Introducing Abuse Frames for Analysing Security Requirements

Requirements Engineering Conference, 2003. Proceedings. 11th IEEE International, 8-12 Sept. 2003 Pages: 371 - 372

By
Luncheng Lin
Basher Nuseibeh
Darrel Ince
Michael Jackson
Jonathan Moffett

Presented by Brendan Cervin

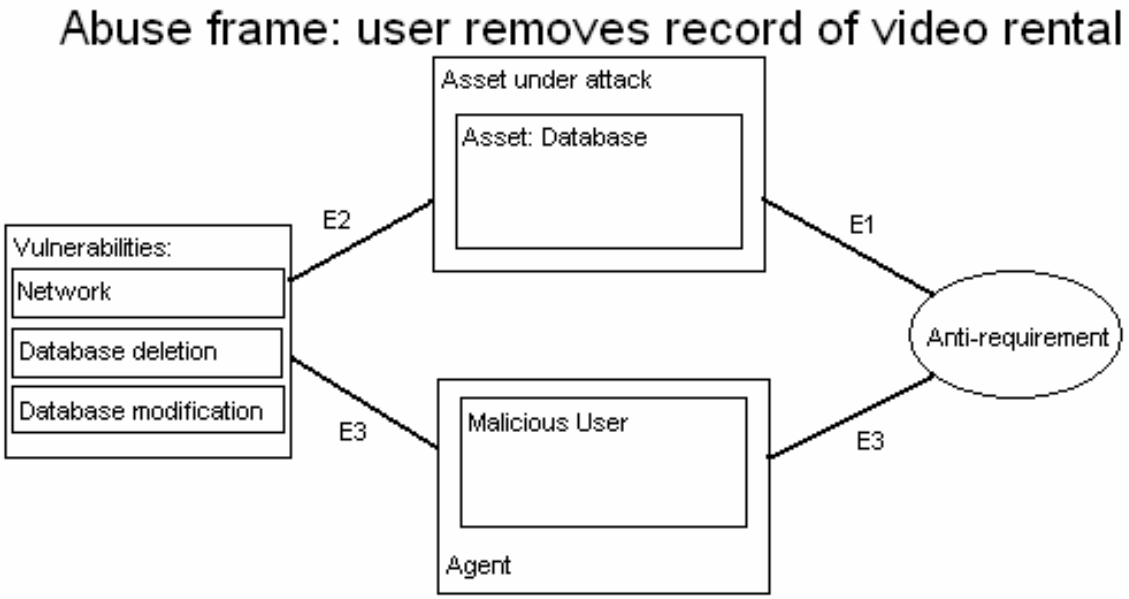
Summary

- Identifies the need to effectively describe security requirements early in development.
- Describes extending Jackson’s Problem Frames into Abuse Frames with anti-requirements to analyse and determine security vulnerabilities.
- These create a manageable scope which can be used to create security requirements.

Appreciative: abuse frames simplify security requirements

- Abuse frames focus on malicious users rather than user mistakes
- Anti-requirements are created to subvert existing requirements
- Given the requirement:
  - Online video store tracks user rentals
- The anti-requirement can be:
  - User removes record of a video they rented

Example abuse frame



- Here we see three domains
  - Asset, Vulnerabilities, Agent
- The anti-requirement describes the behaviour between the agent and the asset

- Vulnerabilities identify the parts of the system that are involved in the abuse
- This abuse frame can be used to create security requirements for the vulnerable parts of the system

Critical: poorly explained

- In general the paper is brief and fails to explain diagrams completely
  - The term E2 and phenomenon are not described
  - E3 is not described for the abuse frame diagram, it is described later in reference to a problem frame diagram
  - There is no summary or conclusion
  - The problem frame is described after the abuse frame which creates confusion

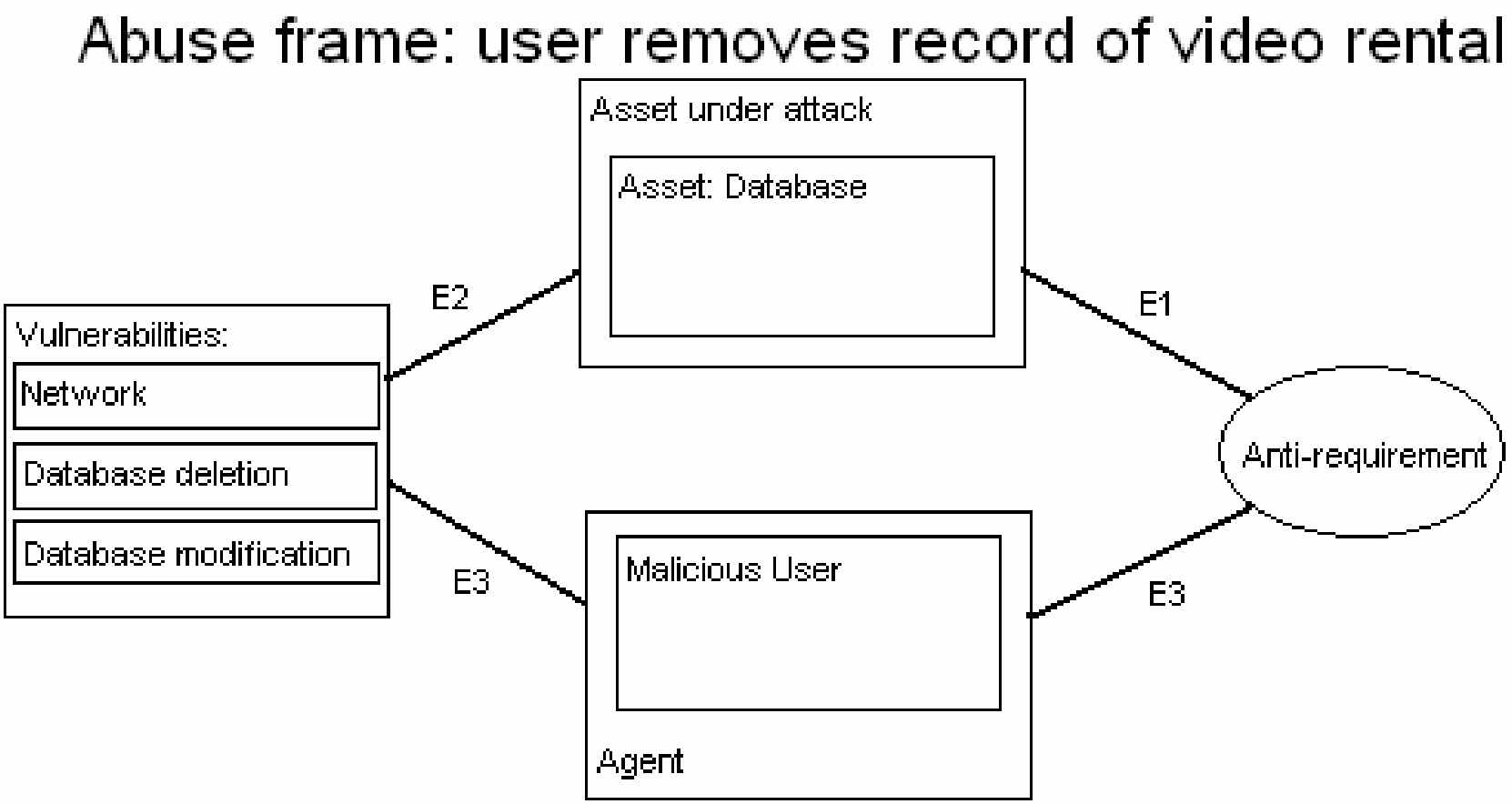
Critical: diagram layout

- While the paper presented only one example the diagram seems to be unnecessarily complicated and constrained:
  - The use of boxes within boxes is excessive
  - It is difficult to write the full anti-requirement within the diagram, the circle is not efficiently using space
  - Having the E3 line twice is confusing
  - The line names are not descriptive

Questions?

- Even though abuse frames are designed to represent malicious users, could it represent malicious programs or user mistakes?
- Would you try abuse frames to create security requirements, why?

Abuse frame


Abuse frame: user removes record of video rental
Asset under attack
Asset: Database
E2
E1
Vulnerabilities:
Network
Database deletion
Database modification
Anti-requirement
E3
E3
Malicious User
Agent