36 Entwicklung sicherer Software-Systeme

Design of Safe Software Systems

Semester	Wahlpflichtbereich
Dauer (Semester)	einsemestrig
Credit Points	5
Pflicht/ Wahlpflicht	Wahlpflicht
Häufigkeit des Angebotes/ Verwendbarkeit	Jedes Semester nach Bedarf der Hochschulen des VFH-Verbundes.
Modulverantwortliche(r)	Dr.-Ing. Martin Schafföner, Technische Hochschule Brandenburg
Teilnahmevoraussetzungen	keine
Lernergebnisse	Die Studierenden können die für die Entwicklung sicherer Softwaresysteme notwendigen Tätigkeiten im gesamten Softwarelebenszyklus sinnvoll auswählen und durchführen. Sie kennen relevante Best Practices (z.B. Microsofts Secure Development Lifecycle, Open Web Application Security Project), Normen (z.B. ISO 27000-Reihe) und regulatorische Werke (z.B. Medizinproduktegesetz). Studierende können Anforderungen bzgl. der Softwaresicherheit mittels Schutzbedarfs- und Risikoanalysen erheben und dokumentieren. Sie können Entwurfsentscheidungen zur Umsetzung der Anforderungen bewerten und auswählen, z.B. durch Anwendung bewährter Sicherheits-Entwurfs- und Architekturmuster, insbsd. für mobile und verteilte Systeme sowie für mandantenfähige Cloud-Anwendungen. Studierende kennen typische Fehlerquellen bei der Implementierung sicherer Software. Sie können mittels Aspektorientierter Programmierung eine sinnvolle Trennung fachlicher und sicherheitsspezifischer Aufgaben, z.B. Authentisierung und Autorisierung, sicheres Logging oder Auditierung, umsetzen. Studierende können besondere Testmethoden und Qualitätssicherungsverfahren zur Überprüfung von Sicherheitsaspekten auf allen Ebenen der Testhierarchie anwenden. Sie können relevante Best Practices für den Betrieb sicherer Software benennen, insbsd. bzgl. Virtualisierung von Hardware, Netzwerksicherheit und Patchmanagement.
Prüfungsvorleistung	Einsendeaufgabe, Präsenzteilnahme

Medien-/ Lernform	Multimedial aufbereitetes Online-Studienmodul zum Selbststudium mit zeitlich parallel laufender Online-Betreuung (E-Mail, Foren, Chat, Webkonferenzen, Einsendeaufgaben u. a.) sowie Präsenzphasen
Arbeitsaufwand	Präsenzteilnahme: ca. 3 h Prüfung: 120 Minuten Selbststudium: 105,5 h Betreutes Lernen: 32,5 h Vorbereitung PVL: 12 h
Präsenzart	erfordert physische Anwesenheit
Präsenzinhalte	Inhaltliche Klärung, Vorstellung der Lösungskonzepte von ausgewählten Aufgaben.
Prüfungsform	Klausur (120 min.)
Literatur	Sachar Paulus: „Basiswissen Sichere Software“, dpunkt.verlag, 2011 Fred Long: „Java Coding Guidelines“, Software Engineering Institute, 2013 Michael Howard: „Sichere Software programmieren“, Microsoft Press, 2002 Bolt William: „Engineering Secure Software“, 2016 Microsoft Security Development Lifecycle (SDL), 2012, https://msdn.microsoft.com/en-us/library/windows/desktop/cc307748.aspx Adam Shostack: „Threat Modeling: Designing for security“, Wiley, 2014 Ross Anderson: „Security Engineering: A Guide to Building Dependable Distributed Systems“, Wiley, 2008 Claudia Eckert: „IT-Sicherheit. Konzepte - Verfahren – Protokolle“, Oldenbourg, 2009, http://www.worldcat.org/oclc/463676855
Vertiefungsrichtung	Informatik und Software-Entwicklung, IT-Sicherheit
weitere Hinweise	Dieses Modul wird auf Deutsch angeboten

Studieninhalte
1. Einbettung und Ziele der Entwicklung sicherer Softwaresysteme 2. Überblick: Secure Software Development Lifecyle 3. Bedrohungsanalyse 4. Sicherheits-Antimuster, Analyse von Bestandscode 5. Architektur- und Entwurfsprinzipien 6. Best Practices für sichere Softwareentwicklung mit ausgewählten Programmiersprachen 7. Identitäts- und Zugriffsverwaltung 8. Aspect-Oriented Programming am Beispiel: Authentisierung/Autorisierung, Audit-Logs 9. Testen von Sicherheitsanforderungen 10. Sicherheits-Metriken für kontinuierliches Feedback im Entwicklungsprozess

11. Nationale und internationale Normen und andere Regelungswerke
12. Betriebsaspekte für sichere Software: Virtualisierung, Patch-Management