

Attack Surface Identification and Reduction Model Applied in Scrum

Dr. George O. M. Yee

Computer Research Lab, Aptusinnova Inc.

Dept. of Systems and Computer Engineering, Carleton University

Ottawa, Canada

gmyee@sce.carleton.ca | george@aptusinnova.com

Content

- Objective
- Approach
- Conclusions and Future Research

Objective

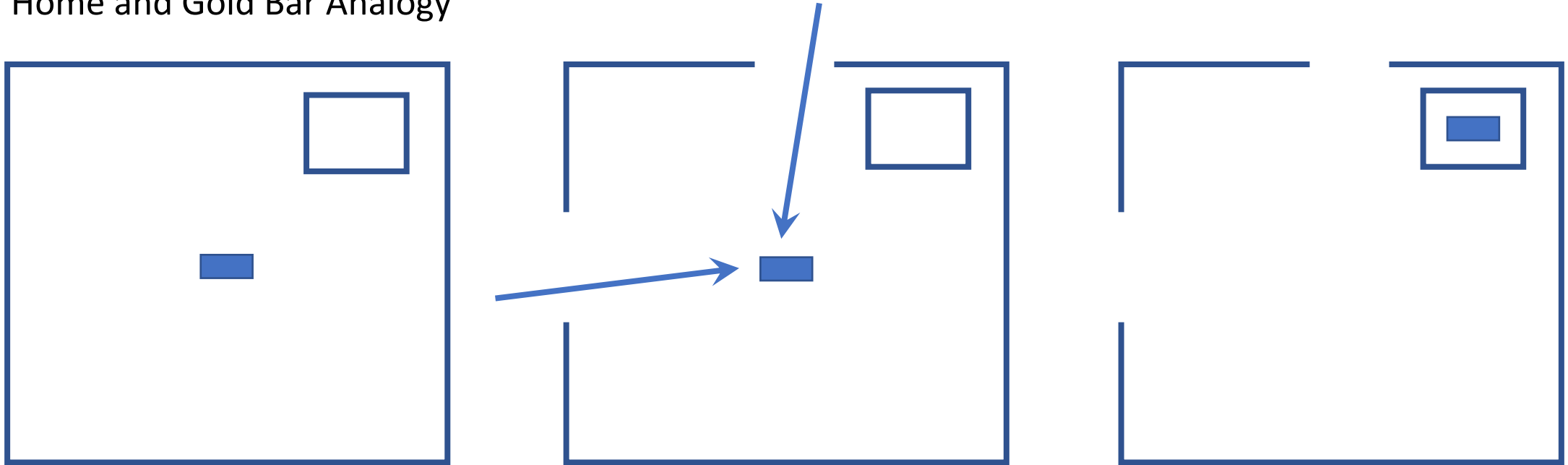
- Define a simple hand drawn visual model and show how it can be used in Scrum to identify and reduce the attack surface of the software system under development.
- The model applies to software systems that contain sensitive data.

Approach – Attack Surface

- Attack surface: set of ways in which an attacker can enter the system and potentially damage the system
- We are interested in protecting sensitive data (SD) stored in the system.
- Analogous to physical security – the protection of a gold bar

Approach – Attack Surface

Home and Gold Bar Analogy



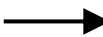



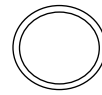

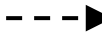

- Attack surface for home: set of all locations in the home that contain unprotected valuables that are reachable by thieves
- Attack surface for this work: set of all locations in the software system that contain unprotected SD that is reachable by attackers

Approach - Method

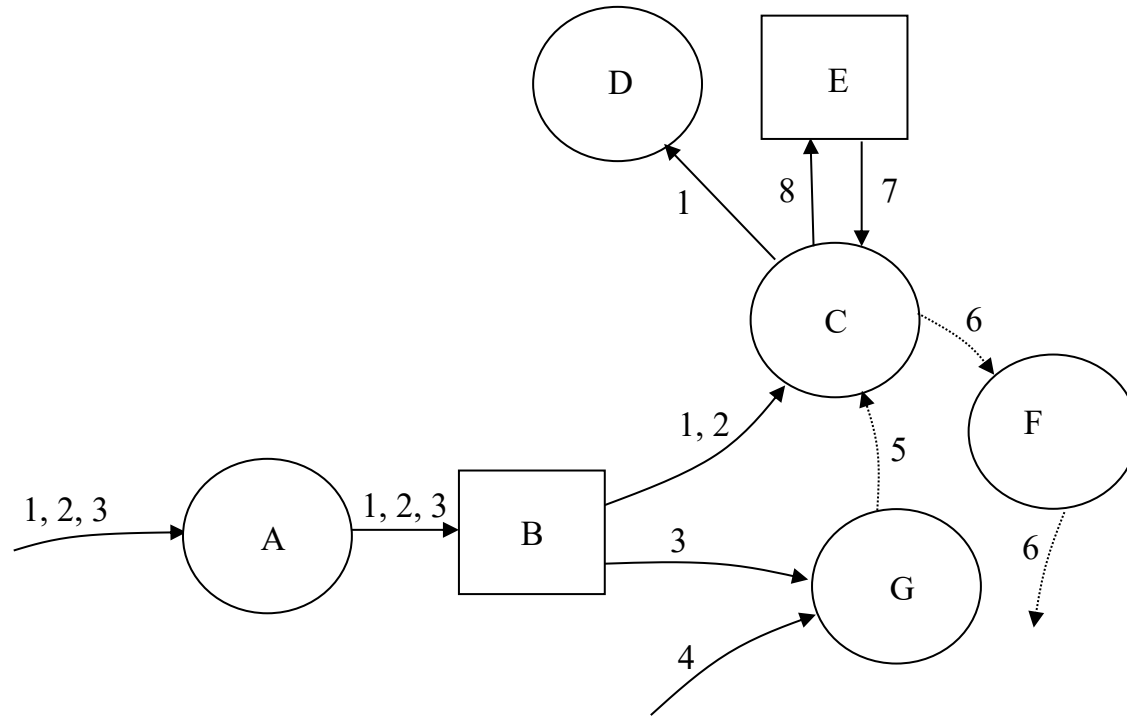
1. Draw the software system using the defined visual model, determining the locations in the modeled system that contain SD.
2. Evaluate the model for the locations of attacker-reachable unprotected SD, resulting in identifying the attack surface .
3. Reduce the attack surface by modifying the model to
 - a) Change the SD's location,
 - b) Obfuscate the SD,
 - c) Deny access to the SD, or
 - d) Eliminate the need to store the SD.
4. Transfer changes to the actual system.

Approach – Visual Model (Partial)

Location Elements	Description
	Data flow elements are labeled with numbers; all other elements are labeled with letters.
SD Use Circle 	Identifies where SD is used.
SD Data Store 	Identifies where SD is stored.
SD Data Flow 	Identifies the movement of SD.
Non-SD Data Flow 	Identifies the movement of non-SD.
Description Element	Description
Legend	Descriptions of above labeled elements.

Attack Surface Reduction Elements	Description
Merged SD Use Circle 	Identifies where one or more use circles have been merged, with corresponding deletion of original circles as needed.
Obfuscated SD Data Store 	Identifies where SD in a data store has been obfuscated (e.g., encrypted, anonymized).
Obfuscated SD Data Flow 	Identifies the movement of obfuscated (e.g., encrypted, anonymized) SD from one location to another.
Reduced Accessibility to SD Data Flow 	Encloses use circles and data stores that execute on the same computing platform, thus reducing the attack surface for traversing data flow.
Description Element	Description
Legend	Descriptions of above labeled elements.

Approach – Internet Seller



Legend:

A: input of customer data

B: customer data store (SD)

C: verify if in stock and ship

D: produce shipping label

E: in stock items data store (SD)

F: output order progress

G: charge payment

1: name & address (SD)

2: item for purchase (SD)

3: payment info (SD)

4: company account info (SD)

5: payment success indicator

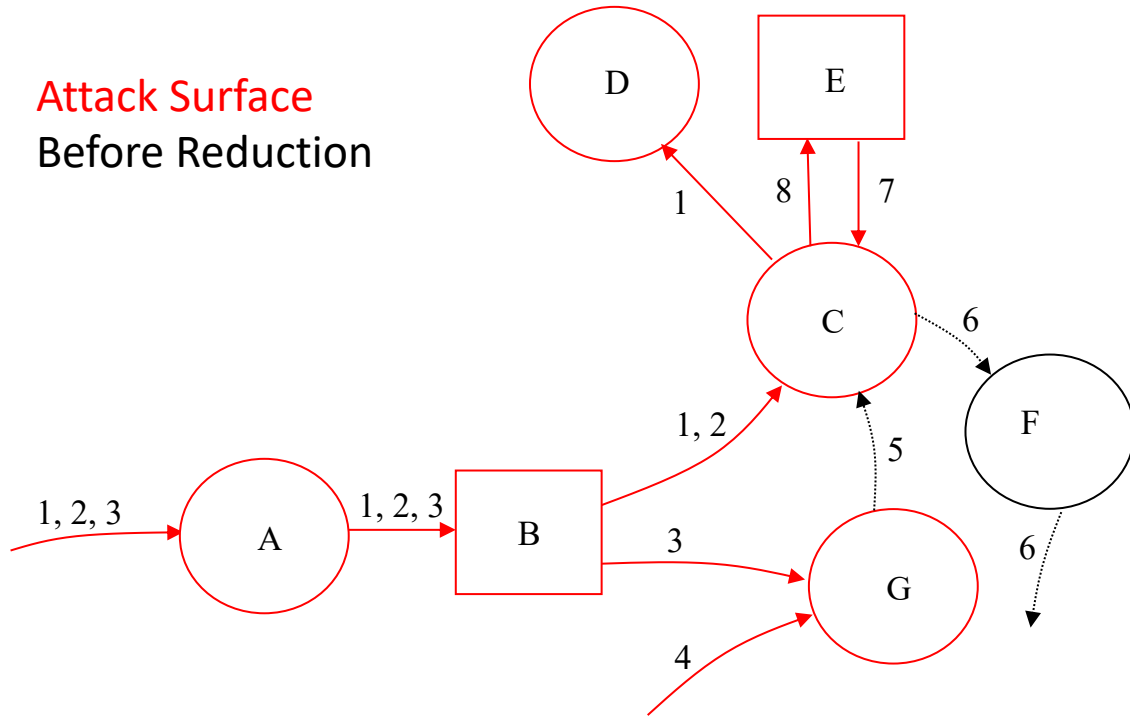
6: order progress indicator

7: in stock data (SD)

8: in stock update (SD)

Approach – Internet Seller

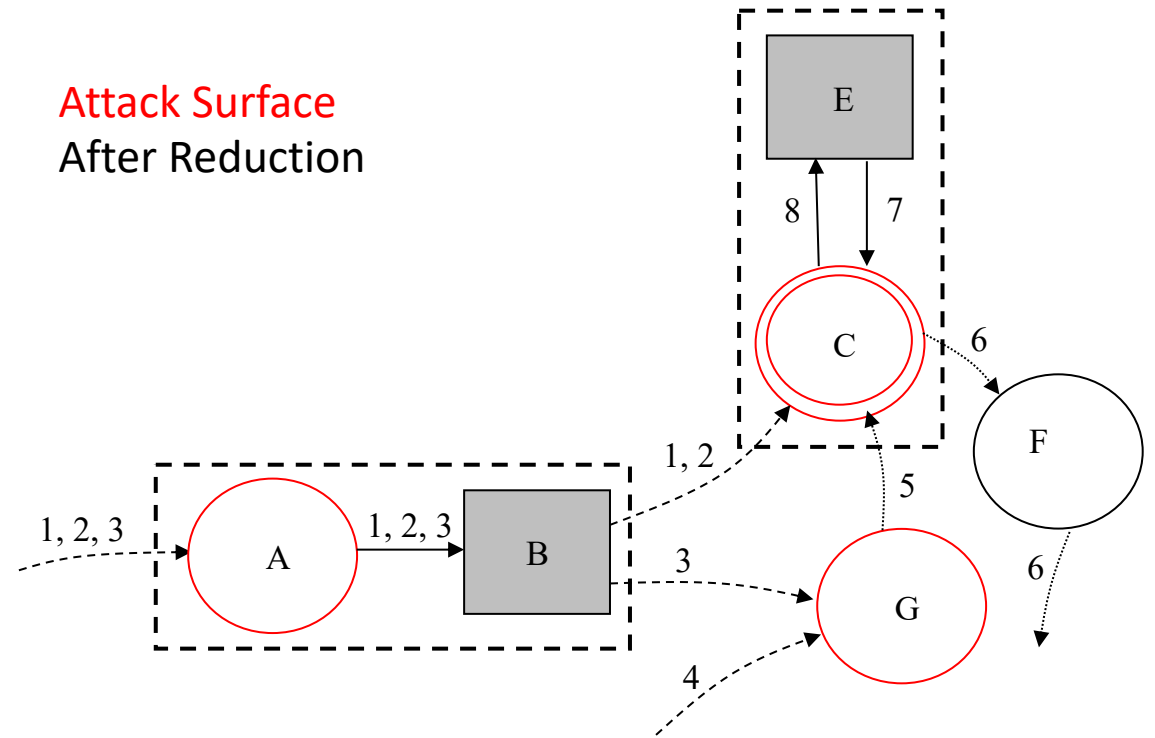
Attack Surface
Before Reduction



Legend:

- | | |
|-----------------------------------|------------------------------|
| A: input of customer data | 1: name & address (SD) |
| B: customer data store (SD) | 2: item for purchase (SD) |
| C: verify if in stock and ship | 3: payment info (SD) |
| D: produce shipping label | 4: company account info (SD) |
| E: in stock items data store (SD) | 5: payment success indicator |
| F: output order progress | 6: order progress indicator |
| G: charge payment | 7: in stock data (SD) |
| | 8: in stock update (SD) |

Attack Surface
After Reduction



Legend:

- | | |
|---|------------------------------|
| A: input of customer data | 1: name & address (SD) |
| B: customer data store (SD) | 2: item for purchase (SD) |
| C: verify if in stock, produce shipping label, and ship | 3: payment info (SD) |
| E: in stock items data store (SD) | 4: company account info (SD) |
| F: output order progress | 5: payment success indicator |
| G: charge payment | 6: order progress indicator |
| | 7: in stock data (SD) |
| | 8: in stock update (SD) |

Approach - Scrum

- Scrum Development Team
 - One or more members knowledgeable in security
 - Attack surface reduction done in first sprint with help of Scrum Master if needed
 - Reduced attack surface model used as reference for later sprints
 - Discussion on transferring changes to actual development takes place in later sprints
 - A new reduced attack surface model can be obtained at a later sprint to accommodate new changes in requirements
- Visual model is a form of agile modeling
 - Fulfills its purpose and no more, understandable, sufficiently accurate, sufficiently consistent, sufficiently detailed, provides positive value, as simple as possible

Approach – Strengths and Weaknesses

- Strengths
 - A design for security approach
 - Can only reduce the attack surface, never increase it (if at least 1 attack surface reduction element is used)
 - Straightforward and clear
- Weaknesses
 - Manual approach
 - Requires security knowledge
 - Incurs additional overhead

Conclusions and Future Research

- Approach identifies and reduces the attack surface in a software system using a straightforward visual model; can be used in Scrum
- Attack surface reductions considered for actual system
- Approach produces a more secure system, not a secure one
- Future Research includes:
 - Resolving the approach's weaknesses, e.g. automation
 - Trials with Scrum developers to validate and refine the approach
 - Investigate new ways to identify and reduce the attack surface

Thank you for your attention.