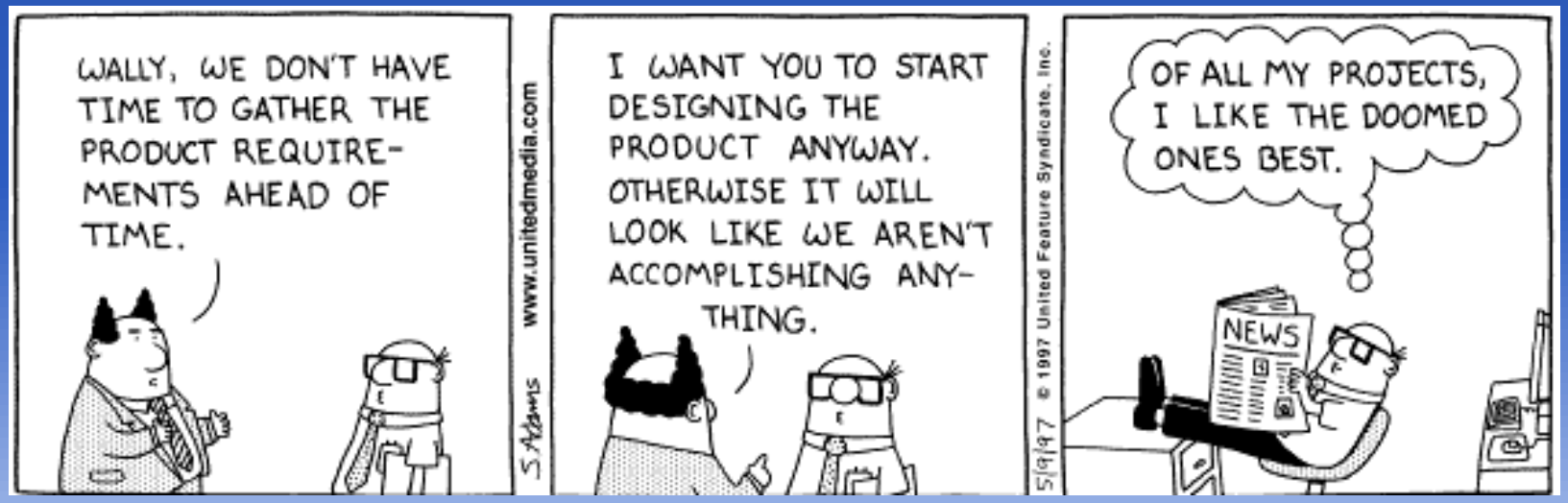
CS 493
Secure Software Systems
Ch 7 Security Requirements Planning



Dilbert

Dr. Williams
Central Connecticut State University

Goals

- Identify security requirements for a proposed system.
- Identify how to secure existing requirements.
- Identify and prioritize stakeholders in the system.
- Apply accountability of stakeholders to the system scope.
- Determine elements of requirements that document and assert security.

You, Me, and the SDLC

The Software Development Life Cycle:

1. Requirements Analysis
2. Design
3. Construction
4. Testing
5. Installation
6. Operation
7. Maintenance

Our project

- This project is from Chapter 7 page 165 and will be (one of) our running examples for the next several classes

Establishing Stakeholders

- A **stakeholder** is anyone with an interest in the project or anyone affected by the project.
- Secondary stakeholders are those who are indirectly affected by the project or those who may indirectly affect the project.
- The process of **stakeholder analysis** is used to determine the members of each group.

Example Stakeholder Analysis

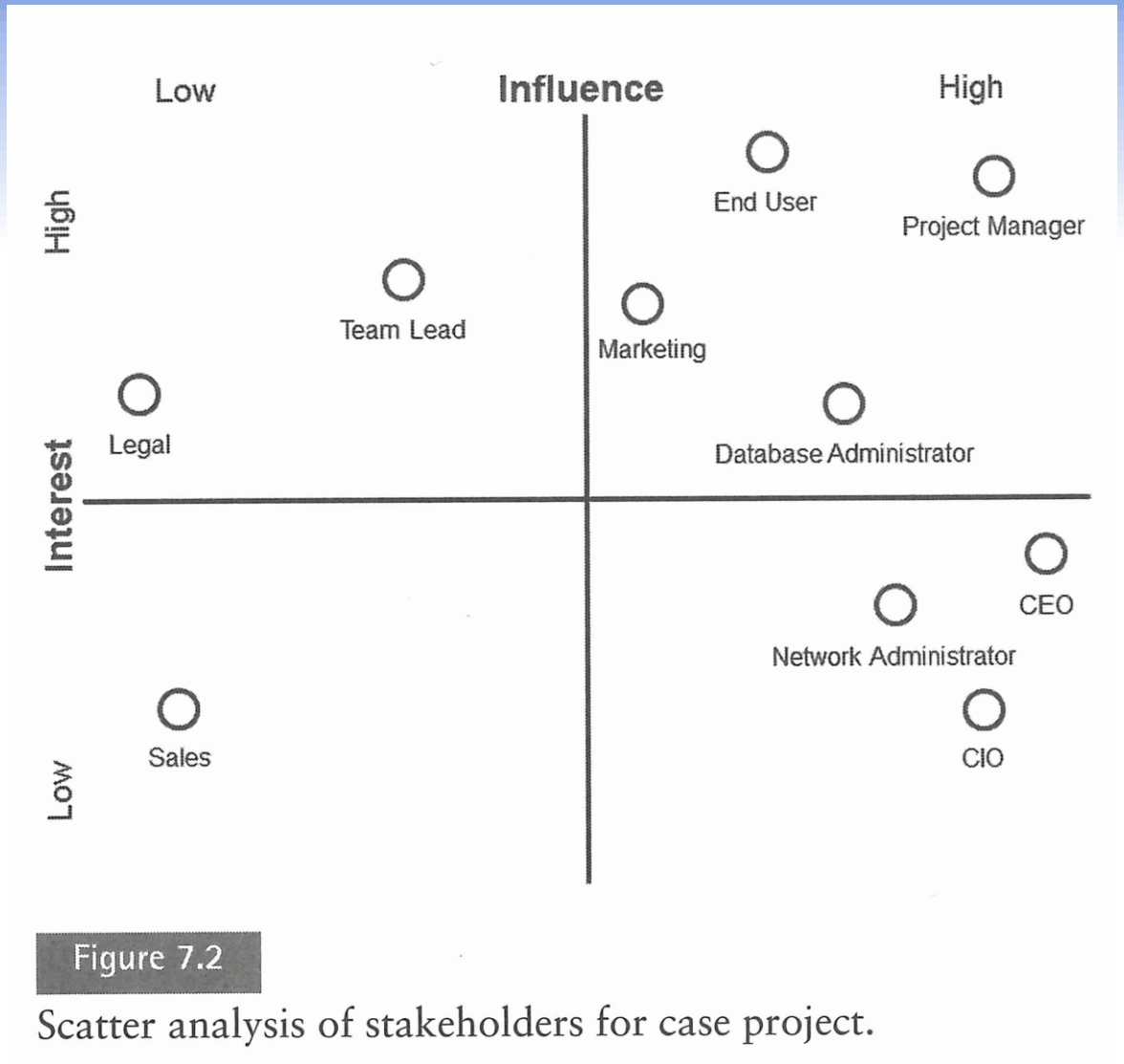


Figure 7.2

Scatter analysis of stakeholders for case project.

Point to Remember….

The earlier security is considered, the more likely it is to be implemented well. While the baseline of security considerations will be confidentiality, integrity, and availability, you can get more granular with the list you present to each of the stakeholders by including items such as the following suggestions:

1. Data privacy
2. Strict authentication and access control
3. Uptime/reliability
4. Failing safely
5. Nonrepudiation

Gathering Requirements

- A **requirement** is an outcome for the proposed system, something that it must perform or a quality it must have.

Two types of requirements:

1. A **functional requirement** is something that the system must do; it is an outcome that the system must produce as part of its useful operation.
2. A **nonfunctional requirement** is a quality or constraint for the system; this is something that must be upheld as the system operates.

Functional and Nonfunctional Security

Asking and answering the following questions will create a well-written requirement:

1. **Why should this be part of the system?**
2. **What are the constraints on this requirement?**
3. **What are the dependencies for this requirement?**
4. **Who are the stakeholders for this requirement?**

Example requirement types

- Functional

  7. “Payment options should be presented for automatic monthly renewal.

  Something the system must do

- Nonfunctional

  25. “Users can search for questions or other user profiles with a single search interface.”

  Constraint that can’t be more than one search interface

Gathering Requirements

- A **security requirement** is an associated protection that must be placed on some part of the system as a contingency to normal operation or a guarantee of some constraint that would otherwise violate the conditions of safe operation.

Functional and Nonfunctional Security

- Security at the requirements level is mainly a consideration of all outcomes for a functional requirement **and an assessment of what happens if the constraint fails for a nonfunctional requirement**.
- Requirements are written in the language of business, but that does not mean there is no room for including technical components, especially in considering constraints.

Security Requirement

- **Fail case:** This is what will happen if the requirement is not fulfilled during operation.
- **Consequence of failure:** This is the result of the fail case. When the fail case is hit for the requirement, this is where the potential outcome should be documented.
- **Associated risks:** The associated risks include sensitive information that could be compromised or revealed, domino effects to the failure of dependent requirements, or violation of laws or system specifications.

Security Requirement

- What are the exceptions to the normal case for this requirement?
- What sensitive information is included in this requirement?
- What are the consequences if the conditions of this requirement are violated?
- What happens if this requirement is intentionally violated?

Note the answer to each of these is in the context of the **specific requirement**

Security Requirement Analysis

Requirement: Users will vote only once per question.

Fail case:

A user is allowed to vote twice on the same question.

Consequence of failure:

The vote tally will be incorrect; confidence in the system will be lost.

Associated risks:

Violation of product purpose; users may stop using product.

Security Requirement Analysis

Requirement: Network communications between the client and the server will follow a stateful pattern to allow traffic anomalies to be detected.

Fail case:

The pattern detection control ceases to operate; network traffic is detected outside of the expected pattern.

Consequence of failure:

Unexpected traffic may be allowed on the network; the system may be stuck waiting for traffic to change the state of the communication.

Associated risks:

Malicious traffic may enter the network, compromising the system or other network resources.

Security Requirement Analysis

Requirement: Registered users can vote A or B on a question they are allowed to answer.

Fail case:

A user leaves the page without voting, a user selects a nonstandard option, or a user selects no option. The system may not successfully complete the processing of the vote.

Consequence of failure:

The user may have to vote again or vote at a later time. The data storage mechanism could be at risk if the result does not conform to what is expected.

Associated risks:

The requirement utilizes user credentials, viewing permissions, and access to the data storage mechanism.

Establishing Scope

- **Product scope** is the collection of functional and nonfunctional requirements that will be included in the final system.
- **Project scope** refers to the work that is to be completed and is more concerned with how the project itself will be governed, such as personnel, timelines, and so on.

Establishing Scope

- **Validation testing**: asserting that the needs of the system and the needs of the stakeholders are being met with the requirements gathered.
- **Validation**: is the process of making sure the right system is being built.
- **Tradeoff-analysis**: is where both competing needs are analyzed and the best outcome for the project is decided.

Summary

- Security that is constructed into a system as it is being designed is the most likely to succeed.
- The best way to handle security is to do so systemically from the project's inception.
- Security needs to be woven throughout the SDLC.
- Establish a strict scope for the project.

Groups

- Next several assignments will be group based
  - 3-4 preferred (no more than 4)

In class security requirement analysis

Break into groups and do security requirement analysis of the following requirements:

18. Registered users can search for a question they are allowed to answer.

19. Registered users can vote A or B on a question they are allowed to answer.

Reminder of parts

Fail case:

Consequence of failure:

Associated risks:

In class security requirement analysis

Break into groups and do security requirement analysis of the following requirements:

24. Users can answer a question when they are allowed to access it by the user who posted the question.

26. Search results will already be filtered by the user's permissions when they are returned.

Reminder of parts

Fail case:

Consequence of failure:

Associated risks:

In class exercises revisited

Bob's Pizza Shack

In groups consider this scenario and

1) Identify 4 functional requirements that would have associated security requirements

2) Then, do the associated security requirement analysis (fail case, consequence of failure, associated risks)

- Bob is a small business owner of Bob's Pizza Shack and wants to create a website to allow online credit card delivery orders

Alice's Online Bank

In groups consider this scenario and

1) Identify 2-4 functional requirements that would have associated security requirements for each bullet below

2) Then, do the associated security requirement analysis (fail case, consequence of failure, associated risks)

- Alice opens Alice's Online Bank (AOB)
  - Internal use bank employees
  - Interoperates with another bank
  - Interacts with customers
    - Web
    - Mobile app

Location based social media app

In groups consider this scenario and

1) Identify 2-4 functional requirements that would have associated security requirements for each bullet below

2) Then, do the associated security requirement analysis (fail case, consequence of failure, associated risks)

- Open source group wants to create a mobile app to allow groups to communicate/find each other in public demonstrations/protests
  - Communication internet, as well as, P2P (WiFi/Bluetooth) in case internet cut off – so if person you want to contact is on other side of crowd and no internet as long as P2P network can be established with app can reach somebody outside your immediate vicinity via the P2P network
  - Should be able to communicate securely messages and images to people you identify within your group (group as whole or direct)
  - Should be able to share GPS location with people in group (group as whole or direct)