Assignment #1 – Secure Software Requirements

SQUARE (The Security Quality Requirements Engineering) was created by the CERT Program at Carnegie Mellon University's Software Engineering Institute.

Why the program at Carnegie Mellon University's Software Engineering institute came up with SQUARE was because security is an important step in software system and often not high priority in the development life cycle.

This makes big problems in company which results in inadequate analysis, cost overruns and vulnerabilities costing billions of dollars annually.

Today Company starting the security requirements and implements when they start implements design. And that is an issue.

To be more effective, security should be an integrated part of system development from the beginning. It should have the same priority as the system requirements.

SQUARE is a nine step security requirement list. Progress of SQUARE can take two to three months for a large project. Studies have shown that SQUARE is very effective.

You can find SQUARE in two different variants. SQUARE and SQUARE-Lite. The difference between this two is how many requirement steps it's contain and other small differences.

The one I will talk more about will be R-SQUARE which integrates reusability into the SQUARE methodology. Security requirements and goals can be made very reusable. In many different projects they can be written at the right level of abstraction.

The security requirements need to be addressed early in the development process. Otherwise the cost of catching and correcting the security requirements can be defected in deployed systems and be much more costly than it should.

SQUARE got a nine step list and SQUARE-Lite got a five step list to follow. So for a smaller project you should use SQUARE-Lite.

R-SQUARE has many benefits in contradistinction to SQUARE. The opportunity for using R-SQUARE is that many projects are using the same security requirements. This led us to reduced cost because you don't have to rewritten it again and then used in future projects. Also the written code have already been given thorough attention and inspected for quality.

More about security. Security you can break into five smaller more specific concepts. Confidentiality, Integrity, Availability, Accountability and Conformance. This five concepts have all very important tasks for making a whole system secure.

Confidentiality means that sensitive information like private data is protected against unauthorized disclosure. Only the owner of data can access to it, not even an administrator. You also encrypt data so that it can only be interpreted by the intended recipient.

Integrity secures that data not modified and needs to be immune to it. And make sure data been confirmed to come for a valid source.

Availability means that data and services must be available when they are requested. If something been interrupted on the way, system must recover and continue the operation fast without any side effects.

Accountability helps us to trace the person or persons making the attack on our system.

Conformance means that everyone who is involved in the system needs to be secure. The software should always operate as intended without variation.

Layered defenses:

The first layer of defense is to prevent attacks from taking place. And of course this means many steps of secure.

Second layer is detection. When an attack or accident dose occurs, that should always trigger a notification to system and users.

Response is the third layer of defense. When system detected an attack, system should automated activate some defense mechanisms and make a response.

Last layer is recovery. System is never completely secure. When a attacks succeeded system need to take recover of the lost. System must be returned to secure operation quickly.

Tim Emanuelsson (Te222ds) UD13