# 5. Workshop Entwicklung zuverlässiger Software-Systeme 2016

## 09. Juni 2016, TechBase Regensburg

Hubert B. Keller, KIT Karlsruhe, hubert.keller@kit.edu

Peter Dencker, Sprecher FG Ada, HS Karlsruhe, dencker@web.de

Der 5. Workshop Entwicklung zuverlässiger Software-Systeme 2016 ist für den 09. Juni 2016 in Regensburg geplant gewesen. Aufgrund terminlicher Überschneidungen musste er leider abgesagt werden. Themen waren speziell die Zuverlässigkeit und Sicherheit sowie sicherheitskritische Aspekte eingebetteter Systeme. Außerdem stand die Entwicklung zuverlässiger Software-Systeme mit allen Themenbereichen wie Zuverlässigkeit mobiler Systeme, IT Sicherheit, Nachweis der Erfüllung der Sicherheitsanforderungen, Analyse von Sicherheitsrisiken, Ada für sichere Systeme etc. im Fokus. Die Gremiensitzungen wie die Mitgliederversammlung der Fachgruppe Ada - Zuverlässige Softwaresysteme der Gesellschaft für Informatik in Kooperation mit dem Fachausschuss 5.11 Embedded Software der Gesellschaft für Automatisierungstechnik im VDI/VDE sowie die Mitgliederversammlung des Förderverein Ada Deutschland e.V. fanden statt. Jens Mehrfeld vom BSIS hatte sich bereit erklärt, den Hauptvortrag, mit dem Thema „Industrial IT Security – Ein Überblick der Lage aus Sicht des BSI“ zu halten. Ein wichtiges Thema, das auf jeden Fall noch einmal angesprochen werden wird. Drei eingereichte Beiträge wurden von den Gutachtern für die Software-Technik-Trends vorgeschlagen.

Der erste Beitrag „Das Nürnberger Anti-Phishing-Device: Beweisbare Korrektheit durch Einfachheit“ von Peter Trommler realisiert auf einem Arduino Micro Pro Mikrocontroller ein sicheres Device für das Online Banking. Das Nürnberger Anti-Phishing Device besitzt die minimal notwendige Funktionalität für die Erstellung digital unterschriebener Transaktionszusammenfassungen und konnte daher formal verifiziert werden. Wichtig dabei war zu zeigen, dass kein Buffer Overflow, die Eingabedaten sind größer als der reservierte Speicher, oder eine incomplete Mediation, das Format der Eingabedaten stimmt nicht oder diese selbst werden nicht adäquat behandelt, auftreten kann. Gerade diese Schwachstellen sind Ursache für Sicherheitslücken in jedweder Art von Software.

Der zweite Beitrag “Systematic Identification of Security Goals and Threats in Risk Assessment“ von Daniel Angermeier, Alexander Nieding und Jörn Eichler geht von einem sich in Entwicklung befindlichen System (SUD – System Under Development) aus. Das Vorgehen besteht aus vier Aktivitäten, “Document SUD”, hier wird das Modell des SUD definiert, “Determine Protection Needs” zur Festlegung der Sicherheitsziele, “Analyze Threats” nimmt die Position eines Angreifers ein und versucht Angriffspunkte zu finden und “Analyze Risks” bewertet die Bedrohungen und den möglichen Schaden. Dieser Beitrag stellt also die Sicht auf die Sicherheit von außen dar und vertritt den eher klassischen Ansatz im Gegensatz zum ersten Beitrag.

Der dritte Beitrag „Integrierte Entwicklung zuverlässiger Software“ von Oliver Schneider und Hubert B. Keller integriert die klassische manuelle Programmierung mit der Modell-getriebenen Softwareentwicklung und setzt einen neuen Ansatz in der Tool-basierten Prozessrückkopplung zur Verbesserung um. Modell-getriebene Softwareentwicklung soll nicht in der Programmierung enden und manuelle Programmierung ist mit Modellen zu verbinden. Erkannte Fehler sind direkt in Form statischer Analysen in den Compiler zu integrieren. Hintergrund dieses Konzepts sind die Entwicklungen beim RUST Compiler der Mozilla Foundation. Dort wurde erkannt, dass mit klassischen Ansätzen auf Basis von Sprachen wie C, C++ oder Java keine zuverlässige Software entwickelt werden kann. Auf Basis von Konzepten der Sprache Ada wurde die neue Sprache RUST definiert und darüber hinaus ein Compiler Konzept mit Plug-In Schnittstellen zur Integration von statischen Analyseregeln entwickelt.

Sicherheit im Sinne von Safety und Security kann nur integriert realisiert werden. Dazu zählen die Sicht von außen, die Vermeidung von Schwachstellen und die Fehlervermeidung in der eigentlichen Entwicklung. Die drei Beiträge weisen daher in Summe in die richtige Richtung.