Hochschule
Albstadt-Sigmaringen
Albstadt-Sigmaringen University

Modulhandbuch

Institut für wissenschaftliche Weiterbildung OpenC³S

ab Sommersemester 2025
Ersteller: Kerstin Hahn
Verantwortlich: Prof. Dr. Bernd Stauß

Inhaltsverzeichnis

1 Einführung

1.1 Über das Zertifikatsprogramm

Das Zertifikatsprogramm ist Teil der wissenschaftlichen Fort- und Weiterbildungsinitiative Open C³S und steht für eine gezielte wissenschaftliche Weiterbildung im Bereich der Cyber-Sicherheit. Zwischen Oktober 2011 und März 2015, wurden in der ersten Phase des vom Bundesministerium für Bil-dung und Forschung (BMBF) geförderten Projekts, mehr als 40 in sich abgeschlossene Studienmodule zu den Themenschwerpunkten entwickelt:

- Sicherheit
- Forensik
- Recht
- praktische Informatik

Die Zertifikatsmodule sind auf wissenschaftlichem Niveau und bilden ein passgenaues Angebot an Qualifikation und Spezialisierung in der berufsbegleitenden Weiterbildung, mit hohem Praxisbezug. Nach erfolgreichem Abschluss eines Moduls erhält jeder Absolvent ein Hochschulzertifikat mit ausgewiesenen ECTS-Leistungspunkten (5 ECTS-Punkte/Modul). Die ECTS-Leistungspunkte können auf weiterführende Studiengänge der Hochschule Albstadt-Sigmaringen, wie zum Beispiel auf den Bachelor-Studiengang "IT-Sicherheit" oder den Masterstudiengang "Digitale Forensik" und andere Studienangebote (national, international, Uni oder Hochschule) angerechnet werden.

Das Zertifikatsprogramm auf einen Blick:

- Es bestehen keine formellen Zulassungsbeschränkungen.
- Die Studiendauer beträgt ca. 8 Wochen pro Modul und schließt mit einer Prüfungsleistung ab.
- Die Module haben ein hohes wissenschaftliches Niveau mit ausgeprägtem Praxisbezug.
- In einem praktischen Teil wird unter anderem der Umgang mit Werkzeugen und Beweisgrundlagen erlernt.
- Pro Modul ist ein Workload von 150 Stunden vorgesehen, davon beträgt das Selbststudium ca. 80%.
- Nach erfolgreichem Abschluss eines Moduls erhalten Sie ein Hochschulzertifikat mit ausgewiesenen ECTS-Leistungspunkten.
- Die Teilnahmegebühr für ein Einzelmodul beträgt gemäß der geltenden Gebührensatzung 2.000,- EUR.

Das Studienprogramm ist als Fernstudium mit integriertem Blended-Learning-Ansatz modular mit Studienbriefen, Präsenz- und Onlinephasen sowie Betreuung durch Online-Tutoren und Dozenten aufgebaut.

1.2 Vorwort

Dieses Dokument enthält die Beschreibungen aller Module des berufsbegleitenden Zertifikatsprogramm Open C³S und soll Ihnen einen Überblick über das aktuelle Modulangebot, sowie deren wichtigsten und charakteristischen Informationen zu Inhalt und Umfang liefern.

Ziel dieses Modulhandbuchs ist es, den Interessenten und angehenden Absolventen eine Übersicht der geforderten Leistungen sowie allen Informationen rund um die Module zu bieten. Die Übersicht auf den nachfolgenden Seiten soll Ihnen bei der Zuordnung und Orientierung helfen und dazu dienen, die Angebotsabläufe zu verstehen. Abkürzungen in diesem Dokument werden erläutert und den Oberbegriffen zugeordnet.

Im Anschluss daran finden Sie die einzelnen Module jeweils nach Universität bzw. Hochschule aufgelistet. Generelle Informationen zum Beispiel zur Veranstaltungssprache, Zeitaufwand und Vorkenntnisse können sie den detaillierten Beschreibungen entnehmen.

2 Modulbeschreibungen

2.1 Friedrich-Alexander-Universität Erlangen-Nürnberg

2.1.1 [Z-101] Methoden digitaler Forensik

<table>
<tr><td colspan="7">Modul: [Z-101] Methoden digitaler Forensik</td></tr>
<tr><td colspan="2">Kennnummer
Z-101</td><td>Workload
150 h</td><td>Modulart</td><td colspan="2">Studiensemester</td><td>Dauer
ca. 8 – 10 Wochen</td><td>Häufigkeit</td></tr>
<tr><td>1</td><td colspan="3">Lehrveranstaltung(en)</td><td>Sprache
deutsch</td><td>Kontakt-zeit
25 h</td><td>Selbst-studium
125 h</td><td>Credits (ECTS)
5</td></tr>
<tr><td>2</td><td colspan="7">Lehrform(en) / SWS:
Präsenzveranstaltung:
Vorlesung, Übung
Onlineveranstaltung:
flexible Vertiefung wichtiger Themen, Lernen im Dialog, Übung</td></tr>
<tr><td>3</td><td colspan="7">Lernergebnisse (learning outcomes), Kompetenzen:

Kompetenz Wissen
Die Studierenden können die Basisterminologie forensischer Arbeit definieren und wiedergeben. [Wissen, 7]

Kompetenz Fertigkeiten
Die Studierenden sind in der Lage, die Qualität forensischer Dokumentation zu definieren und einzuschätzen. Sie können einfache Anwendungen auf forensische Spuren hin untersuchen. [Instrumentelle Fertigkeiten, 7]

Sozialkompetenz
Die Studierenden sind in der Lage, die Fachterminologie der Forensik anzuwenden, Sachverhalte darin auszudrücken, und sich mit anderen darüber zu verständigen. [Kommunikation, 7]

Selbstständigkeit
Die Studierenden können einfache Anwendungen eigenständig auf forensische Spuren hin untersuchen und die Korrektheit ihrer Funde selbst überprüfen. [Eigenständigkeit/Verantwortung, 7]</td></tr>
<tr><td>4</td><td colspan="7">Inhalte:

• klassische (analoge) Forensik: Beispiele, Theorie der Entstehung von Spuren
• Terminologie: Identifizierung, Klassifizierung, Individualisierung, Assoziation
• Quantifizierung der Assoziation: Rechenbeispiele
• Digitale Spuren
• Kurze Einführung in die Datenträgeranalyse: Partitionssysteme (DOS, GPT)
• Regeln für den Aufbau forensischer Gutachten, Qualitätskriterien für forensische Dokumentation

• Übungen:</td></tr>
</table>

	➢ Einübung der Terminologie an Beispielen ➢ Digitale Spuren und digitale Forensik: Abgrenzung und Gemeinsamkeiten ➢ Charakteristische Spuren: Wie man sie berechnet und was sie bedeuten ➢ Analyse und Qualitätsbetrachtungen echter forensischer Berichte • **Praktische Arbeit:** Berechnung charakteristischer Spuren von einer Reihe künstlicher Anwendungen
	Empfohlene Literaturangaben: • Brian Carrier: File System Forensic Analysis. Addison-Wesley, 2005. • Eoghan Casey: Digital Evidence and Computer Crime: Forensic Science, Comput-ers, and the Internet. Academic Press, 2. Auflage, 2004. • Andreas Dewald, Felix Freiling: Forensische Informatik. Books on Demand, 2011. • Alexander Geschonneck: Computer Forensik. dpunkt Verlag, 5. Auflage, 2011. Weitere Literatur wird in der Lehrveranstaltung bekannt gegeben.
5	**Teilnahmevoraussetzungen:** Programmierkenntnisse in einer höheren Programmiersprache; Linux-Kenntnisse; Grundverständnis von Rechnerarchitektur
6	**Prüfungsformen***:* praktische Arbeit zur Analyse von Spuren einer Anwendung (Berechnung der charakteristischen Spurenmenge)
7	***Voraussetzungen für die Vergabe von Kreditpunkten:*** Bestehen der Praktischen Arbeit
8	***Verwendbarkeit des Moduls:*** Master Digitale Forensik
9	***Modulverantwortliche(r):*** Prof. Dr. Felix Freiling
10	***Optionale Informationen:***

2.1.2 [Z-102] Systemnahe Programmierung

<table>
<tr><td colspan="8">Modul: [Z-102] Systemnahe Programmierung</td></tr>
<tr><td colspan="2">Kennnummer
Z-102</td><td>Workload
150 h</td><td>Modulart</td><td colspan="2">Studiensemester</td><td>Dauer
ca. 10 Wochen</td><td>Häufigkeit</td></tr>
<tr><td>1</td><td colspan="3">Lehrveranstaltung(en)</td><td>Sprache
deutsch</td><td>Kontakt-zeit
20 h</td><td>Selbst-studium
130 h</td><td>Credits (ECTS)
5</td></tr>
<tr><td>2</td><td colspan="7">Lehrform(en) / SWS:
Präsenzveranstaltung: Vorlesung, Übung
Onlineveranstaltung: flexible Vertiefung wichtiger Themen, Lernen im Dialog. Übung</td></tr>
<tr><td>3</td><td colspan="7">Lernergebnisse (learning outcomes), Kompetenzen:

Kompetenz Wissen
Die Studierenden kennen die Einsatzszenarien der systemnahen Programmierung, und ihre Prinzipien und Methoden sind ihnen bekannt. Sie können die Grundprinzipien aktueller Rechnerarchitekturen und Betriebssysteme benennen und einordnen. Die Intel IA-32-Architektur ist ihnen im Detail vertraut. Sie sind in der Lage, Assemblerprogramme für diese Architektur zu schreiben und zu verstehen.
Ebenso sind sie in der Lage, Programme in der höheren, systemnahen Programmiersprache C zu verfassen. Den Studierenden sind die Stärken, aber auch die Schwächen - bzgl. Softwaresicherheit - der Programmiersprache C bekannt. Einige der bedeutendsten Sicherheitsprobleme/Sicherheitslücken, die insbesondere durch die Verwendung von C auf heutigen Rechnerarchitekturen entstehen können, können sie erklären. Des Weiteren können Sie übliche Gegenmaßnahmen beschreiben, die die Ausnutzung von Sicherheitslücken unterbinden sollen.
Durch eigenständiges Programmieren sind sie in der Lage, Programmierprojekte in C und Assembler umzusetzen und den Sinn sowie die Notwendigkeit effizienter Algorithmen und Datenstrukturen zu erkennen.
Die Absolventen haben fundierte Grundkenntnisse erworben, die erforderlich sind, um Maschinenprogrammanalysen zum Reverse Engineering durchzuführen. [Wissen, 6]

Kompetenz Fertigkeiten
Die Studierenden haben die Fähigkeit, systemnahe Programme zu erstellen. Hierbei finden auch wichtige Aspekte der Softwaresicherheit Anwendung. Diese Kenntnisse können die Studierenden bei der Realisierung größerer Programmieraufgaben einsetzen. [Systemische Fertigkeiten, 6]

Sozialkompetenz
Aufgrund der Teamarbeit, unter anderem am Präsenzwochenende, erweitern die Studierenden ihre Teamfähigkeit und Sozialkompetenz [Kommunikation, 6]

Selbstständigkeit
Durch das eigenverantwortliche Entwickeln von Programmen erweitern die Studierenden ihr selbstständiges Handeln. Durch das Verfassen eines Berichts wird die Selbstsicherheit der Studierenden gestärkt. [Reflexivität, 6]</td></tr>
<tr><td>4</td><td colspan="7">Inhalte:</td></tr>
</table>

	• Grundlagen Rechnerarchitektur und Assembler-Programmierung
	  o Von-Neumann-Architektur
	  o Allgemeine Prinzipien der Assemblerprogrammierung
	• Grundlagen Betriebssysteme
	  o Grundbegriffe
	  o Prozesse, Threads, Datenstrukturen
	  o Adressräume
	  o Programmierschnittstellen (API)
	• Intel x86-IA-32-Architektur und IA-32-Assembler
	(Starke Vertiefung der allgemeinen Grundlagen)
	  o Architekturmerkmale
	  o Registersatz
	  o Befehlssatz
	  o Adressierung
	  o Stack und Unterprogramm-Aufrufkonventionen
	  o Speicherverwaltung
	  o Befehlsformat
	• Die Programmiersprache C
	  o Datentypen, Operatoren und Ausdrücke
	  o Kontrollstrukturen
	  o Funktionen, Gültigkeitsbereiche und Präprozessor
	  o Zeiger und Felder
	  o Strukturen und Verbunde
	  o Standardbibliothek
	  o Inline-Assembler
	• Softwaresicherheit
	  o Buffer Overflows
	  o Gegenmaßnahmen zur Vermeidung von Buffer Overflows
	  o Gegen-Gegenmaßnahmen (z.B. Return Oriented Programming)
	• Sortieralgorithmen und Sortierbäume als Programmierprojekt
	  o Einführung und Übersicht über Sortierverfahren
	  o Einführung Sortier- und Suchbäume
	  o Programmierprojekt in Assembler und C als Hausarbeit
	• Präsenzwochenende: Vorlesung, Programmierübungen, Vorbereitung auf die Hausarbeit

	Empfohlene Literaturangaben: • Kip R. Irvine: Assembly Language for Intel-based Computer, Prentice Hall, 2010. • Brian W. Kernighan and Dennis M. Ritchie: Programmieren in C, Hanser Fachbuch, 1990. • Th. H Cormen, C.E. Leiserson, R. Rivest, C. Stein, P. Molitor: Algorithmen - Eine Einführung, Oldenbourg Wissenschaftsverlag 2004. Weitere Literatur wird in der Lehrveranstaltung bekannt gegeben.
5	**Teilnahmevoraussetzungen:** Allgemeine Programmierkenntnisse (beliebige Programmiersprache), Kenntnisse über digitale Zahlendarstellungen und Kodierungen (z.B. ASCII)

6	**Prüfungsformen**: Hausarbeit (Programmierprojekt in Assembler und C)
7	***Voraussetzungen für die Vergabe von Kreditpunkten:*** Das Einreichen der Hausarbeiten ist obligatorisch. Fristgerechte Einreichung der Hausarbeit Mind. 50 % erfolgreiche Bearbeitung der Aufgaben
8	***Verwendbarkeit des Moduls:*** Allgemeine Grundlagen in Studiengängen „Informatik“ und „IT-Sicherheit“
9	***Modulverantwortliche(r):*** Dr. rer. nat. Werner Massonne
10	***Optionale Informationen:***

2.1.3 [Z-103] Reverse Engineering

Modul: [Z-103] Reverse Engineering						
Kennnummer Z-103	**Workload** 150 h	**Modulart**	**Studiensemester**	**Dauer** ca. 10 – 12 Wochen	**Häufigkeit**	
1	**Lehrveranstaltung(en)**		**Sprache** deutsch	**Kontakt-zeit** 20 h	**Selbst-studium** 130 h	**Credits (ECTS)** 5
2	**Lehrform(en) / SWS:** Präsenzveranstaltung: Vorlesung, Übung Onlineveranstaltung: flexible Vertiefung wichtiger Themen, Lernen im Dialog, Übung					
3	**Lernergebnisse (learning outcomes), Kompetenzen:** *Kompetenz Wissen* Die Studierenden können den Begriff „Reverse Engineering“ einordnen und definieren. Sie können die typischen Einsatzgebiete von Reverse Engineering benennen. Die Studierenden haben fundierte Kenntnisse in der Programmierung von IA-32 auf Maschinenebene. Die Strukturen von Microsoft Windows sind ihnen bekannt. Den Aufbau von Programmdateien in Windows können sie beschreiben und analysieren. Sie können die Methoden zur Dekompilierung von Maschinenprogrammen benennen und anwenden. Verschiedene Optimierungsverfahren der Compiler, die eine Dekompilierung erschweren, können sie erkennen und benennen. Die üblichsten Werkzeuge zur Programmanalyse können die Absolventen einsetzen, Vorteile und Nachteile einer statischen und dynamischen Programmanalyse sind ihnen bekannt, und sie können diese bedarfsabhängig einsetzen. Sie haben vertiefte Kenntnisse über Malware sowie verschiedene Methoden und Tricks der Malware-Autoren. Die Absolventen können „einfache“ Malware für Windows-Systeme selbstständig analysieren. Sie beherrschen die Grundlagen für eine Vertiefung des weiten Gebietes der Malware-Analyse. [Wissen, 7]					

	Kompetenz Fertigkeiten Die Studierenden haben die Fähigkeit, systemnahe Programme zu erstellen und zu verstehen. Diese Kenntnisse können die Studierenden bei der Analyse unbekannter Malware Binaries einsetzen. [Systemische Fertigkeiten, 7]
	Sozialkompetenz Aufgrund der Teamarbeit, unter anderem am Präsenzwochenende, erweitern die Studierenden ihre Teamfähigkeit und Sozialkompetenz. [Kommunikation, 7]
	Selbstständigkeit Durch das eigenverantwortliche Entwickeln von Programmen und die Programmanalyse erweitern die Studierenden ihr selbstständiges Handeln. Durch das Verfassen eines Berichts wird die Selbstsicherheit der Studierenden gestärkt. [Reflexivität, 7]
4	**Inhalte:** • Einführung in Reverse Engineering o Abgrenzung des Begriffs Reverse Engineering o Einsatzgebiete o Zielsetzung und Grenzen von Reverse Engineering • Intel x86-IA-32-Architektur und IA-32-Assembler o Architekturmerkmale o Registersatz o Befehlssatz o Adressierung o Stack und Unterprogramm-Aufrufkonventionen o Speicherverwaltung o Befehlsformat • Microsoft Windows o Aufbau und Struktur o Anwendungen und Bibliotheken, API-Programmierung o Detaillierte Betrachtung der PE-Struktur zur Programmanalyse: Importe, Exporte, Sections, Windows-Loader, Datenstrukturen o Prozesse, Threads und ihre Datenstrukturen o Exceptions und Exception-Behandlung • Programmanalyse o Codeerzeugung durch Compiler und Dekompilierung o Optimierungsverfahren • Werkzeuge zur Programmanalyse, Schwerpunkt IDA o Statische Analyse o Dynamische Analyse • Malware und Malware-Analyse

	○ Obfuscation ○ Verhinderung von Disassemblierung ○ Malware-Techniken, Packer, Anti-Reverse-Engineering-Methoden ○ Analyse realer Malware in einer virtuellen Analyseumgebung • Präsenzwochenende: Vorlesung, Übungen in Gruppen: Analyse verschleierter Binaries, Analyse von Malware, Vorbereitung auf die Hausarbeit
	Empfohlene Literaturangaben: • Eldad Eilam: Reversing: Secrets of Reverse Engineering, John Wiley & Sons, 2005 • Michael Sikorski and Andrew Honig. Practical Malware Analysis: The Hands-On Guide to Dissecting Malicious Software. No Starch Press, 2012. Weitere Literatur wird in der Lehrveranstaltung bekannt gegeben.
5	**Teilnahmevoraussetzungen:** Grundkenntnisse im Bereich Betriebssysteme sowie im Bereich Rechnerarchitektur und Assemblerprogrammierung (plattformunabhängig), Programmierkenntnisse insbesondere in der Programmiersprache C.
6	**Prüfungsformen***:* Hausarbeit
7	***Voraussetzungen für die Vergabe von Kreditpunkten:*** Das Einreichen der Hausarbeiten ist obligatorisch. Fristgerechte Einreichung der Hausarbeit Mind. 50 % erfolgreiche Bearbeitung der Aufgaben
8	***Verwendbarkeit des Moduls:*** Anerkennung in den Studiengängen Master Digitale Forensik (Hochschule Albstadt-Sigmaringen) und Bachelor Informatik/IT-Sicherheit (FAU) als Pflichtmodul
9	***Modulverantwortliche(r):*** Dr. rer. nat. Werner Massonne
10	***Optionale Informationen:***

2.2 Hochschule Albstadt-Sigmaringen

2.2.1 [Z-202] Programmieren im IT-Security-Umfeld mit Python

Modul: [Z-202] Programmieren im IT-Security-Umfeld mit Python 1						
Kennnummer Z-202	**Workload** 150 h	**Modulart**	**Studiensemester**	**Dauer** ca. 10 Wochen	**Häufigkeit**	
1	**Lehrveranstaltung(en)**	**Sprache** deutsch	**Kontakt-zeit** 25 h	**Selbst-studium** 125 h	**Credits (ECTS)** 5	
2	**Lehrform(en) / SWS:** Präsenzveranstaltung: Vorlesung, Übung Onlineveranstaltung: Vorlesung, flexible Vertiefung wichtiger Themen, Lernen im Dialog, Übung					
3	**Lernergebnisse (learning outcomes), Kompetenzen:** *Kompetenz Wissen* Die Teilnehmer können aus einer abstrakten Aufgabenstellung ein ablauffähiges Programm entwickeln. Wenn die Programmierung konkret wird, so findet die Programmiersprache Python Verwendung. Python ist eine leistungsfähige Skriptsprache, die im Forensik Umfeld häufig verwendet wird. Die Grundkonstrukte von Programmen und deren Umsetzung in Python wurde erlernt. Die Studierenden haben erste Erfahrungen mit programmbasierten Sicherheitsschwachstellen und verstehen einfache Angriffsmechanismen. *Kompetenz Fertigkeiten* Die Teilnehmer können mit den selbst erstellten Programmen häufig in der Praxis vorkommende Aufgabenstellungen bewältigen wie z. B. das Durchsuchen eines Rechners nach auffälligen Bildern (Zuwachs an Methodenkompetenz). *Sozialkompetenz* *Selbstständigkeit* Dieses Modul fördert die Selbstkompetenz durch das unterstützte Selbstlernen bei den praktischen Aufgabenstellungen in besonderem Maße (Erarbeitung von Lösungen in einem festgelegten Zeitrahmen, Hilfe holen bei Bedarf, Erkenntnisgewinn aus korrigierter Lösung).					
4	**Inhalte:** In diesem Modul werden die Kenntnisse in Informatik und Programmieren vermittelt, die ein IT-Sicherheitsexperte braucht, um für ein Rechnersystem spezifische Programme zur Analyse des IT-Sicherheitsstands vornehmen zu können sowie um sicherheitsrelevante Vorgänge überprüfen zu können. Damit ist auch die Grundlage für einen guten Einstieg zum Erlernen weiterer Programmiersprachen gelegt.					

	1. Einführung in Python Syntax und Semantik, Programmierparadigmen, Installation, Interaktiver Modus, objektorientiertes Programmieren , Funktionen, Methoden, Standard-Datentypen, Erstellen von Skriptdateien, Kontrollstrukturen, Definition eigener Klassen, guter Programmierstil **Praktische Übung:** Erstellen eines Programms, welches Dateien sucht und diese anhand des Dateityps kategorisch sortiert. In einer Textdatei werden die Informationen über die Dateien festgehalten. 2. Forensische Analyse mit Python: Datenbanken und Anwendungen, Grundlagen Datenbanken, SQL-Syntax, sqlite3-Modul in Python, Untersuchen von Anwendungs-Artefakten an den Beispielen Skype, Firefox und Chrome **Praktische Übung:** Ergänzung und Optimierung der praktischen Übung aus SB1, Textdateien durch Datenbankeinträge ersetzen, Suchanfragen über sqlite3 realisieren; Extraktion von Anwendungsdaten aus Skype und Firefox 3. Forensische Analyse mit Python: Windows Auslesen der Windows-Registry bei einem Live-System, Analyse der Hive-Dateien (Post Mortem), Entschlüsselung von WLAN-Kennwörtern, Wiederherstellung von gelöschten Daten, Analyse von Metadaten **Praktische Übung:** String-Suche in Hive-Dateien, Wiederherstellung von WLAN-Passwörtern, Metadaten von Bildern auswerten
	Empfohlene Literaturangaben: • Ernesti, Johannes; Kaiser, Peter (2012): Python 3: Das umfassende Handbuch. 3. Aufl. Bonn: Galileo Press GmbH. • Weigend, Michael (2009): OOP mit Python 3; PR. 4. Aufl. München: Hüthig Jehle Rehm. • O'Connor, TJ (2012): Violent Python. A Cookbook for Hackers, Forensic Analysts, Penetration Testers and Security Engineers. London (Newnes). Weitere Literatur wird in der Lehrveranstaltung bekannt gegeben.
5	**Teilnahmevoraussetzungen:** keine
6	**Prüfungsformen***:* Klausur, Hausarbeit
7	***Voraussetzungen für die Vergabe von Kreditpunkten:*** Bestehen der Prüfungen
8	***Verwendbarkeit des Moduls:***
9	***Modulverantwortliche(r):*** Prof. Dr. Martin Rieger
10	***Optionale Informationen:***

2.2.2 [Z-203] Penetration Testing mit Python

Version	Erstellt/geändert von/am	Dokument	Freigabe am/von	Gültig ab
	KH, 19.02.2025	Modulhandbuch_OC3S_neu_ENTWURF	MS, 21.02.2025	SoSe2025

Modul: [Z-203] Penetration Testing mit Python					
Kennnummer Z-203	**Workload** 150 h	**Modulart**	**Studiensemester**	**Dauer** ca. 10 Wochen	**Häufigkeit**

1	**Lehrveranstaltung(en)**	**Sprache** deutsch	**Kontakt-zeit** 25 h	**Selbst-studium** 125 h	**Credits (ECTS)** 5

2 Lehrform(en) / SWS:
Präsenzveranstaltung:
Vorlesung, Übung
Onlineveranstaltung:
Vorlesung, flexible Vertiefung wichtiger Themen, Lernen im Dialog, Übung

3 Lernergebnisse (learning outcomes), Kompetenzen:

Kompetenz Wissen

Python als Basis von Penetrationstests [Wissen, 6]

Kompetenz Fertigkeiten

Durchführen von Pentests an Beispielen mit Python-Programmen [Instrumentelle Fertigkeiten, 6]

Sozialkompetenz

Teamarbeit

Selbstständigkeit

Eigenständiges Bearbeiten von Aufgabenblöcken in Verbindung mit anderen Teammitgliedern [Eigenständigkeit/Verantwortung, 6]

4 Inhalte:
In diesem Modul werden die Kenntnisse vertieft, die ein IT-Sicherheitsexperte benötigt, um den Datenverkehr im Netzwerk zu analysieren oder Schwachstellen durch gezielte Manipulationen aufzudecken. Durch das Aufzeigen von antiforensischen Maßnahmen und das Realisieren von Angriffsszenarien tritt zudem eine Sensibilisierung für das Thema IT-Sicherheit ein.

1. Netzwerkforensik mit Python
   Physikalischer Standort von IP-Adressen ermitteln und visualisieren, Datenpakete und pcap-Dateien parsen, Sniffing
   Praktische Übung:
   String-Suche in Datenpaketen und pcap-Dateien

2. Penetrationstest mit Python
   Internet Wide Scans, Port Scanning, FTP Scanner, SSH-Angriff, DDoS-Angriff, Paket-Injection, Session Hijacking
   Praktische Übung:
   Angreifen eines SSH Honey Pots, Shellshock

3. Python-Hacks
   Erstellen eines Proxys, Proxy-Test-Bot, Python-gestützte E-Mail-Kommunikation, Python-gestütztes

	Webbrowsing, Implementierung von Ransomware **Praktische Übung:** SMTP-Server angreifen und für das Versenden von Spam-Mail missbrauchen.
	Empfohlene Literaturangaben: • Barry, P. (2024). Python von Kopf bis Fuß: Grundlagen und Praxis der Python-Programmierung (3. Aufl.). O'Reilly (2024). • Python lernen – kurz & gut. dpunkt verlag • O'Connor, TJ (2012): Violent Python. A Cookbook for Hackers, Forensic Analysts, Penetration Testers and Security Engineers. London (Newnes). • Justin Seitz (2024),Hacking mit Python , dpunkt verlag
5	**Teilnahmevoraussetzungen:** Kenntnisse aus dem Modul „Python 1 – Programmierung und Forensik (Z-203)“ oder fortgeschrittene Programmierkenntnisse Empfohlen: Kenntnisse über Netzwerkprotokolle und Internettechnologien
6	**Prüfungsformen***:* Klausur, Hausarbeit
7	***Voraussetzungen für die Vergabe von Kreditpunkten:*** Das Einreichen der Übungsaufgaben ist obligatorisch für die Teilnahme an der Klausur. Um am Ende des Moduls ein Hochschulzertifikat mit ausgewiesenen ECTS-Leistungspunkten zu erlangen, müssen folgende Erfordernisse erbracht werden: - Einreichung aller Übungsaufgaben, mind. 50% erfolgreiche Bearbeitung der Aufgaben (Prüfungsleistung: Hausarbeit) - Prüfungsleistung Klausur bestanden Das Modul ist bestanden, wenn alle erforderlichen Leistungen erbracht wurden.
8	***Verwendbarkeit des Moduls:*** Master Digitale Forensik
9	***Modulverantwortliche(r):*** Prof. Dr. Martin Rieger
10	***Optionale Informationen:***

2.2.3 [Z-208] Forensische Analyse eines Windows-Systems

Modul: [Z-208] Forensische Analyse eines Windows-Systems						
Kennnummer Z-208	**Workload** 150 h	**Modulart**	**Studiensemester**	**Dauer** ca. 14 Wochen		**Häufigkeit**
1	**Lehrveranstaltung(en)**		**Sprache** deutsch	**Kontakt-zeit** 25 h	**Selbst-studium** 125 h	**Credits (ECTS)** 5
2	**Lehrform(en) / SWS:** Präsenzveranstaltung: Vorlesung, Übungen Onlineveranstaltung: Vorlesung, flexible Vertiefung wichtiger Themen, Lernen im Dialog, Übung					
3	**Lernergebnisse (learning outcomes), Kompetenzen:** *Kompetenz Wissen* Mögliche Nutzerspuren sind im Windows-System kennen [Wissen, 6] *Kompetenz Fertigkeiten* Nutzerspuren im Windows-System bergen, analysieren und beurteilen *[Instrumentelle Fertigkeiten, 6]* *Sozialkompetenz* *Asservat im Team bearbeiten* [Team-/Führungsfähigkeit, 6] *Selbstständigkeit* Überlegungen zur Plausibilität der Einzuspuren im Gesamtbild [Reflexivität]						
4	**Inhalte:** ▪ **Das Windows-Rechnersystem** Grundlegende Konzepte und Begriffe, Windows-„Bordwerkzeuge“ (Untersuchung von Prozessen und Threads, Leistungsüberwachung), System-Architektur (Gerätetreiber, Systemprozesse, Kernel, HAL), Sicherheitskomponenten und Rechtesystem, Reguläre Ausdrücke, Grundlagen der (Windows-) Netzwerktechnik, Ermitteln der Netzwerkeigenschaften des Rechners ▪ **Spezifische Strukturen und Analysemethoden zu Windows-Systemen** forensisch relevante Verzeichnisse und Dateien, Schattenkopien, Speicherabbilder gewinnen und auswerten, Protokolldateien gewinnen, Windows-Zugriffsrechte analysieren und verändern, Schlüsselwortsuche, Filecarving, Schlupfspeicher extrahieren, indizieren von Metadaten, Forensische Arbeitsweise im Windows-System ▪ **Forensische Erkenntnisse aus der Registry** Aufbau, SIDs, SAMs, GUID Forensisch relevante Registry-Einträge,						

	Werkzeuge zur Registry-Analyse Antiforensische Maßnahmen ▪ **Logfile-Analyse** NTFS-Journal-Protokollierung, Struktur der Logging-Einträge, Auswertung, Windows-Event-Log, Anwendungs- und Dienstprotokolle, Security-Log, Setup-Log, Überwachungsrichtlinien Antiforensische Maßnahmen ▪ **Forensische Untersuchung von Internetdiensten** Peer-to-Peer-Aktivitäten aufdecken, Skype-Accounts untersuchen Client-Datenbanksystem, SQLite-Anwendungs-Artefakte auswerten (Skype, Firefox, Chrome), Microsoft-Anwendungs-Artefakte auswerten (Internet Explorer, Edge, Outlook) ▪ **Forensische Analyse von Arbeitsspeicher und Windows-Live-Artefakten** Flüchtige Informationen ermitteln, Systemzeit auslesen, eingeloggte Benutzer, offene Dateien, Netzwerkverbindungen, Prozessinformationen, Zwischenablage, Dienste/Treiber-Informationen, Erstellung eines Arbeitsspeicherabbilds, Arbeitsspeicheranalyse mit dem Volatility Artefakt-analyse ▪ **Forensische Fallbeispiele** o <u>Analyse eines Rechners mit Malware-Befall:</u> Indicators of Compromise, Schrittweise Analyse mit automatischen und händischen Methoden, Bereinigung des Rechnersystems o <u>Der Fall Evil Knievel:</u> Forensische Untersuchung auf Illegalen Handel, Auswertung vieler Windowsspezifischer Artefakte, Umgehen mit antiforensischen Maßnahmen
	Empfohlene Literaturangaben: Literatur wird in der Lehrveranstaltung bekannt gegeben.
5	**Teilnahmevoraussetzungen:** keine Empfohlen: Kenntnisse im Umgang mit Rechnern, dem Internet und dem Windows-Betriebssystem
6	**Prüfungsformen***:* Klausur, Hausarbeit
7	***Voraussetzungen für die Vergabe von Kreditpunkten:*** Das Einreichen der Übungsaufgaben ist obligatorisch für die Teilnahme an der Klausur. Um am Ende des Moduls ein Hochschulzertifikat mit ausgewiesenen ECTS-Leistungspunkten zu erlangen, müssen folgende Erfordernisse erbracht werden: - Einreichung aller Übungsaufgaben, mind. 50% erfolgreiche Bearbeitung der Aufgaben (Prüfungsleistung: Hausarbeit)

	- Prüfungsleistung Klausur bestanden Das Modul ist bestanden, wenn alle erforderlichen Leistungen erbracht wurden.
8	***Verwendbarkeit des Moduls:*** Master Digitale Forensik
9	***Modulverantwortliche(r):*** Prof. Dr. Martin Rieger
10	***Optionale Informationen:***

2.2.4 [Z-209] Forensische Analyse eines Unix-Systems

Modul: [Z-209] Forensische Analyse eines Unix-Systems						
Kennnummer Z-209	**Workload** 150 h	**Modulart**	**Studiensemester**	**Dauer** ca. 14 Wochen	**Häufigkeit**	
1	**Lehrveranstaltung(en)**		**Sprache** deutsch	**Kontakt-zeit** 25 h	**Selbst-studium** 125 h	**Credits (ECTS)** 5
2	**Lehrform(en) / SWS:** Präsenzveranstaltung: Vorlesung, Übungen Onlineveranstaltung: Vorlesung, flexible Vertiefung wichtiger Themen, Lernen im Dialog, Übung					
3	**Lernergebnisse (learning outcomes), Kompetenzen:** *Kompetenz Wissen* Mögliche Nutzerspuren sind im Unix-System kennen [Wissen, 6] *Kompetenz Fertigkeiten* Nutzerspuren im Unix-System bergen, analysieren und beurteilen [Instrumentelle Fertigkeiten, 6] *Sozialkompetenz* Asservat im Team bearbeiten [Team-/Führungsfähigkeit, 6] *Selbstständigkeit* Überlegungen zur Plausibilität der Einzuspuren im Gesamtbild [Reflexivität, 6]						
4	**Inhalte*:*** In diesem Modul werden Ihnen verschiedene Aspekte des Betriebssystems Unix bzw. Linux vermittelt, die es Ihnen ermöglichen Untersuchungen forensischer Art oder zur IT-Sicherheit an den genannten Betriebssystemen durchzuführen. Grundlage hierfür ist das Verständnis über wichtige Konzepte und Eigenschaften von Unix bzw. Linux. In den einzelnen Studienbriefen werden verschiedene Bereiche unixoide Betriebssysteme betrachtet und analysiert.						

- Das Unix-Rechnersystem
  Grundlegende Begrifflichkeiten, Unix-Varianten, Umgang mit der grafischen Oberfläche und dem Terminal, Dateisysteme, Verzeichnisstruktur, Passwort- und Schattendatei, Zugriffsrechte und Zugriffskontrolle
  Erstellung von und Arbeiten mit Bash-Skripten
  Erstellung von und Arbeiten mit Python-Skripten

- Struktur und Analyse von unixoiden Systemen
  Hardwareinformationen, Systemprozesse und Leistungsüberwachung, Untersuchungen an Prozessen und Threads, Systeminformationen, Dienste, reguläre Ausdrücke, Suchprogramme, Untersuchung zu Rootkits, Nachweis von Rootkits

- Logfile-Analyse:
  Rsyslog-Daemon-Analyse und Konfiguration,
  Logfile-Analyse mit Bordmitteln, mit petit und logwatch,
  Auswertung von Benutzeranmeldungen und Anmeldeversuchen, von USB-Benutzung, von WLAN-Anmeldung, von SW-Installation

- Live-Analyse
  Flüchtige Informationen ermitteln, Systemzeit auslesen, eingeloggte Benutzer, offene Dateien, Netzwerkverbindungen, Prozessinformationen, Dienste/Treiber-Informationen

- Forensische Analyse von Arbeitsspeichern:
  Erstellen eines Arbeitsspeicherabbilds,
  Möglichkeiten der Erfassung, Erstellen von Profilen für das Volatility Framework,
  Analyse mit dem Volatility Framework

- Linux Serverdienste:
  Installation von Apache2 Webserver und Wordpress,
  Cyber-Angriffe auf Wordpress und dessen Nachweis,
  Datengewinnung aus MySQL-Datenbanken,
  Auswertung von E-Mails, Mailservern,
  Auswertung von Routern, Firewalls und Netzwerkkomponenten

- Fallbeispiele:
  - Bau, Suche und Nachweis eine Rootkits
  - Analyse eines Linux-Livesystems mittels eigener Skripte
  - Möglichkeiten von Zeitgeist/Zeitgeist-Explorer an Beispielen
  - Forensische Arbeitsspeicheranalyse eines kompromittierten Servers
  - Fallbeispiel (NAS) QNAP Asservat

Empfohlene Literaturangaben:

	• Herold, H; Lurz, B; Wohlrab, J.: Grundlagen der Informatik. München; Boston [u.a.]: Pearson Studium. • Tanenbaum, A. S. (2006): Computerarchitektur : Strukturen - Konzepte – Grundlagen. München; [Boston {u.a.]: Pearson Studium • Brian Ward: How Linux Works: What Every Superuser Should Know No Starch Press; Auflage: 2 (11. November 2014) • Michael Kofler: Linux: Das umfassende Handbuch. Rheinwerk Computing; Auflage: 14 (30. November 2015) • Nemeth, Evi, Snyder, Garth, Hein, Trent R., Whaley, Ben: UNIX and Linux System Administration Handbook Prentice Hall 4th Edition (2011) • Dr. Philip Polstra: Linux Forensics CreateSpace Independent Publishing Platform; Auflage: 1 (13. Juli 2015). Weitere Literatur wird in der Lehrveranstaltung bekannt gegeben.
5	**Teilnahmevoraussetzungen:** Vertraut mit Unix-Benutzersicht, Kenntnisse des forensischen Arbeitens Empfohlen: Kenntnisse im Umgang mit Rechnern, dem Internet und dem Unix-Betriebssystem
6	**Prüfungsformen***:* Klausur, Hausarbeit
7	***Voraussetzungen für die Vergabe von Kreditpunkten:*** Das Einreichen der Übungsaufgaben ist obligatorisch für die Teilnahme an der Klausur. Um am Ende des Moduls ein Hochschulzertifikat mit ausgewiesenen ECTS-Leistungspunkten zu erlangen, müssen folgende Erfordernisse erbracht werden: - Einreichung aller Übungsaufgaben, mind. 50% erfolgreiche Bearbeitung der Aufgaben (Prüfungsleistung: Hausarbeit) - Prüfungsleistung Klausur bestanden Das Modul ist bestanden, wenn alle erforderlichen Leistungen erbracht wurden.
8	***Verwendbarkeit des Moduls:*** Master Digitale Forensik
9	***Modulverantwortliche(r):*** Prof. Dr. Martin Rieger
10	***Optionale Informationen:***

2.2.5 [Z-210] Forensische Analyse eines MacOS-Systems

Modul: [Z-210] Forensische Analyse eines MacOS-Systems						
Kennnummer Z-210	**Workload** 150 h	**Modulart**	**Studiensemester**	**Dauer** ca. 14 Wochen	**Häufigkeit**	
1	**Lehrveranstaltung(en)**		**Sprache** deutsch	**Kontakt-zeit** 25 h	**Selbst-studium** 125 h	**Credits (ECTS)** 5
2	**Lehrform(en) / SWS:**					
3	**Lernergebnisse (learning outcomes), Kompetenzen:** *Kompetenz Wissen* Mögliche Nutzerspuren sind im MacOS-System kennen [Wissen, 6] *Kompetenz Fertigkeiten* Nutzerspuren im MacOS-System bergen, analysieren und beurteilen [Instrumentelle Fertigkeiten, 6] *Sozialkompetenz* Asservat im Team bearbeiten *[Team-/Führungsfähigkeit, 6]* *Selbstständigkeit* Überlegungen zur Plausibilität der Einzuspuren im Gesamtbild [Reflexivität, 6]						
4	**Inhalte*:*** In diesem Modul werden Ihnen verschiedene Aspekte des Betriebssystems MacOS vermittelt, die es Ihnen ermöglichen Untersuchungen forensischer Art oder zur IT-Sicherheit an den genannten Betriebssystemen durchzuführen. Eine Grundlage hierfür ist das Verständnis über wichtige Konzepte und Eigenschaften von Linux, FreeBSD und vor allem den MacOS-spezifischen Komponenten. In den einzelnen Studienbriefen werden verschiedene Bereiche des MacOS-Betriebssystems betrachtet und analysiert. ▪ **Apple-Hardware** Apple Inc. Firmengeschichte, Desktop und mobile Computer, Set Top Boxen, Server, tragbare Geräte, Software, Hardwarearchitekturen (Litte/Big Endian), Datenübertragungsschnittstellen ▪ **MacOS-Betriebssystem** Klassisches Mac OS/System 7, OPENSTEP, macOS Versionen (Client), macOS Versionen (Server), MacOSX, Betriebssystemarchitektur (Kernel, Userland), Property Lists, AppleScript, Sicherheitsfunktionen (Sandboxing, XPC, Gatekeeper, Xprotect, Fileattributes) Dateisysteme HFS+, APFS Forensisch bedeutsame Artefakte, die durch MacOS und das Dateissystem erzeugt werden.					

	▪ **Persistente Spuren:** Die MacOS Verzeichnisstruktur (Benutzer, Library, Spotlight, Netzwerkkonfiguration, Drucker, Repositoryverwaltungen), Nutzerdomäne (Benutzerverwaltung, Applikationsstruktur, Zeitstempel, Apple Applikationen, Papierkorb, Backups, Virtualisierung, Zuletzt verwendete Objekte) Spezifische Formate und deren Auswertung Forensische Analyse der persistenten Spuren an Beispielen ▪ **Netzwerkbasierte Dienste:** iCloud Services, Mobile Device Management, Synchronisation iCloud und lokale Verzeichnisse Netzwerkdienste in der lokalen Domäne (AFP, SMB, VNC, FTP, SSH, ARD, Webserver) ▪ **Methoden digitaler Forensik im MacOS-Umfeld:** Investigativer Prozess nach Casey MacOS Sicherungsvorgehen, Liveanalyse, Post Mortem Analyse (Disk Arbitration, Verschlüsselung, Livesystem) ▪ **Nichtpersistente Spuren und Forensische Analyse von Arbeitsspeichern:** Einführung in die RAM-Analyse, Struktur des Arbeitsspeichers, Erstellen eines Arbeitsspeicherabbilds, Möglichkeiten der Erfassung, Erstellen von Profilen für das Volatility Framework, ▪ Fallbeispiele: Analyse eines MacOS-Livesystems mittels eigener Skripte Forensische Arbeitsspeicheranalyse eines kompromittierten Clients Übung Fallbeispiel Datenträger: Forensische Untersuchung auf Illegalen Handel, Auswertung vieler MacOS-spezifischer Artefakte, Umgehen mit antiforensischen Maßnahmen
	Empfohlene Literaturangaben: • Jonathan Levin. Mac OS X and IOS Internals: To the Apple's Core. John Wiley & Sons • Topher Kessler. EFI firmware protection locks down newer Macs. Website,. https://www.cnet. com/news/efi- firmware- protection- locks- down- newer- macs/. • Maximillian Dornseif. Vorlesung Computerforensik. Friedrich-Alexander Universität Erlangen-Nürnberg, • Brian Ward: How Linux Works: What Every Superuser Should Know No Starch Press; Auflage: 2 (11. November 2014) • Marc Brandt. Forensische Analyse von Mac OS X. Hochschule Albstadt-Sigmaringen, 2016. • Brian Carrier. File system forensic analysis. Addison-Wesley Pro-fessional, 2005. • Eoghan Casey. Handbook of digital forensics and investigation. Academic Press, 2009. • A. Singh. Mac OS X Internals: A Systems Approach. Pearson Edu-cation, 2006. ISBN 9780132702263. URL https://books.google.co.il/books?id=K8vUkpOXhN4C. • Jesse Varsalone. Mac OS X, iPod, and iPhone forensic analysis DVD toolkit. Syngress, 2008. Weitere Literatur wird in der Lehrveranstaltung bekannt gegeben.
5	**Teilnahmevoraussetzungen:**

	keine Empfohlen: Kenntnisse im Umgang mit Rechnern, dem Internet und dem macOS-Betriebssystem (mit der Nutzersicht vertraut)
6	**Prüfungsformen***:* Klausur, Hausarbeit
7	***Voraussetzungen für die Vergabe von Kreditpunkten:*** Das Einreichen der Übungsaufgaben ist obligatorisch für die Teilnahme an der Klausur. Um am Ende des Moduls ein Hochschulzertifikat mit ausgewiesenen ECTS-Leistungspunkten zu erlangen, müssen folgende Erfordernisse erbracht werden: - Einreichung aller Übungsaufgaben, mind. 50% erfolgreiche Bearbeitung der Aufgaben (Prüfungsleistung: Hausarbeit) - Prüfungsleistung Klausur bestanden Das Modul ist bestanden, wenn alle erforderlichen Leistungen erbracht wurden
8	***Verwendbarkeit des Moduls:*** Master Digitale Forensik
9	***Modulverantwortliche(r):*** Prof. Dr. Martin Rieger
10	***Optionale Informationen:***

2.2.6 [Z-211] Forensische Analyse von Netzwerken

<table>
<tr><td colspan="7">Modul: [Z-211] Forensische Analyse von Netzwerken</td></tr>
<tr><td colspan="2">Kennnummer
Z-211</td><td>Workload
150 h</td><td>Modulart</td><td colspan="2">Studiensemester</td><td>Dauer
ca. 14 Wochen</td><td>Häufigkeit</td></tr>
<tr><td>1</td><td colspan="3">Lehrveranstaltung(en)</td><td>Sprache
deutsch</td><td>Kontakt-zeit
25 h</td><td>Selbst-studium
125 h</td><td>Credits (ECTS)
5</td></tr>
<tr><td>2</td><td colspan="7">Lehrform(en) / SWS:
Präsenzveranstaltung:
Vorlesung, Übungen
Onlineveranstaltung:
Vorlesung, flexible Vertiefung wichtiger Themen, Lernen im Dialog, Übung</td></tr>
<tr><td>3</td><td colspan="7">Lernergebnisse (learning outcomes), Kompetenzen:

Kompetenz Wissen
Mögliche Nutzerspuren sind im Netzwerk kennen [Wissen, 6]
Kompetenz Fertigkeiten
Nutzerspuren im Netzwerk bergen, analysieren und beurteilen [Instrumentelle Fertigkeiten, 6]
Sozialkompetenz
Asservat im Team bearbeiten [Team-/Führungsfähigkeit, 6]
Selbstständigkeit /Kompetenzausprägung wählen
Überlegungen zur Plausibilität der Einzuspuren im Gesamtbild [Reflexivität, 6]</td></tr>
<tr><td>4</td><td colspan="7">Inhalte:

Grundlagen der Netzwerkforensik
▪ Forensische Untersuchungen an Rechnern in Netzwerken
▪ Datengewinnung aus aktiven Netzkomponenten
▪ Datengewinnung aus dem Netzwerkdatenstrom mittels Netzwerk-Sniffer
▪ IT-Strukturen
▪ Network Security Monitoring (NSM) Vorgehensmodell

Post Mortem-Analyse von Server-Diensten und -Komponenten
▪ Analyse von Logdateien
▪ Sicherung und Analyse von Serverdiensten
▪ Analyse Microsoft Serverdiensten
▪ Forensische Auswertung von Routern und anderen Netzwerkkomponenten

Live-Analyse von Server-Diensten und -Komponenten
▪ Aufbereitung von Server-Sicherungen zur Virtualisierung
▪ Live-Analyse laufender Systeme am Beispiel</td></tr>
</table>

	▪ Erstellung und Analyse eines Arbeitsspeicherabbildes ▪ Analyse von IoT-Systemen ▪ Analyse von Schadsoftware **Internet- und Cloudforensik** ▪ Internet- und Cloudspuren im Client ▪ Geräte im Netzwerk erkennen ▪ Sichern von Clouddaten am Beispiel Facebook ▪ Kryptowährungen **Forensische Fallbeispiele** ▪ DoS-Angriff auf virtuelle Netzwerkumgebung: Auswertung der Spuren ergibt das Schadensbild Spuren zum Tathergang und zu den Tätern ▪ ARP-Spoofing- Angriff in virtueller Netzwerkumgebung analysieren: Auswertung der Spuren ergibt das Schadensbild Spuren zum Tathergang und zu den Tätern ▪ Browser-Analyse und Facebook-Analyse: Nachweis von illegalem Handel einer Bande
	Empfohlene Literaturangaben: • Andrew Tanenbaum (2020): Computernetzwerke. Verlag Pearson Studium; 5. Auflage: • Claudia Eckert (2016): IT-Sicherheit Strukturen - Konzepte – Grundlagen. Verlag De Gruyter Oldenbourg, 9. Auflage.) • Jörg Schwenk (2014): Sicherheit und Kryptographie im Internet. Verlag: Springer Vieweg, 4.Auflage: • S. Davidoff, J. Ham (2012): Network Forensics. Verlag Prentice Hall International. • Chris Sanders, Jason Smith (2013): Applied Network Security Monitoring: Collection, Detection, and Analysis. Verlag: Syngress. Weitere Literatur wird in der Lehrveranstaltung bekannt gegeben.
5	**Teilnahmevoraussetzungen:** Kenntnisse in TCP/IP-Rechnernetzen
6	**Prüfungsformen***:* Klausur, Hausarbeit
7	***Voraussetzungen für die Vergabe von Kreditpunkten:*** Das Einreichen der Übungsaufgaben ist obligatorisch für die Teilnahme an der Klausur. Um am Ende des Moduls ein Hochschulzertifikat mit ausgewiesenen ECTS-Leistungspunkten zu erlangen, müssen folgende Erfordernisse erbracht werden: - Einreichung aller Übungsaufgaben, mind. 50% erfolgreiche Bearbeitung der Aufgaben (Prüfungsleistung: Hausarbeit) - Prüfungsleistung Klausur bestanden Das Modul ist bestanden, wenn alle erforderlichen Leistungen erbracht wurden.
8	***Verwendbarkeit des Moduls:*** Digitale Forensik
9	***Modulverantwortliche(r):***

	Prof. Dr. Martin Rieger
10	***Optionale Informationen:***

2.2.7 [Z-213] Incident Response & Malware Defence

Modul: [Z-213] Incident Response & Malware Defence						
Kennnummer Z-213	**Workload** 150 h	**Modulart**	**Studiensemester**		**Dauer**	**Häufigkeit**

1	**Lehrveranstaltung(en)**	**Sprache** deutsch	**Kontakt-zeit** 25 h	**Selbst-studium** 125 h	**Credits (ECTS)** 5
2	**Lehrform(en) / SWS:** Präsenzveranstaltung: Vorlesung, Übungen Onlineveranstaltung: Vorlesung, flexible Vertiefung wichtiger Themen, Lernen im Dialog, Übung				
3	**Lernergebnisse (learning outcomes), Kompetenzen:** *Kompetenz Wissen* Kenntnisse um die verschiedenen Wirkungen [Wissen, 6] *Kompetenz Fertigkeiten* Detektion, Analyse von Angriffen; Eindämmen von Angriffen; Wiederherstellung des Systems [Instrumentelle Fertigkeiten, 6] *Sozialkompetenz* Kompetente Rollenerfüllung im Incident Response Team [Team-/Führungsfähigkeit, 6] *Selbstständigkeit* Verantwortliche Rollenerfüllung bei Incident Response *[Eigenständigkeit/Verantwortung,*				
4	**Inhalte:** **IT-Sicherheit:** ▪ IT-Sicherheitsmarkt in Deutschland ▪ Grundlegende Begriffe zur IT-Sicherheit ▪ Schutzziele in der IT-Sicherheit ▪ Schwachstellen, Bedrohungen und Angriffe ▪ Computer-Forensik, Digitale Forensik ▪ Sicherheitsvorfälle • Phasen eines Sicherheitsvorfalls • Weitere Modelle zur Beschreibung der Phasen eines Sicherheitsvorfalls • Klassifizierung von Sicherheitsvorfällen (Incident Taxonomie)				

	▪ Rechtliche Aspekte: IT-Sicherheitsgesetz & Netzwerkdurchsetzungsgesetz **Grundlagen von Hacking-Techniken:** ▪ Schadsoftware ▪ Bedrohungen im Rechnernetz ▪ Bedrohung aus dem Internet ▪ Social Engineering ▪ Erkennung & Analyse von Schadsoftware ▪ Maschinelles Lernen **IT-Verteidigungsmaßnahmen:** ▪ Prinzipien und grundlegende Mechanismen ▪ Sicherheitsmodelle ▪ IT-Grundschutz ▪ Sichere Software-Entwicklung und deren Betrieb ▪ Schutzmaßnahmen eines PC-Clients ▪ Nutzung einer Sandbox ▪ Konzepte zur Netzwerksicherheit **Penetration Testing:** ▪ Einordnung und Zielsetzung von Penetrationstests ▪ Rechtliche Gesichtspunkte ▪ Methodik von Penetrationstests ▪ Beispiele ▪ Vulnerability Assessment **Incident Response** ▪ Reaktion auf Sicherheitsvorfälle • Strukturelle Organisation • Incident Response Prozess (Vorbereitung, Erkennung, Analyse, Eindämmung / Beseitigung und Wiederherstellung, Nachbereitende Phase) • Checkliste für die Vorfallsbehandlung • Empfehlungen für die Vorfallsbehandlung ▪ Exemplarische Untersuchung eines Linuxsystems ▪ Analysewerkzeug **Fallbeispiele für Incident Response:** ▪ Vorfallsbehandlung eines Ransomware-Angriffs ▪ Vorfallsbehandlung eines SSH-Wurms ▪ Vorfallsbehandlung eines Keyloggers mit Datendiebstahl
	Empfohlene Literaturangaben: Literatur wird in der Lehrveranstaltung bekannt gegeben.
5	**Teilnahmevoraussetzungen:** Netzwerkgrundlagen, Kenntnisse in TCP/IP-Rechnernetzen
6	**Prüfungsformen***:* Klausur, Hausarbeit
7	***Voraussetzungen für die Vergabe von Kreditpunkten:***

	Das Einreichen der Übungsaufgaben ist obligatorisch für die Teilnahme an der Klausur. Um am Ende des Moduls ein Hochschulzertifikat mit ausgewiesenen ECTS-Leistungspunkten zu erlangen, müssen folgende Erfordernisse erbracht werden: - Einreichung aller Übungsaufgaben, mind. 50% erfolgreiche Bearbeitung der Aufgaben (Prüfungsleistung: Hausarbeit) - Prüfungsleistung Klausur bestanden Das Modul ist bestanden, wenn alle erforderlichen Leistungen erbracht wurden.
8	***Verwendbarkeit des Moduls:*** Master Digitale Forensik
9	***Modulverantwortliche(r):*** Prof. Dr. Martin Rieger
10	***Optionale Informationen:***

2.2.8 [Z-214] Forensische Analyse eines Windows-Systems für Sachverständige

Modul: [Z-214] Forensische Analyse eines Windows-Systems für Sachverständige						
Kennnummer Z-214	**Workload** 150 h	**Modulart**	**Studiensemester**		**Dauer** ca. 20 Wochen	**Häufigkeit**
1	**Lehrveranstaltung(en)**		**Sprache** deutsch	**Kontakt-zeit** 50 h	**Selbst-studium** 100 h	**Credits (ECTS)** 5
2	**Lehrform(en) / SWS:** Präsenzveranstaltung: Vorlesung, Übungen Onlineveranstaltung: Vorlesung, flexible Vertiefung wichtiger Themen, Lernen im Dialog, Übung					
3	**Lernergebnisse (learning outcomes), Kompetenzen:** *Kompetenz Wissen* Mögliche Nutzerspuren sind im Windows-System kennen [Wissen, 7]					
	Kompetenz Fertigkeiten Nutzerspuren im Windows-System bergen, analysieren und beurteilen *[Instrumentelle Fertigkeiten, 7]*					
	Sozialkompetenz Asservat im Team bearbeiten *[Team-/Führungsfähigkeit, 7]*					
	Selbstständigkeit Verantwortlicher Umgang mit Beweisen *[Eigenständigkeit/Verantwortung, 7]*					

4	**Inhalte*:***

- Das Windows-Rechnersystem
  Grundlegende Konzepte und Begriffe, Windows-„Bordwerkzeuge“ (Untersuchung von Prozessen und Threads, Leistungsüberwachung), System-Architektur (Gerätetreiber, Systemprozesse, Kernel, HAL), Sicherheitskomponenten und Rechtesystem, Reguläre Ausdrücke, Grundlagen der (Windows-) Netzwerktechnik, Ermitteln der Netzwerkeigenschaften des Rechners

- Spezifische Strukturen und Analysemethoden zu Windows-Systemen
  forensisch relevante Verzeichnisse und Dateien, Schattenkopien, Speicherabbilder gewinnen und auswerten, Protokolldateien gewinnen, Windows-Zugriffsrechte analysieren und verändern, Schlüsselwortsuche, Filecarving, Schlupfspeicher extrahieren, indizieren von Metadaten, Forensische Arbeitsweise im Windows-System

- Forensische Erkenntnisse aus der Registry
  Aufbau, SIDs, SAMs, GUID
  Forensisch relevante Registry-Einträge,
  Werkzeuge zur Registry-Analyse
  Antiforensische Maßnahmen

- Logfile-Analyse
  NTFS-Journal-Protokollierung, Struktur der Logging-Einträge, Auswertung, Windows-Event-Log, Anwendungs- und Dienstprotokolle, Security-Log, Setup-Log, Überwachungsrichtlinien
  Antiforensische Maßnahmen

- Forensische Untersuchung von Internetdiensten
  Peer-to-Peer-Aktivitäten aufdecken, Skype-Accounts untersuchen
  Client-Datenbanksystem, SQLite-Anwendungs-Artefakte auswerten (Skype, Firefox, Chrome), Microsoft-Anwendungs-Artefakte auswerten (Internet Explorer, Edge, Outlook)

- Forensische Analyse von Arbeitsspeicher und Windows-Live-Artefakten
  Flüchtige Informationen ermitteln, Systemzeit auslesen, eingeloggte Benutzer, offene Dateien, Netzwerkverbindungen, Prozessinformationen, Zwischenablage, Dienste/Treiber-Informationen, Erstellung eines Arbeitsspeicherabbilds,
  Arbeitsspeicheranalyse mit dem Volatility
  Artefakt-analyse

- Forensische Fallbeispiele
  - Analyse eines Rechners mit Malware-Befall:
    Indicators of Compromise, Schrittweise Analyse mit automatischen und händischen Methoden, Bereinigung des Rechnersystems

  - Der Fall Evil Knievel:

	Forensische Untersuchung auf Illegalen Handel, Auswertung vieler Windowsspezifischer Artefakte, Umgehen mit antiforensischen Maßnahmen o Der Fall Eve und Mallet: Forensische Untersuchung auf Rechnermanipulationen, Zeitstempelfälschungen; Löschen von Eventlogs; Löschen von Browserverläufen
	Empfohlene Literaturangaben: Literatur wird in der Lehrveranstaltung bekannt gegeben.
5	**Teilnahmevoraussetzungen:** Netzwerkgrundlagen, Kenntnisse in TCP/IP-Rechnernetzen
6	**Prüfungsformen***:* Klausur, Hausarbeit
7	***Voraussetzungen für die Vergabe von Kreditpunkten:*** Das Einreichen der Übungsaufgaben ist obligatorisch für die Teilnahme an der Klausur. Um am Ende des Moduls ein Hochschulzertifikat mit ausgewiesenen ECTS-Leistungspunkten zu erlangen, müssen folgende Erfordernisse erbracht werden: - Einreichung aller Übungsaufgaben, mind. 50% erfolgreiche Bearbeitung der Aufgaben (Prüfungsleistung: Hausarbeit) - Prüfungsleistung Klausur bestanden Das Modul ist bestanden, wenn alle erforderlichen Leistungen erbracht wurden.
8	***Verwendbarkeit des Moduls:*** Digitale Forensik
9	***Modulverantwortliche(r):*** Prof. Dr. Martin Rieger
10	***Optionale Informationen:*** Das Modul ist für Ermittler der Polizeibehörden ideal geeignet.

2.2.9 [Z-215] Forensische Analyse eines Unix-Systems für Sachverständige

<table>
<tr><td colspan="7">Modul: [Z-215] Forensische Analyse eines Unix-Systems für Sachverständige</td></tr>
<tr><td colspan="2">Kennnummer
Z-215</td><td>Workload
150 h</td><td>Modulart</td><td colspan="2">Studiensemester</td><td>Häufigkeit</td></tr>
<tr><td colspan="5"></td><td>Dauer
ca. 20 Wochen</td><td></td></tr>
<tr><td>1</td><td colspan="3">Lehrveranstaltung(en)</td><td>Sprache
deutsch</td><td>Kontakt-zeit
50 h</td><td>Selbst-studium
100 h</td><td>Credits (ECTS)
5</td></tr>
<tr><td>2</td><td colspan="7">Lehrform(en) / SWS:
Präsenzveranstaltung:
Vorlesung, Übungen
Onlineveranstaltung:
Vorlesung, flexible Vertiefung wichtiger Themen, Lernen im Dialog, Übung</td></tr>
<tr><td>3</td><td colspan="7">Lernergebnisse (learning outcomes), Kompetenzen:

Kompetenz Wissen
Mögliche Nutzerspuren sind im Unix-System kennen [Wissen, 7]
Kompetenz Fertigkeiten
Nutzerspuren im Unix-System bergen, analysieren und beurteilen [Instrumentelle Fertigkeiten, 7]
Sozialkompetenz
Asservat im Team bearbeiten [Team-/Führungsfähigkeit, 7]
Selbstständigkeit
Verantwortlicher Umgang mit forensischen Beweisen [Eigenständigkeit/Verantwortung, 7]</td></tr>
<tr><td>4</td><td colspan="7">Empfohlene Literaturangaben:
• Herold, H; Lurz, B; Wohlrab, J.: Grundlagen der Informatik. München; Boston [u.a.]: Pearson Studium.
• Tanenbaum, A. S. (2006): Computerarchitektur : Strukturen - Konzepte – Grundlagen. München; [Boston {u.a.]: Pearson Studium
• Brian Ward: How Linux Works: What Every Superuser Should Know No Starch Press; Auflage: 2 (11. November 2014)
• Michael Kofler: Linux: Das umfassende Handbuch. Rheinwerk Computing; Auflage: 14 (30. November 2015)
• Nemeth, Evi, Snyder, Garth, Hein, Trent R., Whaley, Ben: UNIX and Linux System Administration Handbook Prentice Hall 4th Edition (2011)
• Dr. Philip Polstra: Linux Forensics CreateSpace Independent Publishing Platform; Auflage: 1 (13. Juli 2015).</td></tr>
</table>

5	**Teilnahmevoraussetzungen:**
6	**Prüfungsformen***:* Hausarbeit
7	***Voraussetzungen für die Vergabe von Kreditpunkten:*** Bestehen der Hausarbeit
8	***Verwendbarkeit des Moduls:***
9	***Modulverantwortliche(r):*** Prof. Dr. Martin Rieger
10	***Optionale Informationen:*** Nur für Sachverständige

2.2.10 [Z-216] Forensische Analyse eines MacOS-Systems für Sachverständige

Modul: [Z-216] Forensische Analyse eines MacOS-Systems für Sachverständige					
Kennnummer Z-216	**Workload** 150 h	**Modulart**	**Studiensemester**	**Dauer** ca. 20 Wochen	**Häufigkeit**

1	**Lehrveranstaltung(en)**	**Sprache** deutsch	**Kontakt-zeit** 50 h	**Selbst-studium** 100 h	**Credits (ECTS)** 5
2	**Lehrform(en) / SWS:**				
3	**Lernergebnisse (learning outcomes), Kompetenzen:** *Kompetenz Wissen* Mögliche Nutzerspuren sind im MacOS-System kennen [Wissen, 7]				
	Kompetenz Fertigkeiten Nutzerspuren im MacOS-System bergen, analysieren und beurteilen [Instrumentelle Fertigkeiten, 7]				
	Sozialkompetenz Asservat im Team bearbeiten [Team-/Führungsfähigkeit, 7]				
	Selbstständigkeit Verantwortlicher Umgang mit forensischen Beweisen [Eigenständigkeit/Verantwortung, 7]				
	Inhalte*:*				

In diesem Modul werden Ihnen verschiedene Aspekte des Betriebssystems MacOS vermittelt, die es Ihnen ermöglichen Untersuchungen forensischer Art oder zur IT-Sicherheit an den genannten Betriebssystemen durchzuführen. Eine Grundlage hierfür ist das Verständnis über wichtige Konzepte und Eigenschaften von Linux, FreeBSD und vor allem den MacOS-spezifischen Komponenten. In den einzelnen Studienbriefen werden verschiedene Bereiche des MacOS-Betriebssystems betrachtet und analysiert.

- Apple-Hardware
  Apple Inc. Firmengeschichte, Desktop und mobile Computer, Set Top Boxen, Server, tragbare Geräte, Software, Hardwarearchitekturen (Litte/Big Endian), Datenübertragungsschnittstellen

- MacOS-Betriebssystem
  Klassisches Mac OS/System 7, OPENSTEP, macOS Versionen (Client), macOS Versionen (Server), MacOSX,
  Betriebssystemarchitektur (Kernel, Userland),
  Property Lists, AppleScript,
  Sicherheitsfunktionen (Sandboxing, XPC, Gatekeeper, Xprotect, Fileattributes)
  Dateisysteme HFS+, APFS
  Forensisch bedeutsame Artefakte, die durch MacOS und das Dateissystem erzeugt werden.

- Persistente Spuren:
  Die MacOS Verzeichnisstruktur (Benutzer, Library, Spotlight, Netzwerkkonfiguration, Drucker, Repositoryverwaltungen),
  Nutzerdomäne (Benutzerverwaltung, Applikationsstruktur, Zeitstempel, Apple Applikationen, Papierkorb, Backups, Virtualisierung, Zuletzt verwendete Objekte)
  Spezifische Formate und deren Auswertung
  Forensische Analyse der persistenten Spuren an Beispielen

- Netzwerkbasierte Dienste:
  iCloud Services, Mobile Device Management, Synchronisation iCloud und lokale Verzeichnisse
  Netzwerkdienste in der lokalen Domäne (AFP, SMB, VNC, FTP, SSH, ARD, Webserver)

- Methoden digitaler Forensik im MacOS-Umfeld:
  Investigativer Prozess nach Casey
  MacOS Sicherungsvorgehen,
  Liveanalyse,
  Post Mortem Analyse (Disk Arbitration, Verschlüsselung, Livesystem)

- Nichtpersistente Spuren und Forensische Analyse von Arbeitsspeichern:
  Einführung in die RAM-Analyse, Struktur des Arbeitsspeichers,
  Erstellen eines Arbeitsspeicherabbilds,
  Möglichkeiten der Erfassung, Erstellen von Profilen für das Volatility Framework,

- Fallbeispiele:

	Analyse eines MacOS-Livesystems mittels eigener Skripte Forensische Arbeitsspeicheranalyse eines kompromittierten Clients Übung Fallbeispiel Datenträger: Forensische Untersuchung auf Illegalen Handel, Auswertung vieler MacOS-spezifischer Artefakte, Umgehen mit antiforensischen Maßnahmen Übung Fallbeispiel Antiforensische Maßnahmen Gelöschte Logfiles, gelöschte Konten, gefälschte Zeitstempel
	Empfohlene Literaturangaben: • Jonathan Levin. Mac OS X and IOS Internals: To the Apple's Core. John Wiley & Sons • Topher Kessler. EFI firmware protection locks down newer Macs. Website,. https://www.cnet. com/news/efi- firmware- protection- locks- down- newer- macs/. • Maximillian Dornseif. Vorlesung Computerforensik. Friedrich-Alexander Universität Erlangen-Nürnberg, • Brian Ward: How Linux Works: What Every Superuser Should Know No Starch Press; Auflage: 2 (11. November 2014) • Marc Brandt. Forensische Analyse von Mac OS X. Hochschule Albstadt-Sigmaringen, 2016. • Brian Carrier. File system forensic analysis. Addison-Wesley Pro-fessional, 2005. • Eoghan Casey. Handbook of digital forensics and investigation. Academic Press, 2009. • A. Singh. Mac OS X Internals: A Systems Approach. Pearson Edu-cation, 2006. ISBN 9780132702263. URL https://books.google.co.il/books?id=K8vUkpOXhN4C. • Jesse Varsalone. Mac OS X, iPod, and iPhone forensic analysis DVD toolkit. Syngress, 2008. • Herold, H; Lurz, B; Wohlrab, J.: Grundlagen der Informatik. München; Boston [u.a.]: Pearson Studium. • Tanenbaum, A. S. (2006): Computerarchitektur : Strukturen - Konzepte – Grundlagen. München; [Boston {u.a.]: Pearson Studium • Brian Ward: How Linux Works: What Every Superuser Should Know No Starch Press; Auflage: 2 (11. November 2014) • Michael Kofler: Linux: Das umfassende Handbuch. Rheinwerk Computing; Auflage: 14 (30. November 2015) • Nemeth, Evi, Snyder, Garth, Hein, Trent R., Whaley, Ben: UNIX and Linux System Administration Handbook Prentice Hall 4th Edition (2011) • Dr. Philip Polstra: Linux Forensics CreateSpace Independent Publishing Platform; Auflage: 1 (13. Juli 2015). Weitere Literatur wird in der Lehrveranstaltung bekannt gegeben.
5	**Teilnahmevoraussetzungen:**
6	**Prüfungsformen***:* Klausur, Hausarbeit

7	***Voraussetzungen für die Vergabe von Kreditpunkten:*** Das Einreichen der Übungsaufgaben ist obligatorisch für die Teilnahme an der Klausur. Um am Ende des Moduls ein Hochschulzertifikat mit ausgewiesenen ECTS-Leistungspunkten zu erlangen, müssen folgende Erfordernisse erbracht werden: - Einreichung aller Übungsaufgaben, mind. 50% erfolgreiche Bearbeitung der Aufgaben (Prüfungsleistung: Hausarbeit) - Prüfungsleistung Klausur bestanden Das Modul ist bestanden, wenn alle erforderlichen Leistungen erbracht wurden.
8	***Verwendbarkeit des Moduls:*** Digitale Forensik
9	***Modulverantwortliche(r):*** Prof. Dr. Martin Rieger
10	***Optionale Informationen:*** Nur für Sachverständige

2.2.11 [Z-217] Forensische Analyse von Netzwerken für Sachverständige

Modul: [Z-217] Forensische Analyse von Netzwerken für Sachverständige					
Kennnummer Z-217	**Workload** 150 h	**Modulart**	**Studiensemester**	**Dauer** ca. 20 Wochen	**Häufigkeit**

1	**Lehrveranstaltung(en)**	**Sprache** deutsch	**Kontakt-zeit** 50 h	**Selbst-studium** 100 h	**Credits (ECTS)** 5
2	**Lehrform(en) / SWS:**				
3	**Lernergebnisse (learning outcomes), Kompetenzen:** *Kompetenz Wissen* Mögliche Nutzerspuren sind im Netzwerk kennen [Wissen, 7] *Kompetenz Fertigkeiten* Nutzerspuren im Netzwerk bergen, analysieren und beurteilen [Instrumentelle Fertigkeiten, 7] *Sozialkompetenz* Asservat im Team bearbeiten [Team-/Führungsfähigkeit, 7] *Selbstständigkeit* Verantwortlicher Umgang mit forensischen Beweisen [Eigenständigkeit/Verantwortung, 7]				
4	**Inhalte*:*** Grundlagen der Netzwerkforensik				

- Forensische Untersuchungen an Rechnern in Netzwerken
- Datengewinnung aus aktiven Netzkomponenten
- Datengewinnung aus dem Netzwerkdatenstrom mittels Netzwerk-Sniffer
- IT-Strukturen
- Network Security Monitoring (NSM) Vorgehensmodell

Post Mortem-Analyse von Server-Diensten und -Komponenten

- Analyse von Logdateien
- Sicherung und Analyse von Serverdiensten
- Analyse Microsoft Serverdiensten
- Forensische Auswertung von Routern und anderen Netzwerkkomponenten

Live-Analyse von Server-Diensten und -Komponenten

- Aufbereitung von Server-Sicherungen zur Virtualisierung
- Live-Analyse laufender Systeme am Beispiel
- Erstellung und Analyse eines Arbeitsspeicherabbildes
- Analyse von IoT-Systemen
- Analyse von Schadsoftware

Internet- und Cloudforensik

- Internet- und Cloudspuren im Client
- Geräte im Netzwerk erkennen
- Sichern von Clouddaten am Beispiel Facebook
- Kryptowährungen

Forensische Fallbeispiele

- DoS-Angriff auf virtuelle Netzwerkumgebung:

Auswertung der Spuren ergibt das Schadensbild Spuren zum Tathergang und zu den Tätern

- ARP-Spoofing- Angriff in virtueller Netzwerkumgebung analysieren:

Auswertung der Spuren ergibt das Schadensbild Spuren zum Tathergang und zu den Tätern

- Browser-Analyse und Facebook-Analyse:

Nachweis von illegalem Handel einer Bande

- Die Milka-Bande:

  Nachweis von illegalem Handel einer Bande in einer ausgedehnten virtuellen Netzwerkumgebung; als Methoden werden u.a. Netzwerkmitschnitte, Analyse der mitschnitte, sowie Eindringen und Zugriff auf Server angewendet.

Empfohlene Literaturangaben:

- *Andrew Tanenbaum (2020): Computernetzwerke. Verlag Pearson Studium; 5. Auflage:*
- *Claudia Eckert (2016): IT-Sicherheit Strukturen - Konzepte – Grundlagen. Verlag De Gruyter Oldenbourg, 9. Auflage.)*
- *Jörg Schwenk (2014): Sicherheit und Kryptographie im Internet. Verlag: Springer Vieweg, 4.Auflage:*
- *S. Davidoff, J. Ham (2012): Network Forensics. Verlag Prentice Hall International.*
- *Chris Sanders, Jason Smith (2013): Applied Network Security Monitoring: Collection, Detection, and Analysis. Verlag: Syngress.*

5	**Teilnahmevoraussetzungen:**
6	**Prüfungsformen***:* Klausur, Hausarbeit
7	***Voraussetzungen für die Vergabe von Kreditpunkten:*** Das Einreichen der Übungsaufgaben ist obligatorisch für die Teilnahme an der Klausur. Um am Ende des Moduls ein Hochschulzertifikat mit ausgewiesenen ECTS-Leistungspunkten zu erlangen, müssen folgende Erfordernisse erbracht werden: - Einreichung aller Übungsaufgaben, mind. 50% erfolgreiche Bearbeitung der Aufgaben (Prüfungsleistung: Hausarbeit) - Prüfungsleistung Klausur bestanden Das Modul ist bestanden, wenn alle erforderlichen Leistungen erbracht wurden.
8	***Verwendbarkeit des Moduls:*** Digitale Forensik
9	***Modulverantwortliche(r):*** Prof. Dr. Martin Rieger
10	***Optionale Informationen:*** Nur für Sachverständige

2.2.12 [Z-220] Netzsicherheit I: IT-Sicherheit von Netzwerken

Modul: [Z-220] Netzsicherheit I: IT-Sicherheit von Netzwerken						
Kennnummer Z-220	**Workload** 150 h	**Modulart**	**Studiensemester**	**Dauer** ca. 12 Wochen	**Häufigkeit**	
1	**Lehrveranstaltung(en)**	**Sprache** deutsch	**Kontakt-zeit** 15 h	**Selbst-studium** 135 h	**Credits (ECTS)** 5	
2	**Lehrform(en) / SWS:** E-Learning Lernplattform: Aufzeichnungen, Interaktive Kontrollaufgaben, Hausaufgaben/Projektaufgaben Online-Vorlesungen: Vorlesung, Fragen und Antworten, Übungen, flexible Vertiefung wichtiger Themen Präsenz-/Remoteveranstaltung: Vorlesung, Übungen mit theoretischem und praktischem Schwerpunkt					
3	**Lernergebnisse (learning outcomes), Kompetenzen:** *Kompetenz Wissen* In der Lehrveranstaltung „Netzwerksicherheit I: IT-Sicherheit in Netzwerken“ erhalten Sie einen Überblick über die Bedrohungen und Angriffe gegen und über Netzwerke. Darüber hinaus lernen Sie die in Rechnernetzen eingesetzten Technologien und die wichtigsten Merkmale und Eigenschaften von Datennetzen kennen. Die wichtigsten Sicherheitsprotokolle, die häufigsten Angriffe auf Netzwerke und die entsprechenden Abwehrmaßnahmen werden erläutert. In Übungen im virtuellen Labor führen Sie selbst Angriffe durch, um anschließend Bedrohungsszenarien nachvollziehen und ein-ordnen zu können. Nach erfolgreichem Abschluss des Moduls kennen Sie die wichtigsten Merkmale und Eigenschaften und können die verwendeten Sicherheitskonzepte einordnen. Darüber hinaus haben Sie Kenntnisse über Bedrohungen und den Einsatz von Werkzeugen er-worben, um die Möglichkeiten und Grenzen selbst einschätzen zu können. *Kompetenz Fertigkeiten* *Sozialkompetenz* *Selbstständigkeit*					
4	**Inhalte:** **Teil I: Sniffing & Scanning** Protokolle Media Access Control (MAC), Address Resolution Protocol (ARP), Internet Protocol (IP) Cyber Security Bedrohungen, Angreifertypen, Angriffsarten, Schutzziele Angriffsvektoren Eavesdropping, Hacking-Hardware, Arten von Scans Gegenmaßnahmen Physischer Schutz, Honeypots und Honeynets, Seperation von Netzen **Teil II: Denial-of-Service**					

	Protokolle Transmission Control Protocol (TCP), User Datagram Protocol (UDP), Transport Layer Security (TLS) Cyber Security Hashfunktionen, Verschlüsselungsmethoden Angriffsvektoren ARP-Cache-Flooding, IP-Fragment-Flag, ICMP Flood, TCP-SYN-Flooding, DNS Amplification Gegenmaßnahmen Firewalls, IDS & IPS, TCP-SYN-Cookie, Timings **Teil III: Spoofing & Man-in-the-Middle** Protokolle Domain Name System (DNS), Dynamic Host Configuration Protocol (DHCP) Cyber Security Informationssicherheit vs. Datenschutz Angriffsvektoren ARP-Spoofing, IP-Spoofing, DNS-Spoofing, Rogue-DHCP-Server, Hacking-Hardwar Gegenmaßnahmen MAC-Filter, ARP-Proxy, DNSSEC, Dot, DoH, DHCP-Snooping, TLS **Teil IV: Virtuelle Netze & Verschleierung** Protokolle Virtual Private Network (VPN), Peer-to-Peer (P2P), Tor-Protokoll, Routing Cyber Security Incident Response und IT-Forensik Angriffsvektoren Verschleierung Gegenmaßnahmen Virtual Local Area Network (VLAN)
	Empfohlene Literaturangaben: • IT-Sicherheit für TCP/IP- und IoT-Netzwerke: Grundlagen, Konzepte, Protokolle, Härtung | Steffen Wendzel | Springer Vieweg, Wiesbaden | ISBN 9783864914898 • Netzsicherheit: Grundlagen & Protokolle; mobile & drahtlose Kommunikation; Schutz von Kommunikationsinfrastrukturen | Günter Schäfer; Michael Roßberg | dpunkt.verlag, Heidelberg | ISBN 9783864901157 • IT-Sicherheit: Konzepte - Verfahren - Protokolle | Claudia Eckert | • De Gruyter, Oldenbourg | ISBN 9783110551587 • Kryptographie und IT-Sicherheit | Stephan Spitz; Michael Pramateftakis; Joachim Swoboda | Springer Vieweg, Wiesbaden | ISBN 9783834881205 • Computernetzwerke | Andrew S. Tanenbaum; David J. Wetherall | Pearson, München | ISBN 9783868941371 Weitere Literatur wird in der Lehrveranstaltung bekannt gegeben.
5	**Teilnahmevoraussetzungen:** • Grundlegendes Verständnis von Betriebssystemen (Prozesse, Speicher, Geräte, ...) • Grundlegende Kenntnisse der Netzwerktechnik (Techniken, Protokolle, …) • Umgang mit der Linux Shell (Bash und Zsh) • gutes analytisches Denken und methodisches Vorgehen • Englischkenntnisse auf B1 Niveau • intrinsische Motivation für ein berufsbegleitendes Zertifikatsmodul in Fernlehre Sollten diese Voraussetzungen partiell nicht erfüllt werden, sollte dies im Einzelfall geklärt werden.
6	**Prüfungsformen**: Hausarbeit + Referat (15 + 5 Min.) mit obligatorischer Leistungserbringung im Lernmanagementsystem (Lernsequenzen + Übungsaufgaben)
7	***Voraussetzungen für die Vergabe von Kreditpunkten:*** Das Einreichen der Übungsaufgaben ist obligatorisch. Für den Erhalt der ausgewiesenen ECTS-Leistungspunkten müssen folgende Erfordernisse erbracht werden: - Einreichung aller Übungsaufgaben, mind. 50% erfolgreiche Bearbeitung der Aufgaben - Prüfungsleistung Klausur bestanden

	Das Modul ist bestanden, wenn alle erforderlichen Leistungen erbracht wurden.
8	***Verwendbarkeit des Moduls:***
9	***Modulverantwortliche(r):***
10	***Optionale Informationen:***

2.3 Goethe-Universität Frankfurt am Main / Universität des Saarlandes

2.3.1 [Z-401] Computerstrafrecht

Modul: [Z-401] Computerstrafrecht						
Kennnummer Z-401	**Workload** 150 h	**Modulart**	**Studiensemester**	**Dauer** ca. 10 Wochen	**Häufigkeit**	
1	**Lehrveranstaltung(en)**		**Sprache** deutsch	**Kontakt-zeit** 25 h	**Selbst-studium** 125 h	**Credits (ECTS)** 5
2	**Lehrform(en) / SWS:** Präsenzveranstaltung: Vorlesung Onlineveranstaltung: Vorlesung, gegebenenfalls flexible Vertiefung wichtiger Themen					
3	**Lernergebnisse (learning outcomes), Kompetenzen:** *Kompetenz Wissen* Nach erfolgreichem Abschluss des Moduls haben die Teilnehmer Kenntnisse über die Grundzüge des Computerstrafrechts und die verschiedenen Facetten der Computer- und Internetkriminalität. Sie sind in der Lage, grundsätzliche Aussagen über das Phänomen Computerkriminalität zu treffen und Einschätzungen hinsichtlich der Strafbarkeit einzelner, damit verbundener Verhaltensweisen abzugeben. Dabei erwerben Sie sowohl Fach- als auch eine grundlegende Methodenkompentenz. *Kompetenz Fertigkeiten* *Sozialkompetenz* *Selbstständigkeit*						

4	**Inhalte*:*** Das Modul befasst sich in mehreren Studienbriefen mit dem Phänomen der Computerkriminalität. Um die damit auftretenden Probleme richtig einordnen zu können, wird in Studienbrief 1 zunächst ein Mindestmaß an Grundwissen vermittelt. Diese Einführung in das materielle Strafrecht stellt die Basis für die in den weiteren Studienbriefen vertiefte Auseinandersetzung mit den Tatbeständen dar, die üblicherweise unter den Begriff der Computer- und Internetkriminalität subsumiert werden. Die Studienbriefe fassen die damit zusammenhängenden und dahinterstehenden rechtlichen Probleme in Themenkomplexen zusammen. Beispielfälle und Bezugnahmen auf einschlägige Rechtsprechung sollen helfen, die oft abstrakte Materie greifbar und nachvollziehbar zu machen. Die Darstellung erfolgt dabei anhand der einschlägigen Delikte des Strafgesetzbuches sowie einzelner Tatbestände des Nebenstrafrechts, die im Einzelnen näher erklärt und dargestellt werden. Darüber hinaus werden aber auch Grundzüge der mit dem Medium Internet verbundenen verfassungsrechtlichen Fragen sowie rechtliche Rahmenbedingungen für die Anbieter von Inhalten behandelt. **Praktische Übung:** Übungsfälle am Ende der Studienbriefe, Kontrollaufgaben
	Empfohlene Literaturangaben: Literatur wird in der Lehrveranstaltung bekannt gegeben.
5	**Teilnahmevoraussetzungen:** keine
6	**Prüfungsformen***:* Klausur, Seminararbeit, Präsentation
7	***Voraussetzungen für die Vergabe von Kreditpunkten:*** Bestehen der Prüfungsleistung (Kurzhausarbeit)
8	***Verwendbarkeit des Moduls:*** Master Digitale Forensik
9	***Modulverantwortliche(r):*** Prof. Dr. Christoph Burchard / Prof. Dr. Dominik Brodowski
10	***Optionale Informationen:***

2.3.2 [Z-402] Computerstrafprozessrecht

<table>
<tr><td colspan="8">Modul: [Z-402] Computerstrafprozessrecht</td></tr>
<tr><td colspan="2">Kennnummer
Z-402</td><td>Workload
150 h</td><td>Modulart</td><td colspan="2">Studiensemester</td><td>Dauer
ca. 8 Wochen</td><td>Häufigkeit</td></tr>
<tr><td>1</td><td colspan="3">Lehrveranstaltung(en)</td><td>Sprache
deutsch</td><td>Kontakt-zeit
25 h</td><td>Selbst-studium
125 h</td><td>Credits (ECTS)
5</td></tr>
<tr><td>2</td><td colspan="7">Lehrform(en) / SWS:
Präsenzveranstaltung:
Vorlesung
Onlineveranstaltung:
Vorlesung, gegebenenfalls flexible Vertiefung wichtiger Themen</td></tr>
<tr><td>3</td><td colspan="7">Lernergebnisse (learning outcomes), Kompetenzen:

Kompetenz Wissen
Die Teilnehmer erwerben Grundkenntnisse des Strafprozessrechts. Sie können die Grundzüge des Computerstrafprozessrechts in Bezug zur Informationstechnologie und zum Verfassungsrecht setzen. Nach erfolgreichem Abschluss des Moduls sind die Studierenden in der Lage, verfahrensrechtliche Maßnahmen auf ihre Zulässigkeit zu überprüfen und hierzu kritisch Stellung zu nehmen. Dabei erwerben Sie sowohl Fach- als auch eine grundlegende Methodenkompentenz.
Kompetenz Fertigkeiten
Sozialkompetenz
Selbstständigkeit</td></tr>
<tr><td>4</td><td colspan="7">Inhalte:

Das Modul befasst sich in mehreren Studienbriefen mit den Auswirkungen der Informationstechnologie auf das Strafprozessrecht. Unter Bezugnahme auf die im Modul Computerstrafrecht erworbenen materiellrechtlichen Grundkenntnisse werden im Modul grundlegende Kenntnisse im Bereich des Verfahrensrechts und des formellen Strafrechts vermittelt.
Auch in diesem Modul wird regelmäßig Bezug auf einschlägige Rechtsprechung genommen und Wert auf eine fallbezogene Wissensvermittlung gelegt. Angesichts der besonderen Bedeutung des Strafverfahrensrechts werden aber auch Grundzüge verfassungsrechtlicher Fragestellungen behandelt.

Praktische Übungen:
Übungsfälle am Ende der Studienbriefe, Kontrollaufgaben

Empfohlene Literaturangaben:
Literatur wird in der Lehrveranstaltung bekannt gegeben.</td></tr>
<tr><td>5</td><td colspan="7">Teilnahmevoraussetzungen:
abgeschlossenes Modul Computerstrafrecht</td></tr>
<tr><td>6</td><td colspan="7">Prüfungsformen:</td></tr>
</table>

	Klausur, Seminararbeit, Präsentation
7	***Voraussetzungen für die Vergabe von Kreditpunkten:*** Bestehen der Klausur, Seminararbeit, Präsentation
8	***Verwendbarkeit des Moduls:***
9	***Modulverantwortliche(r):*** Prof. Dr. Christoph Burchard / Prof. Dr. Dominik Brodowski
10	***Optionale Informationen:***

2.4 Universität Passau

2.4.1 [Z-801] Cloud-Sicherheit und Cloud-Forensik - Angriffsanalyse

Modul: [Z-801] Cloud-Sicherheit und Cloud-Forensik mit Schwerpunkt Angriffsanalyse						
Kennnummer Z-801	**Workload** 150 h	**Modulart**	**Studiensemester**		**Dauer** ca. 10 Wochen	**Häufigkeit**
1	**Lehrveranstaltung(en)**		**Sprache** deutsch	**Kontakt-zeit** 15 h	**Selbst-studium** 135 h	**Credits (ECTS)** 5
2	**Lehrform(en) / SWS:** Präsenzveranstaltung: Vorlesung, Übung Onlineveranstaltung: Vorlesung, flexible Vertiefung wichtiger Themen, Lernen im Dialog, Übung					
3	**Lernergebnisse (learning outcomes), Kompetenzen:** *Kompetenz Wissen* Nach Abschluss dieses Moduls verfügen Sie über fundierte Kenntnisse im Bereich von Cloud-Sicherheit und Cloud-Forensik. Neben den Konzepten und Architekturen von Virtualisierung umfassen diese Kenntnisse das Wissen über Sicherheitsherausforderungen und Bedrohungsmodellen in Cloud-Infrastrukturen sowie einen Überblick über aktuelle Forensik Methoden und entsprechende Werkzeuge. Darüber hinaus haben Sie weiterführende Kompetenzen in der Verwendung von Virtual Machine Introspection, Honeypots und Einbruchserkennungssystemen als Werkzeuge zur Angriffsanalyse erworben.					
	Kompetenz Fertigkeiten					
	Sozialkompetenz					
	Selbstständigkeit					

4	**Inhalte:** • Virtualisierungstechnik und Cloud-Computing o Geschichtliche Hintergründe von Virtualisierungstechnik o Virtualisierungsarten o Details von Rechnervirtualisierung (Intel/ARM) o Service- und Verarbeitungsmodelle von Cloud Computing (NIST) • Cloud-Sicherheit und Bedrohungsmodelle o Bedrohungsmodellierung und Risikomanagement o Sicherheitsherausforderungen in der Cloud o Sichere Datenspeicherung und –verarbeitung in der Cloud o Koresidenz und Seitenkanalangriffe • Grundlagen von Cloud-Forensik o Historische Entwicklung von IT-Forensik o Aktuelle Modelle der IT-Forensik o Datenträger-Forensik in der Cloud o Live-Forensik in der Cloud o Forensik-Dienste und Forensik-Readiness-Modelle in der Cloud • Virtual Machine Introspection (VMI) o Grundlagen, Herausforderungen und Anwendungen von VMI o Funktionsweise der Analysebibliothek LibVMI o Untersuchung von Linux-Kernel-Datenstrukturen mit Volatility o Untersuchung aktiver virtueller Maschinen mit Volatility/LibVMI • Cloud-Einbruchserkennungssysteme und Honey Pots o Grundlagen von Einbruchserkennungssystemen o Grundlagen von Honeypots o Einbruchserkennungssysteme in der Cloud o Honeypots in der Cloud *(Voraussichtliche Ergänzungen: Service und Verarbeitungsmodell, Identidy and Access Management, weitere Cloud Deployment Modelle)* *Empfohlene Literaturangaben:* Literatur wird in der Lehrveranstaltung bekannt gegeben.
5	**Teilnahmevoraussetzungen:** Grundlegende Programmierkenntnisse in Python Empfohlen: Grundverständnis von Betriebssystemen, Linux-Kenntnisse
6	**Prüfungsformen***:* Klausur, Hausarbeit
7	***Voraussetzungen für die Vergabe von Kreditpunkten:*** Das Einreichen der Übungsaufgaben ist obligatorisch. Darüber hinaus geht eine erfolgreiche Teilnahme mit der Abgabe aller Übungsaufgaben sowie einer mindestens 50 % erforderlichen erfolgreichen Bearbeitung einher. Um am Ende des Moduls ein Hochschulzertifikat zu erlangen, müssen oben genannte Erfordernisse

	(Einreichung der Übungsaufgaben, mind. 50% erfolgreiche Bearbeitung der Aufgaben sowie alle Prüfungsleistungen) erbracht werden.
8	***Verwendbarkeit des Moduls:*** In welchen Studiengängen ist das Modul einsetzbar?.
9	***Modulverantwortliche(r):*** Prof. Dr. Hans P. Reiser
10	***Optionale Informationen:***

2.4.2 [Z-802] Cloud-Sicherheit und Cloud-Forensik - Zugriffskontrolle

Modul: [Z-802] Cloud-Sicherheit und Cloud-Forensik mit Schwerpunkt Zugriffskontrolle						
Kennnummer Z-802	**Workload** 150 h	**Modulart**	**Studiensemester**		**Dauer** ca. 8 Wochen	**Häufigkeit**
1	**Lehrveranstaltung(en)**		**Sprache** deutsch	**Kontakt-zeit** 15 h	**Selbst-studium** 135 h	**Credits (ECTS)** 5
2	**Lehrform(en) / SWS:** Präsenzveranstaltung: Vorlesung, Übung Onlineveranstaltung: Vorlesung, flexible Vertiefung wichtiger Themen, Lernen im Dialog, Übung					
3	**Lernergebnisse (learning outcomes), Kompetenzen:** *Kompetenz Wissen* Nach Abschluss dieses Moduls verfügen Sie über fundierte Kenntnisse im Bereich von Cloud-Sicherheit und Cloud-Forensik. Neben den Konzepten und Architekturen von Virtualisierung umfassen diese Kenntnisse das Wissen über Sicherheitsherausforderungen und Bedrohungsmodellen in Cloud-Infrastrukturen sowie einen Überblick über aktuelle Forensik Methoden und entsprechende Werkzeuge. Dar-über hinaus haben Sie weiterführende Kompetenzen in der Einrichtung von Identitätsmanagementsystemen erworben, um Zugriffe zu beschränken, und in der Verwendung von Einbruchserkennungssystemen, um unerlaubte Zugriffe nachzuvollziehen.					
	Kompetenz Fertigkeiten					
	Sozialkompetenz					
	Selbstständigkeit					
4	**Inhalte:** • Virtualisierungstechnik und Cloud-Computing					

	○ Geschichtliche Hintergründe von Virtualisierungstechnik ○ Virtualisierungsarten ○ Details von Rechnervirtualisierung (Intel/ARM) ○ Service- und Verarbeitungsmodelle von Cloud Computing (NIST) • Cloud-Sicherheit und Bedrohungsmodelle ○ Bedrohungsmodellierung und Risikomanagement ○ Sicherheitsherausforderungen in der Cloud ○ Sichere Datenspeicherung und -verarbeitung in der Cloud ○ Koresidenz und Seitenkanalangriffe • Grundlagen von Cloud-Forensik ○ Historische Entwicklung von IT-Forensik ○ Aktuelle Modelle der IT-Forensik ○ Datenträger-Forensik in der Cloud ○ Live-Forensik in der Cloud ○ Forensik-Dienste und Forensik-Readiness-Modelle in der Cloud • Cloud-Einbruchserkennungssysteme und Honey Pots ○ Grundlagen von Einbruchserkennungssystemen ○ Grundlagen von Honeypots ○ Einbruchserkennungssysteme in der Cloud ○ Honeypots in der Cloud • Identitätsmanagement und Single Sign-on ○ Grundlagen zu Authentisierung und Autorisierung ○ Identitätsmanagement und Single-Sign-On-Systeme ○ Einrichtung und Verwendung von OAuth/OpenID Connect ○ Aktuelle Herausforderungen in Authentisierung und Autorisierung
	Empfohlene Literaturangaben: Literatur wird in der Lehrveranstaltung bekannt gegeben.
5	**Teilnahmevoraussetzungen:** Grundlegende Programmierkenntnisse in Python Empfohlen: Grundverständnis von Betriebssystemen, Linux-Kenntnisse
6	**Prüfungsformen***:* Klausur, Hausarbeit
7	***Voraussetzungen für die Vergabe von Kreditpunkten:*** Bestehen der Klausur, Hausarbeit
8	***Verwendbarkeit des Moduls:***
9	***Modulverantwortliche(r):*** Prof. Dr. Hans P. Reiser
10	***Optionale Informationen:***