Certificate of Advanced Studies

Security Incident Analysis and Reaction

Angriffe aus dem Internet können trotz Gegenmassnahmen den Betrieb von IT-Systemen erheblich beeinträchtigen. Gegen solche Vorfälle muss schnell, zielgerichtet und präzise vorgegangen werden. Im CAS «Security Incident Analysis and Reaction» lernen Sie, die entdeckten Angriffe methodisch, professionell und effektiv zu analysieren und alle notwendigen Massnahmen einzuleiten.

Berner
Fachhochschule

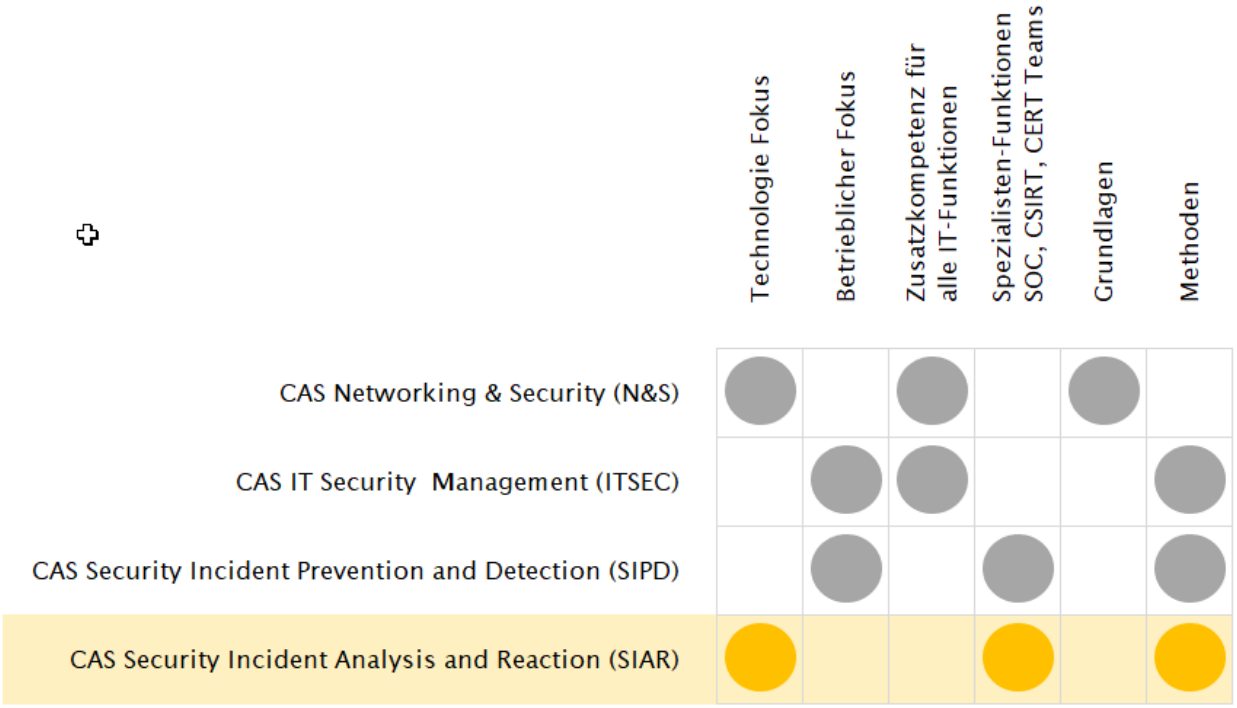
Inhaltsverzeichnis

Stand: 11.03.2024

1 Umfeld

In der heutigen IT-Landschaft lassen sich Sicherheitszwischenfälle nicht vollständig vermeiden. Je früher ein Vorfall erkannt und je schneller er behoben wird, desto geringer ist der entstandene Schaden. Das CAS «Security Incident Analysis and Reaction» (SIAR) setzt den Fokus auf die Analyse von sicherheitsrelevanten Ereignissen sowie eine rasche und zielgerichtete Reaktion.

Das CAS SIAR ergänzt das CAS «Security Incident Prevention and Detection» zu einem ganzheitlichen Security Incident Management. Zusammen mit den beiden CAS «Networking & Security» und «IT Security Management» ergeben alle vier CAS eine hervorragende Ausgangslage für den erfolgreichen Abschluss des MAS Cyber Security.



2 Zielpublikum

Das CAS SIAR richtet sich an IT-Security-Fachkräfte, die in einem spezialisierten Team (SOC, CSIRT, CERT) eine Security-Tätigkeit wahrnehmen und im Rahmen ihrer Arbeit Sicherheitsvorfälle rasch detektieren und effizient behandeln müssen.

3 Ausbildungsziele

- Sie können Angriffe auf verschiedenen Ebenen erkennen und korrekt darauf reagieren.
- Sie können bei Sicherheitszwischenfällen schnell die bestgeeigneten Massnahmen ergreifen und dabei die gewonnenen Erkenntnisse rasch in die Verbesserung der IT-Sicherheit einfliessen lassen.
- Sie können die bei der Analyse gewonnenen Erkenntnisse direkt in die Verbesserung der IT-Security einbringen.
- Sie können kompetent in einem Security Operations Center, einem Computer Security Incident Response Team oder Computer Emergency Response Team mitarbeiten.

4 Voraussetzungen

- Sie besitzen gute Kenntnisse der Internet-Protokolle und beherrschen mindestens eine Skript- und/oder eine Programmiersprache. Sie gehen effizient mit Linux und Windows-Systemen um und kennen die verschiedenen Konfigurationsmöglichkeiten, sowohl für Clients wie auch für Server.
- Sie haben vertiefte Kenntnisse in der Architektur moderner Betriebssysteme, insbesondere im Bereich Filesysteme und Memory Management. Assembler-Grundkenntnisse sind eine ideale Voraussetzung für den Teil Reverse Engineering.
- Sie bringen gute IT-Vorkenntnisse im Rahmen einer Informatik- oder Wirtschaftsinformatik-Ausbildung mit. Insbesondere sind Erfahrungen in der Mitarbeit und Umsetzung von Informatikprojekten in den Bereichen IT-Infrastruktur, Netzwerk-Architektur oder IT-Security erwünscht.
- Der vorgängige Besuch des CAS SIPD oder äquivalente Kenntnisse werden empfohlen.
- Für das Studium der Fachliteratur und Kursunterlagen werden Englischkenntnisse vorausgesetzt.

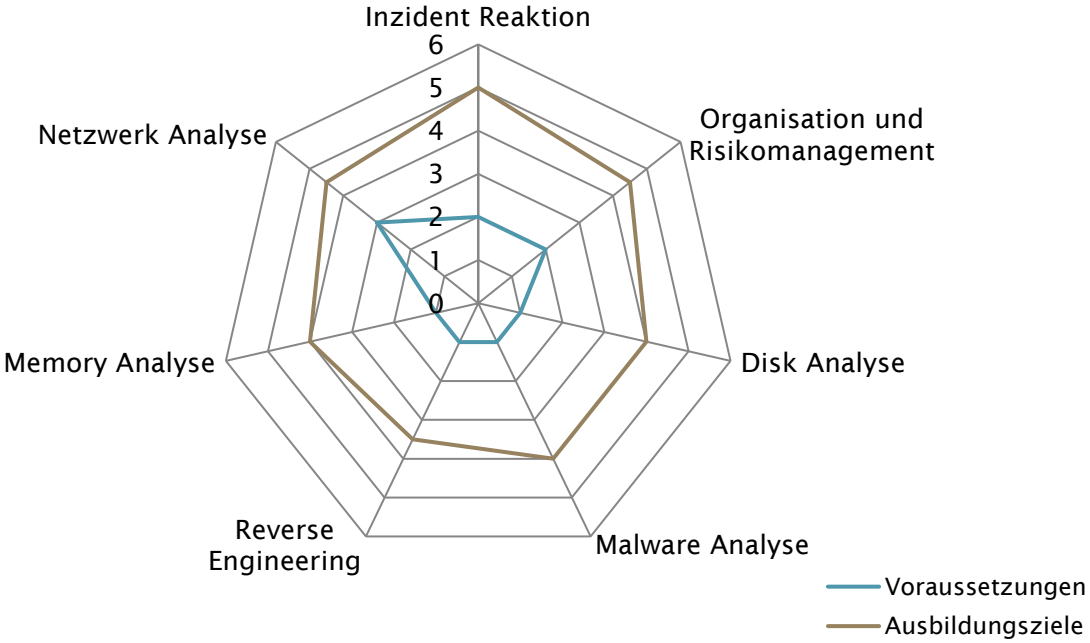
5 Unterrichtssprache

Die Unterrichtssprache ist Deutsch, die Unterlagen sind teilweise in Englisch.

6 Durchführungsort

Berner Fachhochschule, Weiterbildung, Aarbergstrasse 46 (Switzerland Innovation Park Biel/Bienne), 2503 Biel,
Telefon +41 31 848 31 11, E-Mail weiterbildung.ti@bfh.ch.

7 Kompetenzprofil



Kompetenzstufen

1. Kenntnisse/Wissen
2. Verstehen
3. Anwenden
4. Analyse
5. Synthese
6. Beurteilung

8 Kursübersicht

Kurs / Lehreinheit	Lektionen	Stunden	Dozierende
Begrüssung und Einführung	4		Reto Inversini
Analyse • Logfile-Analyse • Disk Forensik • Memory Forensik • Dynamische Malware-Analyse • Reverse Engineering • Threat Intelligence	84		 Heino Kronenberg Adrian Leuenberger Eduard Blenkers Marco Gfeller Andreas Greulich Reto Inversini
Reaktion • XDR / Remote-Forensik • Incident Reaction • Organisation Reaktion und Kommunikation • Attack Mitigation	48		 Stephan Berger Adrian Leuenberger Daniel Röthlisberger Reto Inversini
Workshop Threat Hunting	12		Stefan Egger Dominik Kuhn Heino Kronenberg
Organisation und Meilensteine Semesterarbeit	12		alle
Semester-/Projektarbeit		~ 90	alle
Total	**160**	**~ 90**	

Das CAS umfasst insgesamt 12 ECTS-Credits. Für die einzelnen Kurse ist entsprechend Zeit für Selbststudium, Prüfungsvorbereitung etc. einzurechnen.

9 Didaktik, Präsenz, Distance Learning

Didaktisch ist das CAS geprägt von einer hohen Interaktion zwischen Dozierenden und den Studierenden. Der Theorieteil des Unterrichts wird mit kleinen Aufgaben, Übungen und Diskussionen ergänzt. In der abschliessenden Semesterarbeit soll das im CAS erworbene Wissen an einem konkreten Fall aus dem Umfeld der Studierenden angewendet und die Ergebnisse der Arbeit innerhalb der Klasse in einer Schlusspräsentation weitergegeben werden.

Neben dem klassischen Präsenzunterricht im Klassenzimmer werden einzelne Kursteile auch im Fernunterricht per MS Teams gehalten oder in hybrider Form (Unterricht im Klassenzimmer mit Live-Übertragung per MS Teams) angeboten. Die gewählte Unterrichtsform orientiert sich dabei an den zu behandelnden Themen.

10 Kursbeschreibungen

Nachfolgend sind die einzelnen Kurse dieses Studiengangs beschrieben.
Der Begriff Kurs schliesst alle Veranstaltungstypen ein, es ist ein zusammenfassender Begriff für verschiedene Veranstaltungstypen wie Vorlesung, Lehrveranstaltung, Fallstudie, Living Case, Fach, Studienreise, Semesterarbeiten usw.

10.1 Einführung

Allgemein	Im Einführungsblock wird eine Übersicht über die aktuelle Bedrohungslage gegeben. Grundlegende, für dieses CAS wichtige Basiskenntnisse, werden mit einem kurzen Überblick zusammengefasst. Fehles Wissen ist anschliessend im Selbststudium zeitgerecht aufzufrischen.
Lernziele	– Kenntnis der aktuellen Bedrohungen – Basiskenntnisse der Operational Security
Themen und Inhalte	Übersicht der aktuellen Bedrohungen – Aktuelle Bedrohungen durch Cybercrime und staatliche Cyberangriffe – Schutz der eigenen Identität und der eigenen Arbeitsmittel
Vorkenntnisse	Grundlagenwissen: – Internet-Protokolle – Betriebssysteme – Programmierung Im speziellen werden die folgenden Themen für die nachfolgenden Unterrichtsblöcke als bekannt vorausgesetzt und sind im Bedarfsfall im Selbststudium aufzufrischen: – Gängige Internetprotokolle (TCP/IP, Routing, HTTP, SMTP, etc.) – Sniffing und Protokoll-Analyzer, Eigenschaften, Grenzen und Möglichkeiten – Betriebssystem-Theorie allgemein – Aufbau eines Betriebssystems – Prozess-, Memory- und Ressourcen-Management – Inter Process Communication und Interrupts – Shell Skripting – Verwendung von Regular Expressions – Grundkenntnisse x86-Assembler

Lehrmittel	– Skript / kommentiertes Folienset – Literaturempfehlungen Nr. 1, 2

10.2 Analyse

Allgemein	Einbrüche und Malware sind nach wie vor die grössten Bedrohungen für eine Firma, insbesondere, wenn Malware im Kontext von gezielten Angriffen eingesetzt wird. Um den Impact eines Security Incidents einschätzen zu können und um Gegenmassnahmen zu bestimmen, muss der/die CSIRT/CERT Mitarbeiter*in die Spuren des Einbruchs und von Malware finden, verstehen und analysieren können. Es gibt jedoch Situationen, bei denen mit Hilfe von dynamischer Analyse von Malware keine Informationen gewonnen werden können, z.B. weil die Malware nur auf bestimmten Geräten startet. In diesem Fall muss der Code mittels Reverse Engineering statisch analysiert werden.
Lernziele	– Logfile-Analyse: Vorstellen der Konzepte zu einer zentralisierten Logsammlung. Erkennen von Angriffsmustern in Logdaten. Kennenlernen der Tools auf der Command Line für das effiziente Suchen in Logdaten. Arbeiten mit Splunk als Tool zur Sammlung und Verarbeitung von Logdaten. Korrelation von verschiedenen Logdaten. – Disk-Analyse: Einführung in die Funktionsweise von Festplatten und Dateisystemen, sowie die Datenträgerforensik (Disk, Flash und SSD Speicher). Die Teilnehmenden erlernen und benutzen die Grundtechniken der Disk-Forensik im Kontext der Malware- und Intrusion-Analyse. – Memory-Analyse: Die Teilnehmende erlernen die Grundlagen der Memory-Analyse und sind in der Lage, diese Zwecks Detektion und Analyse von Malware einzusetzen. – Dynamische Malware-Analyse: Die Studierenden lernen die verschiedenen Malware-Typen kennen und können diese mittels der erlernten Techniken auf bestimmte Merkmale hin analysieren. Einführung in obfuskierte Java Scripts, Office-Dokumente mit obfuskierten Makros und PDF-Dokumente mit maliziösen Inhalten. Die Studierenden lernen verschiedene Tools kennen und anwenden, welche beim Analysieren und Deobfuskieren solcher Dokumente von Nutzen sind. – Reverse Engineering: Die Studierenden erhalten einen Einblick ins Reverse Engineering von Malware mit Hilfe von Debugger und Disassembler. – Threat Intelligence und Attribution: Die Studierenden lernen die wichtigsten Konzepte zur Erstellung und Interpretation von Threat Intelligence kennen und haben Kenntnis der Möglichkeiten und Grenzen der Attribution.

Themen und Inhalte	Logfile Analyse: – Zentralisierung von Logs mit Hilfe von Syslog – Logfile-Analyse auf der Commandline – Pattern Matching – Korrelationen – Einsatz von Splunk zur Logfile-Analyse Disk-Forensik: – Akquisition und Analyse eines Datenträgers – Erstellen von Zeitlinien über angelegte, geänderte und gelöschte Dateien – Suche von systemspezifischen Artefakten – Carving von gelöschten Dateien Memory Forensik: – Grundlagen des Memory-Aufbaus und dessen Inhalten – Memory-Akquisition und Analyse-Techniken und deren Anwendungsbereiche – Analysetools wie «Volatility» und «Redline» – Systemanalyse und Detektion von Anomalien mittels Memory Forensik Dynamische Malware-Analyse: – Dynamische Analyse mit verschiedenen Tools – Analysieren von «Malware traffic» – Aufbau und Kommunikation von Bot-Netzen – Verhaltensbasierte Analyse Reverse Engineering: – Einführung ins Reverse Engineering – Assemblergrundlagen x86, 32Bit (Vorkenntnisse sind von Vorteil) – Architektur x86 (Register, Memory Management, etc.) – Statische und dynamische Codeanalyse – Aufbau eines Windows EXE Files – Arbeiten mit Disassemblern (IDA, udis86) und Debuggern (OllyDbg, Immunity, WinDbg) – Grundlagen Kernel Debugging und der Windows API – Umgang mit Packern und Codeobfuskierung Threat Intelligence: – Generierung und Interpretation von Threat Intelligence – Indicators of Compromise (IOCs) und Techniques, Tactics and Procedures (TTPs) – Killchains – MICTIC Framework – Diamond Framework – Aufbereiten und Teilen von Informationen – Grundlagen, Möglichkeiten und Grenzen von Attribution Script/Dokument Analyse: – Analysieren/Deobfuskieren von Java Scripts von Hand – Analysieren/Deobfuskieren von Java Scripts mit verschiedenen Tools (Debugger/Interpreter) – Analyse von Office-Dokumenten mit maliziösen Inhalten

Vorkenntnisse	– Umfassende Systemkenntnisse (Grundlagen der Computerarchitektur, Betriebssysteme, etc.) sowie gängiger Netzwerktechnologien und deren Protokolle. – Python, Grundkenntnisse C. – Grundkenntnisse 32Bit x86 Assembler (siehe [6]), MASM-Syntax (Intel), ohne Extensions (Floating-Point, MMX, SSE, etc.)
Lehrmittel	– Skript / kommentiertes Folienset – Literaturempfehlungen Nr. 3, 4, 5, 6

10.3 Reaktion

Allgemein	Wird durch den Einsatz von entsprechenden Systemen und Werkzeugen ein Security Incident erkannt, muss darauf korrekt reagiert werden. Aus technischer Sicht ist es wichtig zu entscheiden, welche Massnahmen getroffen werden. Ergibt es Sinn, ein System sofort vom Netzwerk zu trennen oder soll man das infizierte System noch weiterlaufen lassen? Solche und ähnliche Fragen müssen im Falle eines Security Incidents oft unter hohem Zeitdruck entschieden werden. Aus organisatorischer Sicht muss die zeitgerechte Kommunikation mit den richtigen Stellen intern wie extern sichergestellt werden. Zudem müssen die vorgängig definierten Rollen bei der Reaktion auf einen Incident aktiviert und eingebunden werden. Um ähnliche Incidents zukünftig zu verhindern, müssen die Lehren daraus gezogen werden. Nur so lässt sich das Risiko einer Wiederholung minimieren.
Lernziele	– XDR / Remote Forensik: Die Studierenden kennen die Möglichkeiten von Endpoint Detection and Response und Remote Forensik Werkzeugen. Sie lernen den Einsatz von Velociraptor und SysMon zur Erkennung und Analyse von Angriffen und zur Extraktion von weiteren forensischen Artefakten. – Incident Reaction: (in Koordination zwischen Daniel Röthlisberger und Reto Inversini) Die Studierenden kennen die Voraussetzungen für eine effiziente Reaktion, welche Daten, Strukturen und Informationen vorhanden sein müssen. Sie wissen, wie eine Reaktion innerhalb der Firma, aber auch in Zusammenarbeit mit Partnerorganisationen effizient durchgeführt wird. – Organisation, Kommunikation und IT Risikomanagement: Die Studierenden kennen die Prozesse, welche für den reibungslosen Betrieb eines CSIRT/CERT/SOC notwendig sind. Sie wissen, wer im Falle eines Incidents beigezogen und mit wem kommuniziert werden muss. Sie besitzen die Grundlagen, um die Lehren aus einem Incident zu ziehen und damit das Risiko einer Wiederholung zu minimieren. Sie kennen zudem die Schnittstelle zum IT Risikomanagement und können dieses mit aktuellen Bedrohungsszenarien versorgen. Auf der Basis von erkannten Risiken wissen die Studierenden, wie sie ihre Detektions- und Abwehrmassnahmen verbessern können.

	– Attack Mitigation: Die Studierenden kennen die möglichen technischen Reaktionen (z.B. Tracking, Blocking, Sinkholing) auf einen Incident und erhalten die Grundlagen, um sich für eine Reaktion zu entscheiden. Dabei werden beispielhaft zwei wichtige Phänomene betrachtet, Ransomware und DDoS-Angriffe und entsprechende Reaktionsmöglichkeiten diskutiert.
Themen und Inhalte	XDR: - Einführung in typische XDR- und Remote-Forensik-Werkzeuge - Limitationen von XDR-Produkten und klassische Umgehungsmöglichkeiten - Erkennen und Analyse von Angriffen und Extraktion von IOCs aus verschiedenen forensischen Artefakten - Arbeiten mit Velociraptor als typisches Remote-Forensik-Werkzeug Incident Response: - Diskussion der nötigen Vorbereitungen innerhalb der Firma, um bereit zu sein, Incidents effizient und zielgerichtet zu behandeln - Kennenlernen von Feeds und Werkzeugen für den Incident Response - After Action Review - Fallbeispiel Organisation und Kommunikation: - Entscheidungsfindung während eines Incidents - Kommunikation im Incidentfall - Rollen & Verantwortlichkeiten - Organisatorische Schnittstellen - Risikomanagement Attack Mitigation: - Technische Mitigation von Incidents - Zwei Fallbeispiele, Ransomware und DDoS - Techniken der DDoS Mitigation (Traffic Scrubbing, Remotely Triggered Blackholing, BCP 38, Flow Specs)
Lehrmittel	– Skript / Folienset – Literaturempfehlung Nr. 7

10.4 Workshop

Allgemein	Workshop zu Threat Hunting (Theat Intel Usage)
Lernziele	Die Studierenden erhalten anhand von kleinen Aufgaben die Möglichkeit, den gelernten Stoff am Beispiel anzuwenden. Die Teilnehmenden werden dabei anhand eines Arbeitsblattes durch die einzelnen Teilaufgaben geführt.
Themen und Inhalte	Anwenden der wichtigsten Technologien wie: – passiveDNS, passiveSSL, – Malware Information Sharing Platform – Quellen (Virustotal, PassiveTotal, Shodan, Spamhaus). – Arbeiten und Einbinden von Threatfeeds in die Detektionsinfrastruktur.

Lehrmittel	– Die im Kurs abgegebenen Skripte – – Ein «Spickzettel» für die wichtigsten Tools – – Live-System / VMware Image mit der Testumgebung – «Google is your Friend»

10.5 Projektarbeit

Allgemein	Die Projektarbeiten sind Einzel- oder Gruppen-Arbeiten aus dem Arbeitsumfeld der Studierenden. Gruppenarbeiten sind wo immer möglich erwünscht und je nach Rahmenbedingungen meist von Vorteil. Der nominelle Aufwand liegt bei 90 Arbeitsstunden pro Gruppenmitglied, kann je nach Vorbereitungsphase und Komplexität der Aufgabenstellung aber auch leicht höher sein. Falls aus Sicht des Auftraggebers notwendig, können die Ergebnisse der Semesterarbeiten vertraulich behandelt werden. Massgebend für die Rahmenbedingungen ist das Studienreglement. Die Vertraulichkeit darf den didaktischen Rahmen nicht behindern: Präsentationen und Diskussionen über das gewählte Thema müssen im Rahmen der Klasse möglich sein.
Zielsetzung und Thema	In der Semesterarbeit befassen sich die Teilnehmenden mit einem Projekt (ev. Teilprojekt) oder einer Fragestellung aus ihrer Firma. Mit dem gewählten Thema vertiefen die Studierenden die im Studium erlernten Methoden und wenden diese an einer konkreten Fragestellung in der Praxis an. Themen von Semesterarbeiten können beispielsweise sein: – Analyse eines sicherheitsrelevanten Vorfalls im eigenen oder einem auftraggebenden Betrieb. – Untersuchung und Nachstellung neuer Angriffsmethoden. – Analyse vorhandener Infrastruktur und anhand der gewonnenen Informationen ein geeignetes Konzept zur Härtung der Systeme erarbeitet. – Analyse und Vergleich von Industrie IoT-Komponenten bezüglich Ihrer IT-Sicherheit mit dem Ziel, die OT-Security zu verbessern. – Sicherheits-Vorfälle oder -Projekte der Dozierenden

Ablauf	Die Semesterarbeit umfasst ca. 90h Arbeitsleistung pro Studierende und beinhaltet die folgenden Meilensteine (siehe auch Zeitplan):
	1. In der Firma ein Thema suchen und finden sowie einen Ansprechpartner/Betreuer in der Firma definieren.
	2. Erstellen einer Projektskizze (Wordvorlage vorhanden).
	3. Die Projektskizze umfasst eine ein- bis maximal zweiseitige Aufgabenstellung mit folgenden Elementen:
	  1. Titel
	  2. Umfeld
	  3. Problemstellung
	  4. Lösungsansatz (Vorgehen, Methoden)
	  5. Angestrebte Ergebnisse und Ziele
	  6. Name und Kontaktadressen aller Gruppenmitglieder und der Ansprechpartner*innen/Betreuer*innen der Firma

	4. Individuelle Kurzpräsentation (10') und Diskussion (10') des gewählten Themas an der Schule vor einem Expert*innen- und Dozierenden-Gremium. 5. Eventuelle Ergänzung oder Überarbeitung der Projektskizze gemäss Feedback an der Präsentation. 6. Zuordnung eine/s/r Expert*in durch die Schule für die Begleitung der Semesterarbeit. 7. Durchführung der Arbeit in eigener Terminplanung. 8. Ca. 2-3 Meetings mit dem/der Expert*in. – Projektskizze besprechen / Kick-Off. – bei Bedarf: Zwischenreview / Beratung. – Schlusspräsentation vor Expert*innen- und Dozierenden-Gremium. – Dauer: 10'-15' und Diskussion: 10'-15' pro Arbeit. 9. Abgabe des Berichtes auf der Studienplattform oder nach Absprache per E-Mail an den/die Expert*in. 10. Beurteilung durch den/die Expert*in.
Ergebnis und Bewertung	Der Bericht ist in elektronischer Form, als PDF-Dokument dem/der bewertenden Expert*in und der CAS-Leitung über die Studienplattform (aktuell Moodle) abzugeben. Der Bericht umfasst ca. 20 Seiten. Der Source Code ist, soweit für die Projektbeurteilung notwendig, als Anhang mitzuliefern. Die Semesterarbeit wird nach den folgenden Kriterien bewertet: – Themeneingabe Projektskizze rechtzeitig und vollständig eingereicht. Themenpräsentation sorgfältig vorbereitet. Idee oder Aufgabe durchdacht und abgegrenzt, Quellen recherchiert, Rahmenbedingungen definiert, Teilziele priorisiert. – Methodik und Ausführung Gewählte Methode(n) systematisch und korrekt angewendet. Kreativ und agil in der Ausführung. Entscheidungen präzis begründet. – Ergebnis Nachvollziehbares und dokumentiertes Ergebnis. Aufgabenstellung erfüllt. Ergebnisse validiert, getestet, verifiziert. Vergleich von Zielsetzung und Ergebnis vorgenommen. Learnings und Ausblick vorhanden. – Bericht und Dokumentation Vollständig und verständlich. Rechtschreibung korrekt. Kapiteleinteilung sinnvoll. Angemessene Darstellung. Grafiken auf das Wesentliche reduziert und beschriftet. – Schlusspräsentation Roter Faden, logisches Vorgehen, klare Aussagen. Identifikation mit dem Thema spür- und erkennbar. Professionelle Präsentationstechnik, Zeitvorgaben genutzt und eingehalten. Fragen präzis und sicher beantwortet. Die aufgeführten Kriterien sind durch den/die Expert*in entsprechend dem bearbeiteten Thema und dem Ablauf der Arbeit in ihrem Gewicht anpassbar.

11 Kompetenznachweis

Für die Anrechnung der 12 ECTS-Credits ist das erfolgreiche Bestehen der Qualifikationsnachweise (Prüfungen, Projektarbeiten) erforderlich, gemäss folgender Aufstellung:

Kompetenznachweis	Gewicht	Art der Qualifikation	Erfolgsquote
Einführung		keine	
Analyse	4	Gruppenarbeit / Prüfung	0 – 100 %
Reaktion	2	Gruppenarbeit / Prüfung	0 – 100 %
Workshop Threat Hunting		keine	
Fallstudie	4	Bewertete Projektarbeit	0 - 100 %
Gesamtgewicht / Erfolgsquote	**10**		0 – 100 %

Der gewichtete Mittelwert der Erfolgsquoten der einzelnen Kompetenznachweise wird in eine Note zwischen 3 und 6 umgerechnet. Die Note 3 (gemittelte Erfolgsquote weniger als 50%) ist ungenügend. Die Noten 4, 4.5, 5, 5.5 und 6 (gemittelte Erfolgsquote zwischen 50% und 100%) sind genügend.

12 Lehrmittel

Ergänzende Lehrmittel sind Empfehlungen, um den Stoff zu vertiefen oder zu erweitern. Die Beschaffung liegt im Ermessen der Studierenden:

Nr	Titel	Autoren	Verlag	Jahr	ISBN Nr.
1	LINUX - Das umfassende Handbuch	Johannes Plötner Steffen Wendzel	Rheinwerk «openbook»	2012	978-3-8362-1822-1 Download
2	RegEx-Tutorial von Max Kleiner	Max Kleiner	Maxbox	2014	Download
3	Malware Analyst's Cookbook and DVD: Tools and Techniques for Fighting Malicious Code. Ligh et al.	Michael Ligh, Steven Adair, Blake Hartstein, Matthew Richard	John Wiley & Sons, Ltd.	2014	0470613033 978-0470613030
4	Practical Malware Analysis: The Hands-On Guide to Dissecting Malicious Software, Sikorski et al.	Andrew Honig Michael Sikorski	No Starch Press, US	2012	1593272901 978-1593272906
5	Assembly Language for x86 Processors (Insbesondere Kap. 1-4, 6-7)	Kip R. Irvine	Pearson	2014	0133769402 978-0133769401
6	X86 Assembly	Wikibooks	Wikibooks	2013	Download
7	ENISA (European Network and Information Security Agency): Good Practice Guide for Incident Management		ENISA	2010	Download

13 Dozierende

Vorname Name	Firma	E-Mail
Stephan Berger	Infoguard	stephan.berger@bfh.ch
Eduard Blenkers	BLS	eduard.blenkers@bfh.ch
Stefan Egger	Bundesamt für Informatik und Telekommunikation	stefan.egger@bfh.ch
Marco Gfeller	NCSC	marco.gfeller@bfh.ch
Andreas Greulich	NCSC	andreas.greulich.1@bfh.ch
Reto Inversini	SBB	reto.inversini@bfh.ch
Heino Kronenberg	Bundesamt für Informatik und Telekommunikation	heino.kronenberg@bfh.ch
Dominik Kuhn	Bundesamt für Informatik und Telekommunikation	dominik.kuhn@bfh.ch
Adrian Leuenberger	VBS	adrian.leuenberger@bfh.ch
Daniel Röthlisberger	Privatwirtschaft	daniel.roethlisberger@bfh.ch

14 Organisation

CAS-Leitung:
Reto Inversini und Rolf Lanz
Tel: +41 32 321 61 29 und 031 848 32 73
E-Mail: reto.inversini@bfh.ch und rolf.lanz@bfh.ch

CAS-Administration:
Andrea Moser
Tel: +41 31 84 83 211
E-Mail: andrea.moser@bfh.ch

Während der Durchführung des CAS können sich Anpassungen bezüglich Inhalten, Lernzielen, Dozierenden und Kompetenznachweisen ergeben. Es liegt in der Kompetenz der Dozierenden und der Studienleitung, aufgrund der aktuellen Entwicklungen in einem Fachgebiet, der konkreten Vorkenntnisse und Interessenslage der Teilnehmenden, sowie aus didaktischen und organisatorischen Gründen Anpassungen im Ablauf eines CAS vorzunehmen.

Berner Fachhochschule
Technik und Informatik
Weiterbildung
Aarbergstrasse 46 (Switzerland Innovation Park Biel/Bienne)
2503 Biel

Telefon +41 31 848 31 11
E-Mail: weiterbildung.ti@bfh.ch

bfh.ch/ti/weiterbildung
bfh.ch/cas-siar