Modulhandbuch IT-Sicherheit Master of Science

Version M_ITS26.0_S

Letzte Änderung: 2026-07-11 15:25:33

Inhaltsverzeichnis

Module

◈ MM005 – Funktionale Programmierung

Verantwortliche:	Torben Tietgen
Moduldauer:	6 Monate
Unterrichtssprache:	deutsch

Bestandteile:

Teilleistung:	TM028 – Funktionale Programmierung
Lernform:	Vorlesung
Prüfungsform:	Klausur / Mündliche Prüfung
Prüfungsdauer:	45 Min.
ECTS:	2.0
Benotung:	Drittelnoten
Turnus:	jährlich
Dauer (pro Termin):	2 Semesterwochenstunden
Termine im Semester:	12 Termine
Zeit in Veranstaltungen:	15 Stunden
Sonstiger Arbeitsaufwand während Vorlesungszeit:	25 Stunden
Aufwand während Semesterferien:	20 Stunden
Flexibel einteilbarer Aufwand:	0 Stunden
Gesamtaufwand:	60 Stunden
Lehrende:	Frank Huch

Teilleistung:	TM029 – Übg. Funktionale Programmierung
Lernform:	Übung
Prüfungsform:	Abnahme
Prüfungsdauer:	10 Min.
ECTS:	3.0
Benotung:	Bestanden/nicht Bestanden
Turnus:	jährlich
Dauer (pro Termin):	2 Semesterwochenstunden
Termine im Semester:	12 Termine
Zeit in Veranstaltungen:	15 Stunden
Sonstiger Arbeitsaufwand während Vorlesungszeit:	75 Stunden
Aufwand während Semesterferien:	0 Stunden
Flexibel einteilbarer Aufwand:	0 Stunden
Gesamtaufwand:	90 Stunden
Lehrende:	Torben Tietgen

Studieninhalte:

• Syntax von Haskell
• Produkt- und Summen-Datentypen
• Rekursive Datenstrukturen
• Polymorphismus
• Funktionen höherer Ordnung
• Hindley-Milner Typsystem, Typüberprüfung und Typinferenz
• Korrektheitsargumentationen
• Funktionale Datenstrukturen
• Bedarfsauswertung und unendliche Strukturen
• Typklassen
• Funktoren und Monaden
• Beispielanwendungen
Zusätzlich werden in der Übung praxisrelevante Aspekte der Anwendungsentwicklung mit der Programmiersprache Haskell behandelt, die nicht Bestandteil der Vorlesung sind. Die Bearbeitung der Übungsaufgaben folgt parallel zum Stoff der Vorlesung in Zweiergruppen.

Lernergebnisse:

Die Studierenden ...
• beherrschen fortgeschrittene Techniken der funktionalen Programmierung am Beispiel der Sprache Haskell.
• können mit Funktionen höherer Ordnung, mit polymorphen Datentypen und Typklassen, mit Funktoren, Monaden, Monoiden und weiteren mathematischen Strukturen umgehen und in neuen Kontexten anwenden.
• beherrschen Fähigkeiten in der Modellbildung und Abstraktion und können sie anwenden.
• kennen die Bezüge zwischen Mathematik und funktionaler Programmierung.
• kennen die Vor- und Nachteile des funktionalen Paradigmas für Anwendungen der IT-Sicherheit.

Voraussetzungen und Empfehlungen:

Voraussetzungen sind Kenntnisse und praktische Erfahrungen in höheren Programmiersprachen, insbesondere mit getypten Sprachen. Außerdem werden Kenntnisse über Diskrete Mathematik und algebraische Strukturen erwartet.
Elementares Wissen über Komplexitätstheorie wird ebenfalls vorausgesetzt.
Zudem ist die Installation von Haskell auf dem eigenen Rechner empfehlenswert.

Literatur:

• Miran Lipovaca: Learn You a Haskell for Great Good!: A Beginner's Guide, No Starch Press, 2011, ISBN: 978-1593272838
• Graham Hutton: Programming in Haskell, Cambridge University Press, 2016, ISBN: 978-1316626221
• Richard Bird: Introduction to Functional Programming using Haskell, 2nd Edition Prentice Hall, New Jersey, 1998, ISBN: 978-0134843469

Studiengänge:

- Informatik Master of Science
- IT-Sicherheit Master of Science

◈ MM009 – Workshop Kryptographie

Verantwortliche:	Gerd Beuster
Moduldauer:	6 Monate
Unterrichtssprache:	english

Bestandteile:

Teilleistung:	TM030 – Workshop Kryptographie
Lernform:	Workshop
Prüfungsform:	Abnahme
Prüfungsdauer:	15 Min.
ECTS:	5.0
Benotung:	Drittelnoten
Turnus:	jährlich
Dauer (pro Termin):	4 Semesterwochenstunden
Termine im Semester:	12 Termine
Zeit in Veranstaltungen:	30 Stunden
Sonstiger Arbeitsaufwand während Vorlesungszeit:	120 Stunden
Aufwand während Semesterferien:	0 Stunden
Flexibel einteilbarer Aufwand:	0 Stunden
Gesamtaufwand:	150 Stunden
Lehrende:	Gerd Beuster

Studieninhalte:

- Theorie der Kryptographie
  - Semantische Sicherheit
  - Unbrechbare Verschlüsselung und One Time Pad
  - Diffusion und Konfusion
- Klassische Kryptographie
  - Substitution und Transposition
  - Affine Verschlüsselung
  - Rotormaschinen
- Moderne Kryptographie
  - Stream- und Block-Chiffren
  - DES und GOST
  - AES
- Blockchiffrierung Betriebsarten
  - ECB, CBC, CTR, AES-GCM
- Zufallszahlengeneratoren
  - TRNG und PRNG
  - Voraussetzungen für CSPRNG
- Hashing
  - Hashing-Algorithmen
    - SHA 2
    - SHA 3
  - Nachrichten-Authentifizierung
    - CMAC und HMAC
- Asymmetrische Kryptographie
  - Diffie-Hellman
  - Elliptische Kurven
  - Asymmetrische Verschlüsselung und digitale Signaturen
- Post-Quantum Cryptography
  - CRYSTALS-Kyber
- Hardware-Kryptographie
  - Trusted Computing
  - Smartcards
  - Differential Power Analysis

Lernergebnisse:

Nach Abschluss des Moduls sind die Studierenden in der Lage, ...

- Sicherheitswerkzeuge als wesentlichen Baustein moderner Informations- und Kommunikationssysteme zu verwenden.
- ihr Wissen über alle relevanten Aspekte der Daten-, Netzwerk- und Websicherheit anzuwenden.
- die Anwendung kryptographischer Methoden, insbesondere zur Authentifizierung, Verschlüsselung und Integritätserhaltung zu betrachten.
- die algorithmischen Stärken und Schwächen der kryptographischen Verfahren zu bewerten.
- kryptographische Protokolle zu bewerten und zu implementieren, insbesondere für die Authentifizierung im E-Commerce.
- alle für die Implementierung und Anwendung kryptographischer Methoden relevanten Nebenbedingungen zu berücksichtigen.
- die Qualität von Zufallszahlengeneratoren zu beurteilen.
- die Eignung von Software- und Hardware-Kryptographie für eine bestimmte Aufgabe abzuschätzen.

Voraussetzungen und Empfehlungen:

Für dieses Modul sind Grundkenntnisse der diskreten Mathematik erforderlich. Weiterhin müssen die Studierenden über grundlegende Kenntnisse der Programmierung verfügen, insbesondere in der Programmiersprache Python.

Literatur:

- Bos, Joppe, et al. : CRYSTALS – Kyber: A CCA-secure module-lattice-based KEM. In: O'Conner, L. (ed.) : Proceedings of the 2018 IEEE European Symposium on Security and Privacy (EuroS&P). London, UK, 2018, pp. 353-367, doi: 10.1109/EuroSP.2018.00032.
- Ferguson, Niels; Schneier, Bruce; Kohno, Tadayoshi: Cryptography Engineering : Design Principles and Practical Applications. Indianapolis (IN), USA: Wiley Publishing, 2010.
- Katz, Jonathan; Lindell, Yehuda : Introduction to Modern Cryptography. Boca Raton (FL), USA: CRC Press, 2007.
- Mao, Wenbo: Modern Cryptography: Theory and Practice, Upper Saddle River (NJ), USA: Prentice Hall, 2003.
- NIST : SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions. FiPS PUB 202, https://doi.org/10.6028/NIST.FIPS.202 , 2015.
- Ristic, Ivan : Bulletproof TLS and PKI. 2. Edition. London, UK: Feisty Duck, 2022.
- Stallings, William: Cryptography and Network Security : Principles and Practice. 8. Edition. London, UK: Pearson, 2022.
- Stinson, Douglas R. : Cryptography : Theory and Practice. 4. Edition. Boca Raton (FL), USA: CRC Press, 2018.
- Swenson, Christopher: Modern Cryptanalysis : Techniques for Advanced Code Breaking. Indianapolis (IN), USA: Wiley Publishing, 2008.

Studiengänge:

- Informatik Master of Science
- IT-Sicherheit Master of Science
- IT Engineering Master of Science

◈ MM019 – Security Engineering

Verantwortliche:	Gerd Beuster
Moduldauer:	6 Monate
Unterrichtssprache:	english

Bestandteile:

Teilleistung:	TM040 – Security Engineering
Lernform:	Vorlesung
Prüfungsform:	Klausur / Mündliche Prüfung
Prüfungsdauer:	60 Min.
ECTS:	5.0
Benotung:	Drittelnoten
Turnus:	jährlich
Dauer (pro Termin):	4 Semesterwochenstunden
Termine im Semester:	12 Termine
Zeit in Veranstaltungen:	30 Stunden
Sonstiger Arbeitsaufwand während Vorlesungszeit:	120 Stunden
Aufwand während Semesterferien:	0 Stunden
Flexibel einteilbarer Aufwand:	0 Stunden
Gesamtaufwand:	150 Stunden
Lehrende:	Gerd Beuster

Studieninhalte:

- Grundbegriffe der IT-Sicherheit
- Sicherheitsmodellierung
- Sicherheitsadministration und physische Sicherheit
- Sicherheit des Betriebssystems
- Identitätsmanagement, Zugriffskontrolle, Security Tokens, Biometrie
- IoT-Sicherheit
- Cloud-Sicherheit
- Reverse Engineering
- IT-Forensik
- Sicherheitsprotokolle
- Methoden der Entwicklung sicherer Software
- Sicherheit bei Nutzung von Large Language Models (LLMs)
- Typische Angriffe auf Softwaresysteme
- Verteilte Systeme / Netzwerksicherheit
- Sichere Hardware

Lernergebnisse:

Nach Abschluss des Moduls sind die Studierenden in der Lage, ...

- die grundlegenden Konzepte der IT-Sicherheit anzuwenden.
- Sicherheitsanforderungen für Software zu definieren und zu überprüfe.
- sichere Software zu entwickeln und zu evaluieren.
- Large Language Models (LLMs) sicher in Software-Systeme einzubinden
- die Qualität von durch LLMs generiertem Code einzuschätzen
- die Sicherheit von Hardware-Komponenten zu beurteilen und evaluieren.
- die Sicherheit von Computernetzwerken zu bewerten.
- sichere Computernetzwerke zu entwerfen.

Voraussetzungen und Empfehlungen:

Das Modul erfordert Grundkenntnisse in den Bereichen Computerarchitektur, Betriebssysteme,

Computernetzwerke und Programmierung.

Literatur:

- Anderson, Ross J.: Security Engineering : A Guide to Building Dependable Distributed Systems. 3. Auflage. Hoboken (NJ), USA: Wiley & Sons, 2020.
- Graves, Michael W.: Digitial Archaeology : The Art and Science of Digital Forensics. Bosten (MA), USA: Addison Wesley, 2014.
- Johansen, Gerard: Digital Forensics and Incident Response : An intelligent way to respond to attacks. 3. Auflage. Birmingham, UK: Packt, 2020.
- Oettinger, William: Learn Computer Forensics : A beginner's guide to searching, analyzing, and securing digital evidence. Packt Publishing. Birmingham, UK, 2020.
- Pfleeger, Charls P.;Pfleeger, Shari Lawrence: Security in Computing. 6. Auflage. München: Prentice Hall, 2023.
- Shimeall, Timothy J.; Spring, Jonathan M.: Introduction to Information Security : A Strategic-based Approach. Amsterdam, NL: Elsevier Syngress, 2013.
- Stallings, William: Computer Security : Principles and Practice. 4. Auflage. München: Pearson, 2018.
- Watson, David; Jones, Andrew: Digital Forensics Processing and Procedures. Amsterdam, NL: Elsevier Syngress, 2013.
- Weidman, Georgia: Penetration Testing : A Hands-On Introduction to Hacking. San Francisco (CA), USA: No Starch Press, 2014

Studiengänge:

- IT-Sicherheit Master of Science
- IT Engineering Master of Science

◈ MM120 – Web- und Applikationssicherheit

Verantwortliche:	Gerd Beuster
Moduldauer:	6 Monate
Unterrichtssprache:	deutsch/englisch

Bestandteile:

Teilleistung:	TM042 – Web- und Applikationssicherheit
Lernform:	Workshop
Prüfungsform:	Abnahme
Prüfungsdauer:	
ECTS:	5.0
Benotung:	Bestanden/nicht Bestanden
Turnus:	Sommersemester
Dauer (pro Termin):	4 Semesterwochenstunden
Termine im Semester:	12 Termine
Zeit in Veranstaltungen:	30 Stunden
Sonstiger Arbeitsaufwand während Vorlesungszeit:	*Nicht angegeben.*
Aufwand während Semesterferien:	*Nicht angegeben.*
Flexibel einteilbarer Aufwand:	*Nicht angegeben.*
Gesamtaufwand:	*Gesamtaufwand kann nicht ermittelt werden (braucht SWS und Häufigkeit, ggf. müssen die übrigen Zeiten explizit auf 0 gesetzt werden).*
Lehrende:	Timo Pagel

Studieninhalte:

- Auffrischung moderne Web-Architekturen
- Ursachen für unsichere Webanwendungen
- Schwachstellen und Angriffe
- Technische Sicherheitsmaßnahmen
- Organisatorische (Web-)Anwendungssicherheit
- Integration von Sicherheit in den Entwicklungszyklus
- Testverfahren
- Relevante Standards

Lernergebnisse:

Die Studierenden ...

- kennen typische Schwachstellen und Sicherheitsprobleme von (Web-)Anwendungen und können die notwendigen Maßnahmen entwickeln und umsetzen, um diese zu vermeiden.
- sind in der Lage typische Schwachstellen bei selbstentwickelten (Web-)Anwendungen zu vermeiden.
- sind in der Lage, neben den technischen auch die relevanten organisatorischen Maßnahmen umzusetzen, um die Sicherheit von (Web-)Anwendungen zu gewährleisten.
- sind mit den relevanten Standards zur Applikationssicherheit vertraut und können diese anwenden.
- sind mit den relevanten Testmethodiken vertraut und sind in der Lage gängige Sicherheitsprobleme in Webanwendungen selbst zu identifizieren.

Voraussetzungen und Empfehlungen:

Für diese Veranstaltung werden grundlegende Programmierkenntnisse, grundlegende Container-Kentnisse und grundlegende Web-Architekturkentnisse benötigt.

Literatur:

- Shostack, Adam. THREATS: What Every Engineer Should Learn from Star Wars. John Wiley & Sons, 2023.
- Bratus, Sergey, et al. "Curing the vulnerable parser: Design patterns for secure input handling." login:, Usenix Magazine 42 (2017).
- Cao, Sicong, et al. "Improving java deserialization gadget chain mining via overriding-guided object generation." arXiv preprint arXiv:2303.07593 (2023).
- Matthias Rohr: Sicherheit von Webanwendungen in der Praxis, 2015, 78-3658038502
- Ross J. Anderson, Security Engineering: A Guide to Building Dependable Distributed Systems, 2008, ISBN 978-0470068526
- David Rice, Geekonomics: The Real Cost of Insecure Software, 2010, David Rice, ISBN 978-0321735973
  Gary McGraw, Software Security: Building Security in (Addison-Wesley Software Security), 2006, 978-0321356703
- Joel Scambray, Vincent Liu, Caleb Sima: Hacking Exposed: Web Applications: Web Application Security Secrets and Solutions. 3. Auflage. Verlag Mcgraw-Hill Professional, 2010, ISBN 978-0-07-174064-7.
  William Stallings, Lawrie Brown: Computer Security - Principles and Practice, Third Edition, Pearson, 2015
- Abhinav Singh: Metasploit Penetration Testing Cookbook, Packt Publishing, 2012-06-22
  Georgia Weidmanf: Penetration Testing - A Hands-On Introduction to Hacking, No Starch Press, 2014
  Vivek Ramachandran, Cameron Buchanan: Kali Linux - Wireless Penetration Testing Beginners Guide, Packt Publishing, 2015
- Kevin Cardwell: Building Virtual Pentesting Labs for Advanced Penetration Testing, Packt Publishing, 2014-07-19
- Aaron Johns: Mastering Wireless Penetration Testing for Highly Secured Environments, Packt Publishing, 2015
- Joseph Muniz, Aamir Lakhani: Web Penetration Testing with Kali Linux, Packt Publishing, 2013

Studiengänge:

- IT-Sicherheit Master of Science

◈ MM162 – Moderne Software-Architekturen

Verantwortliche:	Ulrich Hoffmann
Moduldauer:	6 Monate
Unterrichtssprache:	deutsch

Bestandteile:

Teilleistung:	TM039 – Moderne Software-Architekturen
Lernform:	Vorlesung
Prüfungsform:	Portfolio-Prüfung
Prüfungsdauer:	30 Min.
ECTS:	5.0
Benotung:	Drittelnoten
Turnus:	Sommersemester
Dauer (pro Termin):	4 Semesterwochenstunden
Termine im Semester:	12 Termine
Zeit in Veranstaltungen:	30 Stunden
Sonstiger Arbeitsaufwand während Vorlesungszeit:	24 Stunden
Aufwand während Semesterferien:	0 Stunden
Flexibel einteilbarer Aufwand:	32 Stunden
Gesamtaufwand:	*Gesamtaufwand 86 Stunden weicht signifikant von erwarteten 5.0 ECTS ab.*
Lehrende:	Ulrich Hoffmann

Studieninhalte:

- Virtualisierung
- Container
- Cloud Computing
- Serverless Computing
- grundledende verteilte Algorithmen
  - Konsensalgorithmen,
  - Synchronisationsalogrithmen
- Architekturmuster
  - Map/Reduce
  - Service Orientierte Architektur
  - Microservices
  - Pipelines
    - AI Pipelines
    - Deployment Pipelines
    - DevOps Pipelines
- Seminaristischer Unterricht
  - Kurzvortrag zu ausgewähltem Grundlagenthema
  - praktisches Projekt zu ausgewählter Aufgabenstellung
    - regelmäßige Projektbesprechungen
    - Wissenschaftlicher Vortrag zur Vorstellung der Projektergebnisse

Lernergebnisse:

Die Studierenden ...

- besitzen eingehende Kenntnisse über Struktur, Entwicklung, Bereitstellung und Betrieb großer Software-Systeme.
- haben die Fähigkeit umfangreiche Softwaresysteme bereitzustellen.
- haben die Fähigkeit eigene Softwaresysteme bereitzustellen.
- besitzen Kenntnisse und Fähigkeiten, große Softwaresystem automatisch überwachen zu lassen und Messungen an ihnen durchzuführen.
- Sie besitzen grundlegende Fähigkeiten aufgrund von Messergebnissen eine Optimierung und Skalierung von Softwaresystemen durchzuführen.

- haben die Fähigkeit, für ein Softwaresystem Anforderungen zu erheben, es geeignet zu strukturieren und auf angemessene Weise in Betrieb zu nehmen.
- besitzen die Fähigkeit ihr Vorgehen kritisch zu hinterfragen und nachvollziehbar zu kommunizieren.
- besitzen die Fähigkeit die von ihnen im Rahmen der Projektarbeit erarbeiteten Sachverhalte zu kondensieren und in angemessenen Vortragsstil und geeigneter Präsentationstechniken nachvollziehbar dazustellen. In freier Diskussion können sie sich über komplexe wissenschaftliche Sachverhalte der großen Softwaresysteme und ihres Betriebs auseinandersetzen.

Voraussetzungen und Empfehlungen:

- Erfahrung in der imperativen und objekt-orientierten Programmierung
- grundlegendes Verständnis der Sicherheitsanforderungen in der Informationstechnik
- Fähigkeit zur Implementierung von Sicherheitsmaßnahmen
- Kentnisse über die Funktionsweisen von Rechnetzen und des World Wide Webs / Internets.

Literatur:

- Bass, Weber, Zhu: DevOps: A Software Architect's Perspective, Addison Wesley 2015
- Bass, Kazman, Clements: Software Architecture in Practice, Addison Wesley, 2012
- Fowler: Patterns of Enterprise Application Architecture, Addison Wesley, 2002
- Hapke, Nelson: Building Machine Learning Pipelines, O'Reilly Media, 2020
- Humble, Farley: Continuous Delivery: Reliable Software Releases Through Build, Test, and Deployment Automation, Addison-Wesley 2010
- Josuttis: SOA in Practice: The Art of Distributed System Design, O'Reilly Media 2007
- Kim, Willis, Debois, Humble: The DevOPS Handbook, IT Revolution Press, 2016
- Krafzig, Banke, Slama: Enterprise SOA: Service-Oriented Architecture Best Practices, Prentice Hall 2004
- Newman: Monolith to Microservices, O'Reilly Media, 2019
- Saito, Lee, Wu: DevOps with Kubernetes, Packt Publishing, 2019

Studiengänge:

- Informatik Master of Science
- IT-Sicherheit Master of Science
- Wirtschaftsinformatik / IT-Management Master of Science

◈ MM170 – Seminar IT-Sicherheit

Verantwortliche:	Gerd Beuster
Moduldauer:	6 Monate
Unterrichtssprache:	deutsch/englisch

Bestandteile:

Teilleistung:	TM024 – Seminar
Lernform:	Seminar
Prüfungsform:	Schriftl. Ausarbeitung (ggf. mit Präsentation)
Prüfungsdauer:	60 Min.
ECTS:	5.0
Benotung:	Drittelnoten
Turnus:	Sommersemester
Dauer (pro Termin):	1 Semesterwochenstunden
Termine im Semester:	12 Termine
Zeit in Veranstaltungen:	7 Stunden
Sonstiger Arbeitsaufwand während Vorlesungszeit:	135 Stunden
Aufwand während Semesterferien:	0 Stunden
Flexibel einteilbarer Aufwand:	0 Stunden
Gesamtaufwand:	142 Stunden
Lehrende:	*Gerd Beuster*

Studieninhalte:

Fachvorträge mit anschließender Gruppendiskussion.

Lernergebnisse:

Die Studierenden ...

- sind in der Lage, eine wissenschaftliche fundierte Lösung für theoretische und/oder praktische Problemstellungen primär aus dem Themengebiet sowie ähnlichen Gebieten zu entwickeln.
- zeigen eine verbesserte Problemlösungstechnik, sicherere Verwendung von Termini, präzise Strukturierung im Aufbau schriftlicher Arbeiten und Einhalten der Formalia.
- zeigen eine auf Masterniveau angemessene Vortragstechnik im Rahmen der Präsentation der Ergebnisse.

Voraussetzungen und Empfehlungen:

Voraussetzungen und Empfehlungen nicht angegeben.

Literatur:

Recherche nach aufgabenbezogener Literatur, teilweise aufgabenspezifische Vorgabe einzelner Literaturquellen.

Studiengänge:

- IT-Sicherheit Master of Science

◈ MM197 – Konzepte des Machine Learning

Verantwortliche:	Christian-Arved Bohn
Moduldauer:	6 Monate
Unterrichtssprache:	deutsch

Bestandteile:

Teilleistung:	TM121 – Konzepte des Machine Learning
Lernform:	Vorlesung mit integrierter Übung
Prüfungsform:	Abnahme
Prüfungsdauer:	20 Min.
ECTS:	5.0
Benotung:	Drittelnoten
Turnus:	Sommersemester
Dauer (pro Termin):	4 Semesterwochenstunden
Termine im Semester:	24 Termine
Zeit in Veranstaltungen:	60 Stunden
Sonstiger Arbeitsaufwand während Vorlesungszeit:	24 Stunden
Aufwand während Semesterferien:	12 Stunden
Flexibel einteilbarer Aufwand:	54 Stunden
Gesamtaufwand:	150 Stunden
Lehrende:	Ulrich Hoffmann

Studieninhalte:

Diese Vorlesung bietet eine fundierte Einführung in die theoretischen Grundlagen und praktischen Ansätze des Maschinellen Lernens. Ziel ist es, den Studierenden nicht nur die Anwendung bestehender Algorithmen zu vermitteln, sondern ein tiefes Verständnis für die zugrunde liegenden mathematischen und logischen Prinzipien zu entwickeln. Es wird hierbei der gesamten Lebenszyklus eines ML-Modells – von der Datenrepräsentation bis zur Evaluierung betrachtet.

Lernergebnisse:

Nach erfolgreichem Abschluss des Moduls verfügen die Studierenden über ein tiefgreifendes Verständnis der theoretischen Grundlagen des Machine Learning und können zwischen überwachten, unüberwachten sowie bestärkenden Lernparadigmen differenzieren. Sie besitzen die methodische Kompetenz, für spezifische Problemstellungen eigenständig geeignete Algorithmen auszuwählen und diese effizient zu implementieren. Sie sind in der Lage, Modelle durch gezieltes Hyperparameter-Tuning zu optimieren und deren Vorhersagegüte objektiv zu validieren.

Voraussetzungen und Empfehlungen:

Basiswissen in der Programmierung, insbesondere mittels Python, Grundlagen der Informatik und Statistik.

Literatur:

- Ian Goodfellow, Yoshua Bengio & Aaron Courville: Deep Learning
- Christopher M. Bishop: Pattern Recognition and Machine Learning
- Kevin P. Murphy: Probabilistic Machine Learning

Studiengänge:

- Data Science & Artificial Intelligence Master of Science
- Informatik Master of Science
- IT-Sicherheit Master of Science

- Wirtschaftsinformatik / IT-Management Master of Science

◈ MM201 – Physische Sicherheit

Verantwortliche:	Gerd Beuster
Moduldauer:	6 Monate
Unterrichtssprache:	Deutsch

Bestandteile:

Teilleistung:	TM124 – Physische Sicherheit
Lernform:	Vorlesung
Prüfungsform:	Schriftl. Ausarbeitung (ggf. mit Präsentation)
Prüfungsdauer:	
ECTS:	5.0
Benotung:	Drittelnoten
Turnus:	Sommersemester
Dauer (pro Termin):	4 Semesterwochenstunden
Termine im Semester:	19 Termine
Zeit in Veranstaltungen:	47 Stunden
Sonstiger Arbeitsaufwand während Vorlesungszeit:	102 Stunden
Aufwand während Semesterferien:	0 Stunden
Flexibel einteilbarer Aufwand:	0 Stunden
Gesamtaufwand:	149 Stunden
Lehrende:	Gerd Beuster

Studieninhalte:

- Perimetersicherung
- Materialien (Eigenschaften, Funktion und Fähigkeiten)
- Risikoanalyse und Planung
- bauliche Absicherung von Sonderbauten
- Testverfahren und Berechnungsmethoden
- Zufahrtsschutz
- Normen und Richtlinien
- KritisDachG, BSI-Grundschutz und NIS2/BSIG

Lernergebnisse:

Die Studierenden erhalten anhand von Praxisbeispielen, Begehungen und Übungen tiefgehenden Einblick in die Auswahl und Projektierung von Maßnahmen der physischen Sicherheit im Objektschutz. Darauf aufbauend werden die Studierenden dazu befähigt in enger Abstimmung mit Fachplanern Sicherheitskonzeptionen für Liegenschaften zu entwickeln und ausschreibungsreif zu gestalten.

Voraussetzungen und Empfehlungen:

Formal: keine
Inhaltlich: keine

Literatur:

- Verband für Sicherheitstechnik (Hrsg.), Handbuch Perimetersicherung, IGT Verlag, 3. Auflage, München 2014.
- Verband für Sicherheitstechnik (Hrsg.), Handbuch Zufahrtsschutz, IGT Verlag, München 2024.
- Kraheck, Adolf/Zahn, Susanne, Grundlagen der Perimetersicherung, Berlin, Offenbach, VDE Verlag 2016.

Siehe zusätzlich Literaturliste in den aktuellen Studienmaterialien!

Studiengänge:

- IT-Sicherheit Master of Science

◈ MM027 – Konzepte der Datenbanktechnologie

Verantwortliche:	Dennis Proppe
Moduldauer:	6 Monate
Unterrichtssprache:	deutsch

Bestandteile:

Teilleistung:	TM002 – Konzepte der Datenbanktechnologie
Lernform:	Vorlesung
Prüfungsform:	Klausur / Mündliche Prüfung
Prüfungsdauer:	60 Min.
ECTS:	3.0
Benotung:	Drittelnoten
Turnus:	jährlich
Dauer (pro Termin):	2 Semesterwochenstunden
Termine im Semester:	*Häufigkeit nicht angegeben.*
Zeit in Veranstaltungen:	*Kontaktzeit kann nicht ermittelt werden (braucht SWS und Häufigkeit).*
Sonstiger Arbeitsaufwand während Vorlesungszeit:	*Nicht angegeben.*
Aufwand während Semesterferien:	*Nicht angegeben.*
Flexibel einteilbarer Aufwand:	*Nicht angegeben.*
Gesamtaufwand:	*Gesamtaufwand kann nicht ermittelt werden (braucht SWS und Häufigkeit, ggf. müssen die übrigen Zeiten explizit auf 0 gesetzt werden).*
Lehrende:	Dennis Proppe Ulrich Hoffmann

Teilleistung:	TM003 – Übg. Konzepte der Datenbanktechnologie
Lernform:	Übung
Prüfungsform:	Abnahme
Prüfungsdauer:	10 Min.
ECTS:	2.0
Benotung:	Bestanden/nicht Bestanden
Turnus:	jährlich
Dauer (pro Termin):	2 Semesterwochenstunden
Termine im Semester:	*Häufigkeit nicht angegeben.*
Zeit in Veranstaltungen:	*Kontaktzeit kann nicht ermittelt werden (braucht SWS und Häufigkeit).*
Sonstiger Arbeitsaufwand während Vorlesungszeit:	*Nicht angegeben.*
Aufwand während Semesterferien:	*Nicht angegeben.*
Flexibel einteilbarer Aufwand:	*Nicht angegeben.*
Gesamtaufwand:	*Gesamtaufwand kann nicht ermittelt werden (braucht SWS und Häufigkeit, ggf. müssen die übrigen Zeiten explizit auf 0 gesetzt werden).*
Lehrende:	Tim Wetzel

Studieninhalte:

- Grundlagen Datenbanksysteme
  - Persistenz
  - Transaktionen
  - 2PL
  - Datenschutz und Datensicherheit
- Objekt-relationales Mapping
  - Java Persistence API (JPA)
- NoSQL-Datenbanksysteme
  - Verteilte Wert/Schlüssel-Speicher
  - Dokumentendatenbanken
  - Graph-Datenbanken
- Verteilung von Daten

Vorlesungsbegleitende praktische Übungen zu Objektrelationalem Mapping und anderen alternativen Persistenzansätzen.

Erstellung einer NoSQL-Datenbank mit einem kompletten CRUD-Zyklus.

Lernergebnisse:

Studierende ...

- beherrschen die Fähigkeit Objektrelationales Mapping anzuwenden bzw. in Betrieb zu nehmen und es zur Lösung von Problemen einzusetzen.
- sind mit den praktisch auftretenden Schwierigkeiten vertraut und können sie systematisch überwinden.
- sind in der Lage eine NoSQL-Datenbank einzurichten, sie mit Daten zu füllen und anfragen an sie zu stellen

---

Die Studierenden erlangen die ...

- Kenntnis, der für die Implementierung von Datenbanksystemen wichtigen Architekturprinzipien, Datenstrukturen und Algorithmen und damit Kenntnis des Aufbaus und der internen Arbeit eines großen komplexen Softwaresystems.
- Fähigkeit, die Arbeitsweise von Datenbanksystemen zu optimieren bzw. selbst Architekturen für große komplexe Softwaresysteme zu entwerfen.
- Fähigkeiten eines Datenbankadministrators für Datenbanksysteme.
- Konzepte und Techniken des Datenschutzes, als auch der Datensicherheit

Voraussetzungen und Empfehlungen:

- Vertrautheit mit den grundlegenden Konzepten von Datenbanksystemen, einschließlich der Prinzipien der Persistenz und Transaktionen.
- Verständnis der relationalen Datenbankmodelle und Kenntnisse in SQL.
- Kenntnisse in der objekt-orientierten Programmierung

Literatur:

- siehe Vorlesung
- diverse Online-Quellen

- KEMPER, Alfons; EICKLER, Andre:
  Datenbanksysteme - Eine Einführung. Oldenbourg Verlag, 2004
- KEITH, Mike; SCHINCARIOL, Merrik:
  Pro JPA 2 - Mastering the Java Persistence API. APress, 2009
- BAUER, Christian; KING, Gavin:
  Java Persistence with Hibernate,
  Manning, Greenwich, 2007
- SQL- & NoSQL-Datenbanken – Andreas Meier, Michael Kaufmann; eXamen.press Springer Vieweg
- Sieben Wochen, sieben Datenbanken – Eric Redmond, Jim R. Wilson; O‘Reilly
- NoSQL for Dummies, Adam Fowler; For Dummies-Verlag
- div. Konferenzbeiträge und Forschungsarbeiten zu moderneren Entwicklungen der Datenbanktechnologie

Studiengänge:

- Data Science & Artificial Intelligence Master of Science
- Informatik Master of Science
- IT-Sicherheit Master of Science
- Wirtschaftsinformatik / IT-Management Master of Science

◈ MM029 – Berechenbarkeit und Verifikation

Verantwortliche:	Sebastian Iwanowski
Moduldauer:	6 Monate
Unterrichtssprache:	deutsch/englisch

Bestandteile:

Teilleistung:	TM033 – Berechenbarkeit und Komplexität, Formale Spezifikation und Verifikation
Lernform:	Vorlesung
Prüfungsform:	Klausur / Mündliche Prüfung
Prüfungsdauer:	120 Min.
ECTS:	5.0
Benotung:	Drittelnoten
Turnus:	Wintersemester
Dauer (pro Termin):	6 Semesterwochenstunden
Termine im Semester:	12 Termine
Zeit in Veranstaltungen:	45 Stunden
Sonstiger Arbeitsaufwand während Vorlesungszeit:	45 Stunden
Aufwand während Semesterferien:	60 Stunden
Flexibel einteilbarer Aufwand:	0 Stunden
Gesamtaufwand:	150 Stunden
Lehrende:	Ulrich Hoffmann Sebastian Iwanowski

Studieninhalte:

- Berechenbarkeit und Nichtberechenbarkeit
  - Präzisierung der Begriffe Problem und Algorithmen für die Theorie der Berechenbarkeit
  - Turingmaschinen im Detail
  - Komplexitätsklassen für Turingmaschinen
  - Beispiele für unentscheidbare Probleme
  - Beweise der Unentscheidbarkeit für ausgewählte Probleme
- NP-vollständige Probleme
  - Historie des P-NP-Problems
  - Beweis der NP-Vollständigkeit von SATISFIABILITY
  - Übersicht über NP-vollständige Probleme
  - Reduktionsmethode zum Beweis von NP-Vollständigkeit mit Beispielen
- Optimierungsaufgaben für NP-vollständige Probleme
  - Lösungstechniken für NP-vollständige Probleme
  - Übersicht über wichtige Anwendungen - Vergleich zu Verfahren der Künstlichen Intelligenz

---

- Mathematische und logische Grundlagen der Spezifikation und Verifikation; Mengen, Multimengen, Verbände, partielle und totale Funktionen, algebraische Strukturen, Aussagen- und Prädikatenlogik, Modallogik, temporale Logik
- Algebraische Spezifikation; Terme, Gleichungen; Fallbeispiel einer algebraischen Spezifikation; Datenstrukturen, Operationen, Nachweis von Eigenschaften; maschinenunterstütztes Beweisen von Eigenschaften
- Modellorientierte Spezifikation; Fallbeispiel einer modellorientierten Spezifikation
- Konstruktion korrekter Programme aus Spezifikationen
- Aktuelle Spezifikationssprachen im Überblick

Lernergebnisse:

Nach Abschluss der Veranstaltung besitzen die Studierenden folgende Kompetenzen:

- Fundierter theoretischer Überblick über die Möglichkeiten des Problemlösens.
- Theoretisch fundierte Kenntnis der Grenzen der Berechenbarkeit und der effizienten Lösbarkeit.
- Kenntnis der Alternativen für die Praxis bei theoretisch unbefriedigenden Resultaten.

Die Studierenden ...

- erlangen fundierte Kenntnisse der mathematischen Grundlagen der formalen Spezifikation und Verifikation.
- beherrschen verschiedene Spezifikationsstile.
- bekommen einen Einblick in verschiedene formale Spezifikationssprachen.
- erlangen die Fähigkeit, Spezifikationen systematisch zu konstruieren.
- können mathematische Beweise von Eigenschaften spezifizierte Software-Systeme führen.
- erlangen grundlegende Kenntnisse der Verifikation mit automatischen Beweissystemen.
- sind mit der Spezifikation von Sicherheitsbedingungen vertraut.
- können beurteilen, wie sich formale Methoden auf die IT-Sicherheit auswirken.

Voraussetzungen und Empfehlungen:

Diskrete Mathematik aus dem Bachelorstudium, im Besonderen die Mengenlehre und Beweise

Literatur:

- Garey, Michael R.; Johnson, David S. (1979), Computers and Intractability: A Guide to the Theory of NP-Completeness, W. H. Freeman, ISBN 0-7167-1045-5
- Hopcroft, John E.; Motwani, Rajeev; Ullman, Jeffrey D.:
  Einführung in die Automatentheorie, Formale Sprachen und Komplexitätstheorie.
  2. überarb. Aufl. München: Addison-Wesley Longman Verlag, 2002.
- Vossen,Gottfried; Witt, Kurt-Ulrich:
  Theoretische Informatik.
  Braunschweig: Verlag Vieweg & Teubner 2004 (3. Auflage), ISBN 978-3528231477
- Wagenknecht, C.:
  Algorithmen und Komplexität,
  Fachbuchverlag Leipzig 2003
- Winter, R.:
  Theoretische Informatik,
  Oldenbourg-Verlag München 2002

- BJØRNER, Dines:
  Software Engineering 1.
  Heidelberg: Springer Verlag, 2006
- DILLER, Antoni:
  Z An Introduction to Formal Methods.
  New York: Wiley & Sons, 1994
- EHRICH/GOGOLLA/LIPECK:
  Algebraische Spezifikation abstrakter Datentypen.
  Stuttgart: Teubner Verlag, 1989
- GOOS, Gerhard:
  Vorlesungen über Informatik Band 1 - Grundlagen und funktionales Programmieren.
  Heidelberg: Springer Verlag, 2005
- LAMPORT, Leslie:
  Specifying Systems.
  Amsterdam: Addison-Wesley, 2002
- SCHÖNING, Uwe:
  Logik für Informatiker.
  Heidelberg: Spektrum Akademischer Verlag, 2000
- WORDSWORTH, J., B.:
  Software Development with Z.
  New York: Addison-Wesley, 1992

Studiengänge:

- Informatik Master of Science
- IT-Sicherheit Master of Science

◈ MM035 – Distributed Systems

Verantwortliche:	Ulrich Hoffmann
Moduldauer:	6 Monate
Unterrichtssprache:	english

Bestandteile:

Teilleistung:	TM006 – Distributed Systems
Lernform:	Vorlesung
Prüfungsform:	Klausur / Mündliche Prüfung
Prüfungsdauer:	60 Min.
ECTS:	3.0
Benotung:	Drittelnoten
Turnus:	jährlich
Dauer (pro Termin):	2 Semesterwochenstunden
Termine im Semester:	*Häufigkeit nicht angegeben.*
Zeit in Veranstaltungen:	*Kontaktzeit kann nicht ermittelt werden (braucht SWS und Häufigkeit).*
Sonstiger Arbeitsaufwand während Vorlesungszeit:	*Nicht angegeben.*
Aufwand während Semesterferien:	*Nicht angegeben.*
Flexibel einteilbarer Aufwand:	*Nicht angegeben.*
Gesamtaufwand:	*Gesamtaufwand kann nicht ermittelt werden (braucht SWS und Häufigkeit, ggf. müssen die übrigen Zeiten explizit auf 0 gesetzt werden).*
Lehrende:	Ulrich Hoffmann

Teilleistung:	TM007 – Tutorial: Distributed Systems
Lernform:	Übung
Prüfungsform:	Abnahme
Prüfungsdauer:	5 Min.
ECTS:	2.0
Benotung:	Bestanden/nicht Bestanden
Turnus:	jährlich
Dauer (pro Termin):	2 Semesterwochenstunden
Termine im Semester:	*Häufigkeit nicht angegeben.*
Zeit in Veranstaltungen:	*Kontaktzeit kann nicht ermittelt werden (braucht SWS und Häufigkeit).*
Sonstiger Arbeitsaufwand während Vorlesungszeit:	*Nicht angegeben.*
Aufwand während Semesterferien:	*Nicht angegeben.*
Flexibel einteilbarer Aufwand:	*Nicht angegeben.*
Gesamtaufwand:	*Gesamtaufwand kann nicht ermittelt werden (braucht SWS und Häufigkeit, ggf. müssen die übrigen Zeiten explizit auf 0 gesetzt werden).*
Lehrende:	Ulrich Hoffmann

Studieninhalte:

- Praktische Beispiele
- Allgemeine Anforderungen an verteilte Systeme
- Die Client-Server-Beziehung und daraus resultierende Fragen
- Kommunikation in verteilten Systemen
- Dienste benennen
- Techniken für Gleichzeitigkeit
- Ferngespräche
- Alternative Paradigmen (Akteurskonzept, ...)
- Synchronisierung von Daten und Prozessen
- Koordinationsmethoden
- Replikationstechniken
- WEB-Dienste mit SOAP und REST
- Fehlertoleranzkonzepte
- Sicherheit in verteilten Systemen
- Programmierung mit Threads
- Kommunikation über Sockets, Struktur von Clients und Servern
- Ferner Prozeduraufruf / entfernter Methodenaufruf
- Verwendung von Benennungsdiensten
- Programmierung von WEB-Diensten (SOAP, Server/Client, WSDL, Datenbindung)
- verteiltes Programmieren mit alternativen Konzepten
- Programmierung von Synchronisierungsalgorithmen
- Programmierung verteilter Wahlalgorithmen
- Programmierung von REST-basierten Dienstleistungen und Kunden
- Fehlertolerante Programmierung in verteilten Systemen

Vorlesung mit begleitenden praktischen Übungen zur Programmierung verteilter Systeme und ihrer Algorithmen in verschiedenen Programmierparadigmen.

Lernergebnisse:

Die Studierenden gewinnen ...

- gründliches Verständnis der Prinzipien verteilter Anwendungen.
- Kenntnisse in der Beherrschung von Basistechnologien und aktuellen Software-Werkzeugen für verteilte Systeme.
- Zustandskenntnis der sind in verschiedenen Anwendungsbereichen wie Dienstleistungsvermittlung und E-Commerce.
- Kenntnisse über IT-Sicherheitsfragen in verteilten Systemen sowie über verschlüsselter Kommunikation.
- Kenntnisse der grundlegenden Algorithmen in verteilten Systemen.
- genaue Kenntnis der aktuellen Web-Service-Architekturen.
- praktische Fähigkeiten zur Realisierung eines Projekts.
- verteilte Programmierkenntnisse in verschiedenen Paradigmen.

Die Studenten ...

- erlangen die Fähigkeit, typische Softwaresysteme (Middleware) im Bereich der verteilten Systeme zu bedienen und zur Problemlösung einzusetzen.
- sind an Probleme gewöhnt, die in der Realität auftreten, und in der Lage, diese zu überwinden.
- haben einige praktische Erfahrungen mit IT-Sicherheitsfragen.
- wissen, wie man Verschlüsselung in verteilten Umgebungen einsetzt.
- eignen sich durch praktische Erfahrung ein tiefes Wissen über die spezifischen Eigenschaften verteilter Systeme an. Sie können diese Eigenschaften kategorisieren und bewerten.

Voraussetzungen und Empfehlungen:

- Erfahrung in der imperativen und objekt-orientierten Programmierung
- grundlegendes Verständnis der Sicherheitsanforderungen in der Informationstechnik
- Fähigkeit zur Implementierung von Sicherheitsmaßnahmen
- Kentnisse über die Funktionsweisen von Rechnetzen und des World Wide Webs / Internets.

Literatur:

- siehe Vorlesung
- Zahlreiche Online-Ressourcen

- ARMSTRONG, Joe:
  Programming Erlang.
  Pragmatic Programmers, 2007
- ODERSKY, Martin; SPOON, Lex; VENNERS, Bill:
  Programming in Scala.
  Artima Press, Mountain View, 2008
- COULOURIS, George; DOLLIMORE, Jean; KINDBERG, Tim:
  Distributed Systems, Concepts and Design.
  Addison-Wesley, 2011, ISBN 0-1321-4301-1
- TANENBAUM, Andrew; VAN STEEN, Marten:
  Distributed Systems, Priciples and Paradigms.
  Prentice Hall, 2006, ISBN 0-1323-9227-5

Studiengänge:

- Informatik Master of Science
- IT-Sicherheit Master of Science
- IT Engineering Master of Science

- Wirtschaftsinformatik / IT-Management Master of Science

◈ MM047 – Projekt IT-Sicherheit

Verantwortliche:	Gerd Beuster
Moduldauer:	6 Monate
Unterrichtssprache:	deutsch/englisch

Bestandteile:

Teilleistung:	TM041 – Projekt IT-Sicherheit
Lernform:	Projektarbeit
Prüfungsform:	Schriftl. Ausarbeitung (ggf. mit Präsentation)
Prüfungsdauer:	60 Min.
ECTS:	5.0
Benotung:	Drittelnoten
Turnus:	jährlich
Dauer (pro Termin):	4 Semesterwochenstunden
Termine im Semester:	12 Termine
Zeit in Veranstaltungen:	30 Stunden
Sonstiger Arbeitsaufwand während Vorlesungszeit:	120 Stunden
Aufwand während Semesterferien:	0 Stunden
Flexibel einteilbarer Aufwand:	0 Stunden
Gesamtaufwand:	150 Stunden
Lehrende:	Gerd Beuster

Studieninhalte:

Die Inhalte variieren von Veranstaltung zu Veranstaltung. Die Themensetzung orientiert sich an aktuellen Produkten und Entwicklungen in der IT-Sicherheit.

Lernergebnisse:

Die Studierenden ...

- führen über weiterführende theoretische und praktische Kenntnisse in einem ausgewählten Bereich der IT-Sicherheit.
- verfügen über die Fähigkeit, in IT-Sicherheitsprojekten Leitungsfunktionen zu übernehmen.
- sind zur Arbeit in internationalen Teams befähigt.
- können die besonderen Anforderungen von IT-Sicherheitsprojekten im Change Management berücksichtigen.

Voraussetzungen und Empfehlungen:

Die Studierenden benötigen die Kenntnisse in Informatik und Programmierung, die für die Zulassung für das Studium erforderlich sind.

Literatur:

Themenabhängig

Studiengänge:

- IT-Sicherheit Master of Science

◈ MM049 – Security Management

Verantwortliche:	Gerd Beuster
Moduldauer:	6 Monate
Unterrichtssprache:	english

Bestandteile:

Teilleistung:	TM008 – Security Management
Lernform:	Vorlesung
Prüfungsform:	Klausur / Mündliche Prüfung
Prüfungsdauer:	60 Min.
ECTS:	5.0
Benotung:	Drittelnoten
Turnus:	jährlich
Dauer (pro Termin):	4 Semesterwochenstunden
Termine im Semester:	12 Termine
Zeit in Veranstaltungen:	30 Stunden
Sonstiger Arbeitsaufwand während Vorlesungszeit:	120 Stunden
Aufwand während Semesterferien:	0 Stunden
Flexibel einteilbarer Aufwand:	0 Stunden
Gesamtaufwand:	150 Stunden
Lehrende:	Gerd Beuster

Studieninhalte:

- Einführung in das IT-Security-Management
- Unternehmenssicherheit als ökonomischer Faktor
- Angreifer und Angriffsziele
- Management sicherheitskritischer IT-Projekte
- IT-Grundschutz und ISO/IEC 27001
- Evaluierungs- und Zertifizierungsschemata in der IT-Sicherheit
- IT-Gesetzgebung
- Business Continuity Management
- Sicherheitstrainings
- Physikalische Sicherheit
- Sicherheitsaudits und Revisionskontrolle
- Sicherheitsmanagement und Qualitätsmanagement

Lernergebnisse:

In dem Modul Security Management lernen die Studierenden, IT-Sicherheit im Kontext von Unternehmensstrategien zu bewerten und zu gestalten. Den Studierenden wird die Fähigkeit vermittelt, Management-Aufgaben im Bereich der IT-Sicherheit zu übernehmen und als IT-Sicherheitsmanager zu arbeiten.

Sie erlangen die ...

- Fähigkeit, Bedrohungen zu identifizieren und zu modellieren.
- Fähigkeit, Risiken zu bewerten.
- Fähigkeit, die Angemessenheit von Sicherheitsmaßnahmen zu bewerten und angemessene Sicherheitsmaßnahmen zu konzipieren.
- Kenntnis der relevanten Standards und Zertifizierungsschemata im Bereich der IT-Sicherheit.
- Fähigkeit, IT-Sicherheit gesetzeskonform umzusetzen.
- Fähigkeit, IT-Sicherheit im Zusammenspiel mit organisatorischen und physischen Sicherheitsanforderungen und -maßnahmen zu gewährleisten.
- Kenntnisse der Zusammenhänge zwischen Sicherheits- und Qualitätsmanagement

Voraussetzungen und Empfehlungen:

Das Modul setzt keine speziellen Kenntnisse voraus, allgemeine Fähigkeiten zum analytischen Denken und zur Modellbildung werden jedoch benötigt.

Literatur:

- BSI - Bundesamt für Sicherheit in der Informationstechnik: BSI-Standards 200-1, 200-2 und 200-3. Version 1.0. Bonn: BSI, 2017.
- Cole, Eric: Advanced Persistent Threat : Understanding the Danger and How to Protect Your Organization. Amsterdam, NL: Elsevier Syngress, 2012.
- Common Criteria for Information Technology Security Evaluation. CC:2022. CCMB-2022-11-001, 2022
- Gantz, Stephen D.: The Basics of IT Audit : Purposes, Processes, and Practical Information. Amsterdam, NL: Elsevier Syngress, 2014.
- Kersten, Heinrich; Klett, Gerhard: Der IT Security Manager. 4. Auflage. Wiesbaden: Springer Vieweg, 2015.
- Smith, Clifton L.; Brooks, David J.: Security Science : The Theory and Practice of Security. Oxford, UK: Butterworth-Heinemann, 2013.
- Snedaker, Susan: IT Security Project Management Handbook. Amsterdam, NL: Elsevier Syngress, 2006.
- Stallings, William: Computer Security : Principles and Practice. 4. Edition. London, UK: Pearson Education, 2018.
- Vacca, John R. (Hrsg.): Computer and Information Security Handbook. 3. Auflage. Burlington (MA), USA: Morgan Kaufmann, 2017.
- Watson, David; Jones, Andrew: Digital Forensics Processing and Procedures. Amsterdam, NL: Elsevier Syngress, 2013.

Studiengänge:

- Betriebswirtschaftslehre Master of Science
- IT-Management / -Consulting & -Auditing Bachelor of Science
- IT-Sicherheit Master of Science
- IT Engineering Master of Science
- Wirtschaftsinformatik / IT-Management Master of Science
- Wirtschaftsingenieurwesen Master of Science

◈ MM121 – Workshop Netzwerksicherheit

Verantwortliche:	Ilja Kaleck
Moduldauer:	6 Monate
Unterrichtssprache:	deutsch

Bestandteile:

Teilleistung:	TM043 – Workshop Netzwerksicherheit
Lernform:	Workshop
Prüfungsform:	Abnahme
Prüfungsdauer:	
ECTS:	5.0
Benotung:	Bestanden/nicht Bestanden
Turnus:	Wintersemester
Dauer (pro Termin):	4 Semesterwochenstunden
Termine im Semester:	12 Termine
Zeit in Veranstaltungen:	30 Stunden
Sonstiger Arbeitsaufwand während Vorlesungszeit:	0 Stunden
Aufwand während Semesterferien:	90 Stunden
Flexibel einteilbarer Aufwand:	30 Stunden
Gesamtaufwand:	150 Stunden
Lehrende:	Ilja Kaleck

Studieninhalte:

Grundlage der eigenständige Arbeit bildet der Aufbau einer eigenem virtualisierten Netzwerk- und Entwicklungsumgebung auf einem bzw. mehreren Workstations (PC) im Labor, darauf aufsetzend erfolgt

- die Konfiguration aktueller Netzkomponenten (Hardware) insbesondere in Bezug aus Sicherheitsaspekte
- eine elementare Cisco-Router bzw. Cisco IOS-Konfiguration, sowie Einsatz von Access Control Lists (ACL)zur Beschränkung des Datenflusses von Zugriffsrechten in Unternehmensnetzen
- eine grundlegende Firewall-Konfiguration (inkl. DMZ-Konzept, Einsatz von VLAN-Technik) und Entwicklung geeigneter Traffic-Management Konzepte (Traffic-Shaper, Proxy-Dienste)
- der Einsatz verschiedener VPN-Konzepte (IPsec, SSL-VPN) und ihre Konfiguration (Site-to-Site,
- der Einsatz von Zertifizierungsstellen zur Absicherung vertraulicher Übertragungskanäle
- der praktische Einsatz von LAN-Analyzer, IDS- und allg. Monitoring-Systemen in Netzen
- die Realisierung einer Layer-2 Anmeldesicherheit in LAN- und WLANs (u.a. per Radius-Server)
- die Einrichtung von allg. Layer-2 Sicherheit durch redundante Kopplung von Teilnetzen (Link-Aggregation Technik, Einsatz von Spanning-Tree Verfahren)

Lernergebnisse:

Die Studierenden erlangen ...

- grundlegende Kenntnisse über den sicheren Aufbau und Betrieb moderner IP-basierter Unternehmensnetze
- die Fähigkeit, bestehende Netzwerke in Bezug auf ihre Sicherheitseigenschaft zu beurteilen und gezielt Schwachstellenanalyse zu betreiben
- aktuelle Werkzeuge zur Traffic-Analyse bzw. allgemeinen Network-Monitoring richtig einzusetzen und praktisch auch selbst nutzen zu können
- die Fähigkeit zum Aufbau sicherer VPN-Szenarien in Unternehmensnetze
- Kenntnisse über den praktischen Einsatz aktueller Verschlüsslungstechniken speziell in Computernetzwerken
- Kenntnisse über Zugriffskontrollmechanismen zu Netzwerken (logisch, physisch)
- Kenntnisse zur Herstellung von Ausfallsicherheit in typischen LAN-Strukturen

Voraussetzungen und Empfehlungen:

Teilnahme am Bachlor-Modul "Rechnernetze" oder einer vergleichbaren Veranstaltung an einer anderen Hochschule, um das notwendige Grundverständis für die "IP-Kommunikation in Unternehmensnetzen" zu haben.

Literatur:

- William Stallings: Cryptography and Network Security, Sixth Edition: Pearson, 2014
- Claudia Eckert: IT-Sicherheit, 9. Auflage 2014: Oldenbourg Verlag
- Günter Schäfer, Michael Roßberg: Netzsicherheit - Grundlagen und Protokolle, 2. Auflage 2014 ; dpunkt.Verlag
- Manfred Lipp: VPN - Virtuelle Private Netzwerke: Aufbau und Sicherheit, 1. Auflage, 2007: Addision-Wesley Verlag
- Eric F Crist, Jan Just Keijser: Mastering OpenVPN, 2015 : Packt Publishing
- John R., Vacca: Network and System Security, 2.ed, 2013: Syngress,
- Mike OLeary: Cyber Operations: Building, Defending, and Attacking Modern Computer Networks, 1.ed 2015: Apress
- James Baxter: Wireshark Essentials, 2014: Packt Publishing
- Justin Hutchens: Kali Linux Network Scanning Cookbook, 2014: Packt Publishing
- David Shaw: Nmap Essentials, 2015: Packt Publishing
- Matt Williamson: pfSense 2 Cookbook, 2011 : Packt Publishing
- Dirk van der Walt: FreeRadius - Beginners Guide, 2011: Packt Publishing
- Alexandre M.S.P. Moraes: Cisco Firewalls, 2011 : Cisco Press
- Christian Sperzel: Netzwerksicherheit, Video-Training 2014: video2brain.com
- Jörg Bueroße: Sichere E-Mails - Verschlüsselung und digitale Signatur, Videotraining 2014: video2brain.com
- Tom Wechsler: Einstieg in die Netzwerkanalyse mit Kali Linux, Videotraining 2015: video2brain.com
- Oliver Bauer, Michael Fritz: Wireshark Grundlagen, Videotraining 2015: video2brain.com

Studiengänge:

- IT-Sicherheit Master of Science

◈ MM202 – IT-Forensik, -Compliance und -Security

Verantwortliche:	Gerd Beuster
Moduldauer:	6 Monate
Unterrichtssprache:	Deutsch

Bestandteile:

Teilleistung:	TM125 – IT-Forensik, -Compliance und -Security
Lernform:	Vorlesung
Prüfungsform:	Portfolio-Prüfung
Prüfungsdauer:	
ECTS:	5.0
Benotung:	Drittelnoten
Turnus:	Wintersemester
Dauer (pro Termin):	4 Semesterwochenstunden
Termine im Semester:	19 Termine
Zeit in Veranstaltungen:	47 Stunden
Sonstiger Arbeitsaufwand während Vorlesungszeit:	102 Stunden
Aufwand während Semesterferien:	0 Stunden
Flexibel einteilbarer Aufwand:	0 Stunden
Gesamtaufwand:	149 Stunden
Lehrende:	Gerd Beuster

Studieninhalte:

- Grundlagen der Informationssicherheit
- Phänomene der Cyberkriminalität
- Herausforderungen und Risiken bei forensischen Untersuchungen
    - Forensische Prinzipien in der digitalen Forensik - Anforderungen, Grenzen und Umsetzungsmöglichkeiten („Chain of Evidence“; „Chain of Custody“)
    - Spurenträger und Computernetzwerke in der IT-Forensik, wie beispielsweise Arbeitsspeicher, Festplatte, Cloudspeicher, Bluetooth, Wifi-Beacons, (versteckte) Dateiinhalte, Metadaten von Dateien, Mobilgeräte, embedded systems, IoT-Devices, Fahrzeugcomputer, ...
    - Forensische Prozesse, deren Implikationen und Grenzen
    - Besondere Herausforderungen der IT-Forensik, insbesondere:
        - volatile vs. nicht-volatile Spuren
        - Rechtliche, technische und organisatorische Grenzen beim Zugriff auf Daten und deren Sicherung als Beweismittel
    - Rechtliche Rahmenbedingungen, insbesondere:
        - DSGVO, BDSG, LDSG
        - NIS-2 Directive
        - DORA
        - Vertragliche und Jedermanns-Befugnisse
        - Beteiligungs- und Mitbestimmungsrechte (z.B. BetrVG)
    - Maßnahmen der IT-Sicherheit sowie der Counter-Forensik in Bezug auf Vertraulichkeit, Integrität, Verfügbarkeit, Transparenz, Authentizität, Zurechenbarkeit, etc. insbesondere durch:
        - Business Continuity Maßnahmen
        - Cyber-Resilienz
        - Verschlüsselung, digitale Signaturen
        - Steganographie
        - Digitale Identitäten
        - Privatsphäre / Anonymisierung
        - OSINT / Social Engineering
        - Hash-Verfahren

Lernergebnisse:

Die Studierenden werden befähigt, cyberkriminelle Handlungen zu erkennen, zu verstehen und durch forensische Maßnahmen aufzuklären. Aufbauend auf Grundbegriffe und Grundzüge der Informationssicherheit können sie die Herausforderungen und Risiken sowie Prozesse bei forensischen Untersuchungen bewerten. Sie lernen die relevanten Rahmenbedingungen zu berücksichtigen. Sie werden in die Lage versetzt, präventive Maßnahmenkonzepte zu entwickeln und zu bewerten.

Voraussetzungen und Empfehlungen:

Formal: keine
Inhaltlich: keine

Literatur:

- Bauer, F. (2000). Entzifferte Geheimnisse. Springer.
- Büchel, M./Hirsch, P. (2020). Internetkriminalität. 2. Auflage. Kriminalistik Verlag.
- Beutelspacher, A. (2015). Kryptologie. Springer Spektrum.
- Casey, E. (2001). Handbook of Computer Crime Investigation: Forensic Tools and Technology. Academic Press.
- Dalby, J. (2016). Grundlagen der Strafverfolgung im Internet und in der Cloud. Springer Fachmedien.
- Deusch, F./Eggendorfer, T. (2024). Beauftragte für IT-Sicherheit und Informationssicherheit. Fachmedien Recht und Wirtschaft.
- Ertel, W./Löhmann, E. (2019). Angewandte Kryptographie. 6. Auflage. Carl Hanser.
- Hartl/Schröder. (2024). Cyber Resilience Act: CRA. Nomos.
- Huber, E. (2019). Cybercrime – Eine Einführung. Springer.
- Kaschner, H. (2020). Cyber Crisis Management. Springer Vieweg.
- Kühling, J./Buchner, B. (2024). Datenschutz-Grundverordnung/BDSG. 4. Auflage. C.H. Beck.
- Lang, M/Löhr, H. (2025). IT-Sicherheit. 2. Auflage. Hanser.
- Lange, H.-J./Bötticher, A. (Hrsg.) (2015). Cyber-Sicherheit, Springer.
- Malek et al. (2024). Strafsachen im Internet. 3. Auflage. C.F. Müller.
- Pohlmann, N. (2022). Cyber-Sicherheit. 2. Auflage. Springer Vieweg.
- Raap-Düsterhöft, A. (2024). IT-Forensik. Springer Vieweg.
- Reddy, N. (2019). Practical Cyber Forensics. Apress.
- Rüdiger, T.-G./Bayerl, P. S. (2023). Handbuch Cyberkriminologie. Band I + II. Springer.
- Schwenk, J. (2020). Sicherheit und Kryptographie im Internet. 5. Auflage. Springer Vieweg.
- Sowa et al. (2019). IT-Revision, IT-Audit und IT-Compliance. 2. Auflage. Springer Vieweg.
- Wernert, M. (2021). Internetkriminalität. Boorberg.

Siehe zusätzlich Literaturliste in den aktuellen Studienmaterialien!

Studiengänge:

- IT-Sicherheit Master of Science

◈ MM203 – Krisenmanagement, Kritische Infrastrukturen und Resilienz

Verantwortliche:	Gerd Beuster
Moduldauer:	6 Monate
Unterrichtssprache:	Deutsch

Bestandteile:

Teilleistung:	TM126 – Krisenmanagement, Kritische Infrastrukturen und Resilienz
Lernform:	Vorlesung
Prüfungsform:	Präsentation / Referat
Prüfungsdauer:	45 Min.
ECTS:	5.0
Benotung:	Drittelnoten
Turnus:	Wintersemester
Dauer (pro Termin):	4 Semesterwochenstunden
Termine im Semester:	19 Termine
Zeit in Veranstaltungen:	47 Stunden
Sonstiger Arbeitsaufwand während Vorlesungszeit:	102 Stunden
Aufwand während Semesterferien:	0 Stunden
Flexibel einteilbarer Aufwand:	0 Stunden
Gesamtaufwand:	149 Stunden
Lehrende:	Gerd Beuster

Studieninhalte:

- Ursachen, Ausprägung und Folgen von Krisenereignissen
- Konzept der „neuen Krisen“ in vernetzten Gesellschaften und die damit verbundenen Herausforderungen
- Arten und Umsetzung des Krisenmanagements (Strukturen, Rollen und Methoden)
- Konzept der kritischen Entitäten und aktuelle Regulierung
- Die Idee der Resilienz im Sicherheitsmanagement und abgeleitete Handlungsfelder
- Grundlagen des Business Continuity Managements
- Klassische und digitale Krisenkommunikation
- Wirtschaftliche Verteidigung
- Führung im Krisenmanagement

Lernergebnisse:

Die Studierenden verstehen Krisen als Herausforderung für Entscheidungs- und Handlungsprozesse in Organisationen und können sie von Störungen, Notfällen und Katastrophen abgrenzen. Sie kennen alle wesentlichen Rollen und Methoden des Krisenmanagements inklusive der Krisenkommunikation (analog und digital), können diese anwenden und interdisziplinäre Krisenteams führen. Sie sind in der Lage, neue Krisenformen (Polykrisen, Perma-Krisen, etc.) und deren Auswirkungen auf die Gesellschaft und einzelne Organisationen zu diskutieren und zu entwickeln. In diesem Zusam- menhang verstehen sie die Bedeutung von kritischen Infrastrukturen und kennen das System der aktuellen Regulierung in der EU. Die Studierenden erweitern ihr allgemeines Verständnis von Krisenmanagement um den Begriff der Resilienz und können Handlungsfelder für Organisationen ableiten. In diesem Zusammenhang sind sie mit nationalen und internationalen Konzepten und Grundanforderungen des Business Continuity Managements vertraut.

Voraussetzungen und Empfehlungen:

Keine Teilnahmebeschränkungen

Literatur:

- Crask, James (2024). Business Continuity Management: A Practical Guide to Organization Resilience and ISO 22301. Kogan Page.
- Kayer, Christopher (2015). Organizational Resilience: How Learning Sustains Organizations in Crisis, Disaster, and Breakdown. Oxford University Press.
- Rubens, David (2923). Strategic Risk and Crisis Management: A Handbook for Modelling and Managing Complex Risks. Kogan Page.
- DIN EN ISO 22361:2023-02
- ISO 22301-2019

Siehe zusätzlich Literaturliste in den aktuellen Studienmaterialien!

Studiengänge:

- IT-Sicherheit Master of Science

◈ MM050 – Master-Thesis

Verantwortliche:	Sergei Sawitzki
Moduldauer:	6 Monate
Unterrichtssprache:	deutsch

Bestandteile:

Teilleistung:	MTH – Master-Thesis
Lernform:	Thesis
Prüfungsform:	Abschlussarbeit
Prüfungsdauer:	60 Min.
ECTS:	27.0
Benotung:	Zehntelnoten
Turnus:	jedes Semester
Dauer (pro Termin):	0 Semesterwochenstunden
Termine im Semester:	12 Termine
Zeit in Veranstaltungen:	0 Stunden
Sonstiger Arbeitsaufwand während Vorlesungszeit:	400 Stunden
Aufwand während Semesterferien:	400 Stunden
Flexibel einteilbarer Aufwand:	40 Stunden
Gesamtaufwand:	840 Stunden
Lehrende:	Sergei Sawitzki

Studieninhalte:

themenabhängig

Lernergebnisse:

Die Studierenden

- können komplexe Aufgabenstellungen selbständig zu erarbeiten
- können Problemstellungen im größeren Kontext zu verorten
- sind in der Lage wissenschaftliche Methoden für die Problemlösung einzusetzen
- können Ergebnisse überzeugend unter besonderer Berücksichtigung der Aspekte wissenschaftlichen Arbeitens darzustellen

Voraussetzungen und Empfehlungen:

Fachliche und persönliche Kompetenzen der zurückliegenden Semester, insbesondere themenabhängig fachverwandte Module

Literatur:

themenabhängig

Studiengänge:

- Betriebswirtschaftslehre Master of Science
- Data Science & Artificial Intelligence Master of Science
- E-Commerce Master of Science
- Informatik Master of Science
- IT-Sicherheit Master of Science
- IT Engineering Master of Science
- Sustainable & Digital Business Management Master of Science
- Wirtschaftsinformatik / IT-Management Master of Science

- Wirtschaftsingenieurwesen Master of Science

◈ MM058 – Master-Kolloquium

Verantwortliche:	Sergei Sawitzki
Moduldauer:	6 Monate
Unterrichtssprache:	deutsch

Bestandteile:

Teilleistung:	TM010 – Master-Kolloquium
Lernform:	Kolloquium
Prüfungsform:	Kolloquium
Prüfungsdauer:	60 Min.
ECTS:	3.0
Benotung:	Drittelnoten
Turnus:	jedes Semester
Dauer (pro Termin):	0 Semesterwochenstunden
Termine im Semester:	12 Termine
Zeit in Veranstaltungen:	0 Stunden
Sonstiger Arbeitsaufwand während Vorlesungszeit:	10 Stunden
Aufwand während Semesterferien:	10 Stunden
Flexibel einteilbarer Aufwand:	40 Stunden
Gesamtaufwand:	*Gesamtaufwand 60 Stunden weicht signifikant von erwarteten 3.0 ECTS ab.*
Lehrende:	Sergei Sawitzki

Studieninhalte:

- nach Thema der Master-Arbeit unterschiedlich
- Fachvortrag über Thema der Master-Thesis sowie über die gewählte Vorgehensweise und die Ergebnisse
- Diskussion der Qualität der gewählten Lösung
- Fragen und Diskussion zum Thema der Master-Arbeit und verwandten Gebieten

Lernergebnisse:

Die Studierenden ...

- besitzen die Fähigkeit der konzentrierten Darstellung eines intensiv bearbeiteten Fachthemas unter besonderer Berücksichtigung der Aspekte wissenschaftlichen Arbeitens
- verfestigen die Kompetenz, eine fachliche Diskussion über eine Problemlösung und deren Qualität zu führen
- verfügen über ausgeprägte Kommunikations- und Präsentationsfähigkeiten

Voraussetzungen und Empfehlungen:

Fachliche und persönliche Kompetenzen der zurückliegenden Semester, insbesondere themenabhängig fachverwandte Module und Master-Thesis

Literatur:

themenabhängig

Studiengänge:

- Betriebswirtschaftslehre Master of Science
- Data Science & Artificial Intelligence Master of Science
- E-Commerce Master of Science
- Informatik Master of Science

- IT-Sicherheit Master of Science
- IT Engineering Master of Science
- Sustainable & Digital Business Management Master of Science
- Wirtschaftsinformatik / IT-Management Master of Science
- Wirtschaftsingenieurwesen Master of Science