will be discovered only when the requirements have been detailed. Hence the requirements elicitation approach may become iterative.

What steps are followed in the SQUARE process?

- Agree on definitions.
- Identify assets and security goals.
- Develop artifacts to support the security requirements definition.
- Perform the risk assessment.
- Select elicitation technique(s).
- Elicit security requirements.
- Categorize requirements.

---

1 http://www.owasp.org/index.php/Category:OWASP_Guide_Project

2 https://buildsecurityin.us-cert.gov/resources/dhs-software-assurance-resources/software-assurance-in-acquisition--mitigating-risks-to-the-enterprise%20target=

- Prioritize requirements.
- Inspect requirements.

There are lots of security term and definition pairs. What definitions does the stakeholder need to filter out before distributing a definition set to the requirements elicitation (RE) team?

You should include all definitions that are relevant to the project, even if they are common terms. For example, different stakeholders have different ideas about what "availability" means, even though it is a common term. The objective here is to agree on definitions for terms that might be needed on the project, not for every security term that exists. Some of our case study reports show the definitions selected by clients. (See the appendix in *SQUARE-Lite: Case Study on VADSoft Project*.[3])

Why is SQUARE Step 2, "Identify assets and security goals," very important?

Without overall security goals for the project, it is impossible to identify the priority and relevance of any security requirements that are generated. The establishment of security goals scopes the rest of the SQUARE process.

For Step 2, is there any preferred technology such as unified modeling language (UML) or data flow diagrams (DFDs)? Can we use the diagrams to start modeling threats?

Since SQUARE presumes that the client is developing artifacts, it does not make a specific recommendation. In our case studies, we used Visio to create diagrams. UML may be a little too detailed for this purpose, so DFDs would be a better choice.

The diagrams can be used to start threat modeling. We used normal usage and intrusion scenarios for this purpose, creating flows through the diagrams. Again, this is your choice.

Are normal usage scenarios analogous to use cases, and are intrusion scenarios analogous to misuse cases?

Yes, but only to a certain extent. They are different ways of representation. The scenarios can be a little easier to explain to stakeholders who are not tech savvy, and use cases/UML are another way of looking at it. Use what you are comfortable with-it should not make a difference in terms of the results.

What types of artifacts should be collected?

- System architecture diagrams
- Use case scenarios/diagrams
- Misuse case scenarios/diagrams

---

3 http://www.sei.cmu.edu/publications/documents/08.reports/08sr017.html

- Attack trees

Can a technique other than misuse cases be used to model threats in Step 3?

Yes, any method that enables you to completely model the threats is fine.

Can we go directly from a use case to an attack tree, or should we do misuse cases first?

You do not need misuse cases in order to produce attack trees. You can produce attack trees directly by looking at the architecture, deciding on the types of attacks that might take place, and developing the trees. You do not need both misuse cases and attack trees.

However, if you are inexperienced, by doing both you may catch any missing threats. People with more math and logic backgrounds tend to like attack trees, whereas UML types tend to like misuse cases. The important thing is to be able to do traceability.

What risk assessment methods can be used with SQUARE?

The risk assessment methods that we identified for security follow. (There are others as well.)

- General Accounting Office model
- National Institute of Standards and Technology model (*recommended*)
- NSA's INFOSEC Assessment Technology
- Shawn Butler's Security Attribute Evaluation Method
- Carnegie Mellon University's "V-RATE" method
- Yacov Haimes' RFRM model (*recommended*)
- Carnegie Mellon University's Survivable Systems Analysis method
- Martin Feather's DDP model

References for the above methods can be found in the "Requirements Best Practices" section of the Build Security In (BSI) website.[4]

---

[4] https://buildsecurityin.us-cert.gov/articles/best-practices/requirements-engineering/requirements-engineering-annotated-bibliography

What are the steps in the NIST risk assessment method?

- Threat identification
- Vulnerability identification
- Control analysis
- Likelihood determination
- Impact analysis
- Risk determination

What elicitation techniques can be used with SQUARE?

These are the elicitation techniques that we identified in our case studies. (There are others as well; see " Requirements Elicitation Introduction" on the Build Security In website.[5])

- misuse cases
- Soft Systems Methodology (SSM)
- Quality Function Deployment (QFD)
- Controlled Requirements Expression (CORE)
- issue-based information systems (IBIS)
- Joint Application Development (JAD)
- feature-oriented domain analysis (FODA)
- critical discourse analysis (CDA)
- Accelerated Requirements Method (ARM)
- Structured/unstructured Interviews

How can Brainstorm, Organize, and Name (BON) be used to guide the security requirements elicitation process?

The team must have the risks/threats, assets, and security goals in front of them. A brainstorming session would result in a mapping between the security requirements and the risk/threats, assets, and security goals that were identified previously.

Alternatively, a specialist can "seed" the brainstorming process by developing some requirements ahead of the meetings that are called for in the BON method. Once the team sees a few example requirements, they are likely to come up with more. It's important to have stakeholder representatives involved, as they tend to see things differently. For example, a stakeholder in HR might be concerned with securing personnel data, whereas someone else might not consider that type of data particularly sensitive.

---

5 https://buildsecurityin.us-cert.gov/articles/best-practices/requirements-engineering/requirements-engineering-annotated-bibliography

What pitfalls are there in Step 6, "Elicit security requirements"?

The largest mistake that the requirements engineering team can make in this step is to elicit non-verifiable or vague, ambiguous requirements. Each requirement must be stated in a manner that will allow relatively easy verification once the project has been implemented. For instance, the requirement "The system shall improve the availability of the existing customer service center" is impossible to measure objectively. Instead, the requirements engineering team should encourage the production of requirements that are clearly verifiable and, where appropriate, quantifiable. A better version of the previously stated requirement would thus be "The system shall handle at least 300 simultaneous connections to the customer service center."

A second mistake that the requirements engineering team can make in this step is to elicit implementations or architectural constraints instead of requirements. Requirements are concerned with what the system should do, not how it should be done.

What are the joint responsibilities for the stakeholders and the requirements elicitation team while requirements are elicited?

- Encourage the generation of verifiable, and preferably quantifiable, security requirements
- Ensure that only requirements are generated and not implementations or architectural constraints

What are the crucial inputs while prioritizing requirements?

- Risk assessment
- Categorization

What prioritization techniques can be used with SQUARE?

These are the techniques that we identified. We had good success with the Analytical Hierarchy Process (AHP) and the prioritization method that is built into the Accelerated Requirements Method (ARM).

- Binary search trees
- Numeral assignment techniques
- Planning game
- 100-Point method
- Theory-W
- Triage
- Wiegers' Method
- Requirements Prioritization Framework
- Analytical Hierarchy Process

Why is the last step one of the most important elements in creating a set of accurate and verifiable security requirements?

The goal of the inspection is to find any defects in the requirements such as ambiguities, inconsistencies, or mistaken assumptions. Step 9 also serves as the last chance to remove any requirements from the working set.

Are there any alternative strategies to SQUARE?

Yes. There are other methods for various aspects of security requirements engineering. You can find a brief discussion of them in Chapter 3 of the book *Software Security Engineering: A Guide for Project Managers*[6] and also in *Integrating Security and Software Engineering: Advances and Future Visions*.[7]

---

6 http://www.softwaresecurityengineering.com/

7 http://www.igi-global.com/book/integrating-security-software-engineering/615

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

This material is based upon work funded and supported by the Department of Defense under Contract No. FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

DM-0004348