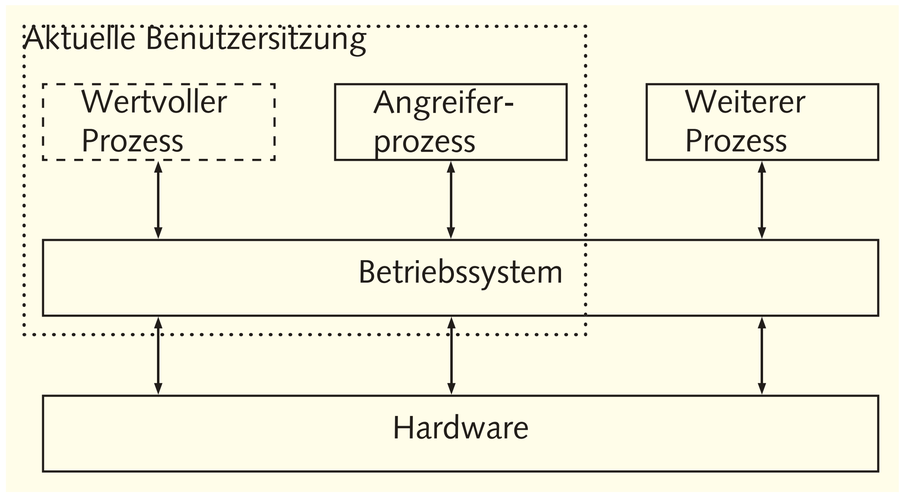
Eine CC-basierte Ontologie zur Analyse architektureller Schwachstellen

Hanno Langweg, Gjøvik University College

Motivation

* Aufspüren und Bewerten von Schwachstellen in der implementierungsunabhängigen Architektur eines Softwaresystems. (Architektur hier definiert über beteiligte Komponenten und ihre Beziehungen.)
* Analyse der Widerstandsfähigkeit architektureller Eigenschaften, z.B. Präsenz von Tokens und Akteuren, Kooperation von Angreifern, Zeitfenster, Angriffsflächen, Schnittstellen



* Gleiche Maschine, gleiche Betriebssystemsitzung
* Gleiches oder ähnliches Benutzerkonto
* Gleiche oder ähnliche Prozessprivilegien
* Angriff erfolgt nicht von außen, sondern mittels auf System vorhandenem Prozess

Messbare Sicherheit

* Wie gut wehren bestimmte Mechanismen durch ihr Vorhandensein/durch Wahl geeigneter Parameter einen Angriff ab?
* Wie wirken sich Änderungen in der Architektur auf die Widerstandsfähigkeit aus?
* Welche neuen Architekturen reduzieren die Auswirkungen eines Angriffs?

Ontologie und Modell

* Während diese Ontologie das Vokabular repräsentiert, existieren dazu weitere UML-Diagramme, welche das Verhalten der beteiligten Komponenten beschreiben.

Daten

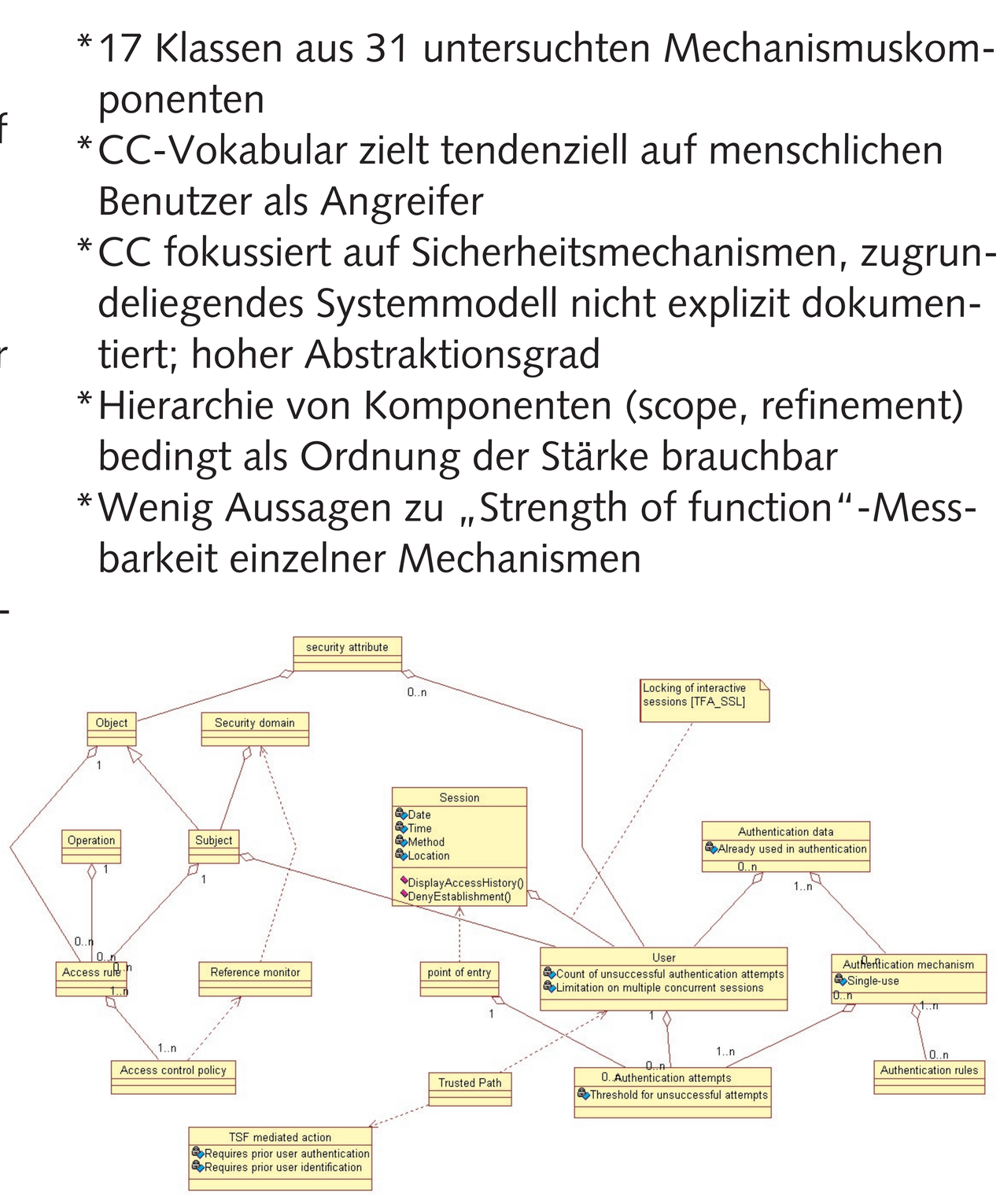
* Common Criteria (CC) als anerkanntes und verbreitetes internationales Kriterienwerk
* Zunächst Einschränkung auf Familien FDP (User data protection), FIA (Identification and authentication), FPT (Protection of the TSF), FTA (TOE access), FTP (Trusted path/channels)
* Geplante ergänzende Auswertung: FAU (Security audit) [und ausgelassene Teilkomponenten...]

Methode

* Betrachtung der einzelnen beschriebenen Sicherheitsmechanismen ausgewählter Klassen
* Sammlung der am Mechanismus und seinem Verhalten beteiligten Objekte und ihrer Beziehungen zueinander
* Notation in UML

Resultat

* 17 Klassen aus 31 untersuchten Mechanismuskomponenten
* CC-Vokabular zielt tendenziell auf menschlichen Benutzer als Angreifer
* CC fokussiert auf Sicherheitsmechanismen, zugrundeliegendes Systemmodell nicht explizit dokumentiert; hoher Abstraktionsgrad
* Hierarchie von Komponenten (scope, refinement) bedingt als Ordnung der Stärke brauchbar
* Wenig Aussagen zu „Strength of function“-Messbarkeit einzelner Mechanismen



Hanno Langweg, hanno.langweg@hig.no, Norwegian Information Security Laboratory – NISlab, Department of Computer Science and Media Technology, Gjøvik University College, Norwegen