

Aufgabe 3:

Implementierung sicherer Software

Wenn bereits die Konzeption sicherer Software (siehe Aufgabe 2) uns vor größere Probleme stellt, wie sieht es dann mit der Umsetzung aus?

- Welche Ansätze gibt es, um aus einem sicheren Softwarekonzept eine sicher Software zu machen? Was verstehen wir unter *sicherer Software*? Was bedeutet *Korrektheit* in Bezug auf die Funktionalität einer Software?
- Was müssen Sie bei der Entwicklung des Codes bedenken? Gibt es allgemeine Regeln für das Schreiben sicheren Quellcodes?
- Gibt es formale (oder weniger formale) Methoden, um eine Software bzw. eine Software-Komponente hinsichtlich ihrer Sicherheit und Korrektheit zu überprüfen?
- Wie wird Software für eine Zertifizierung, beispielsweise nach den *Common Criteria*, geprüft?