

Eric Amberg | Daniel Schmid

Hacking

Der umfassende Praxis-Guide

Inkl. Prüfungsvorbereitung zum CEHv12

3. Auflage



Inhaltsverzeichnis

	Einleitung.....	29
	Danksagung.....	36
Teil I	Grundlagen und Arbeitsumgebung	37
1	Grundlagen Hacking und Penetration Testing.....	39
1.1	Was ist Hacking?.....	40
1.2	Die verschiedenen Hacker-Typen	41
1.3	Motive und Absichten eines Hackers	43
1.3.1	Das Motiv.....	43
1.3.2	Ziel des Angriffs	44
1.4	Ethical Hacking.....	45
1.5	Der Certified Ethical Hacker (CEHv12)	46
1.5.1	Was steckt dahinter?.....	47
1.5.2	Die CEHv12-Prüfung im Detail.....	48
1.6	Die Schutzziele: Was wird angegriffen?	49
1.6.1	Vertraulichkeit.....	49
1.6.2	Integrität	51
1.6.3	Verfügbarkeit.....	53
1.6.4	Authentizität und Nicht-Abstreitbarkeit	54
1.6.5	Die Quadratur des Kreises	54
1.7	Systematischer Ablauf eines Hacking-Angriffs	56
1.7.1	Phasen eines echten Angriffs	56
1.7.2	Unterschied zum Penetration Testing	58
1.8	Praktische Hacking-Beispiele	60
1.8.1	Angriff auf den Deutschen Bundestag	60
1.8.2	Stuxnet – der genialste Wurm aller Zeiten.....	61
1.8.3	Angriff auf heise.de mittels Emotet.....	61
1.9	Zusammenfassung und Prüfungstipps	62
1.9.1	Zusammenfassung und Weiterführendes	62
1.9.2	CEH-Prüfungstipps	62
1.9.3	Fragen zur CEH-Prüfungsvorbereitung	63
2	Die Arbeitsumgebung einrichten.....	65
2.1	Virtualisierungssoftware.....	66
2.1.1	Software-Alternativen.....	67
2.1.2	Bereitstellung von VirtualBox	68
2.2	Die Laborumgebung in der Übersicht.....	70

2.3	Kali Linux	71
2.3.1	Einführung	71
2.3.2	Download von Kali Linux als ISO-Image	72
2.3.3	Kali Linux als VirtualBox-Installation	73
2.3.4	Kali Linux optimieren	77
2.4	Windows 10 als Hacking-Plattform	81
2.4.1	Download von Windows 10	81
2.4.2	Windows-10-Installation in VirtualBox	82
2.4.3	Windows 10 – Spyware inklusive	82
2.4.4	Gasterweiterungen installieren	83
2.5	Übungsumgebung und Zielscheiben einrichten	84
2.5.1	Metasploitable	85
2.5.2	Die Netzwerkumgebung in VirtualBox anpassen	87
2.5.3	Multifunktionsserver unter Linux	90
2.5.4	Windows XP und ältere Windows-Betriebssysteme	90
2.5.5	Eine Windows-Netzwerkumgebung aufbauen	91
2.6	Zusammenfassung und Weiterführendes	91
3	Einführung in Kali Linux	93
3.1	Ein erster Rundgang	93
3.1.1	Überblick über den Desktop	94
3.1.2	Das Startmenü	97
3.1.3	Der Dateimanager	99
3.1.4	Systemeinstellungen und -Tools	101
3.2	Workshop: Die wichtigsten Linux-Befehle	102
3.2.1	Orientierung und Benutzerwechsel	103
3.2.2	Von Skripts und Dateiberechtigungen	105
3.2.3	Arbeiten mit Root-Rechten	107
3.2.4	Das Dateisystem und die Pfade	110
3.2.5	Dateien und Verzeichnisse erstellen, kopieren, löschen etc.	111
3.2.6	Dateien anzeigen	112
3.2.7	Dateien finden und durchsuchen	114
3.2.8	Die Man-Pages: Hilfe zur Selbsthilfe	116
3.2.9	Dienste starten und überprüfen	117
3.3	Die Netzwerk-Konfiguration anzeigen und anpassen	119
3.3.1	IP-Adresse anzeigen	119
3.3.2	Routing-Tabelle anzeigen	120
3.3.3	DNS-Server anzeigen	120
3.3.4	Konfiguration der Schnittstellen	121
3.4	Software-Installation und -Update	123
3.4.1	Die Paketlisten aktualisieren	123
3.4.2	Installation von Software-Paketen	124
3.4.3	Software suchen	124
3.4.4	Entfernen von Software-Paketen	125
3.5	Zusammenfassung und Prüfungstipps	126

3.5.1	Zusammenfassung und Weiterführendes	126
3.5.2	CEH-Prüfungstipps	126
3.5.3	Fragen zur CEH-Prüfungsvorbereitung	126
4	Anonym bleiben und sicher kommunizieren	129
4.1	Von Brotkrumen und Leuchtspuren	129
4.2	Proxy-Server – schon mal ein Anfang	131
4.2.1	Grundlagen – so arbeiten Proxys	131
4.2.2	Einen Proxy-Server nutzen	132
4.2.3	Öffentliche Proxys in der Praxis	134
4.2.4	Vor- und Nachteile von Proxy-Servern	135
4.2.5	Proxy-Verwaltung mit FoxyProxy	136
4.3	VPN, SSH und Socks – so bleiben Black Hats anonym	137
4.3.1	Virtual Private Networks (VPN)	137
4.3.2	SSH-Tunnel	139
4.3.3	SOCKS-Proxy	141
4.3.4	Kaskadierung für höchste Anonymität und Vertraulichkeit	145
4.3.5	Proxifier – Für unwillige Programme	146
4.4	Deep Web und Darknet – im Untergrund unterwegs	146
4.4.1	Wo geht es bitte zum Untergrund?	146
4.4.2	Das Tor-Netzwerk	147
4.4.3	Das Freenet Project	153
4.4.4	Die Linux-Distribution Tails	154
4.5	Anonym mobil unterwegs	156
4.5.1	Mobile Proxy-Tools und Anonymizer	156
4.6	Sonstige Sicherheitsmaßnahmen	157
4.6.1	System säubern mit dem CCleaner	158
4.6.2	G-Zapper: Cookies unter Kontrolle	159
4.7	Zusammenfassung und Prüfungstipps	159
4.7.1	Zusammenfassung und Weiterführendes	159
4.7.2	CEH-Prüfungstipps	160
4.7.3	Fragen zur CEH-Prüfungsvorbereitung	161
5	Kryptografie und ihre Schwachstellen	163
5.1	Einführung in die Krypto-Algorithmen	164
5.1.1	Alice und Bob ... und Mallory	164
5.1.2	Algorithmen und Schlüssel	165
5.1.3	Das CrypTool – Kryptografie praktisch erfahren	166
5.2	Die symmetrische Verschlüsselung	167
5.2.1	Grundlagen der symmetrischen Verfahren	167
5.2.2	Verschlüsselung im alten Rom: Die Cäsar-Chiffre	168
5.2.3	Strom- und Blockchiffre	168
5.2.4	Vor- und Nachteile von symmetrischen Algorithmen	169
5.2.5	Wichtige symmetrische Algorithmen	169
5.2.6	Symmetrische Verschlüsselung in der Praxis	172

5.3	Die asymmetrische Verschlüsselung	175
5.3.1	Wo liegt das Problem?	175
5.3.2	Der private und der öffentliche Schlüssel	176
5.3.3	Der Schlüsselaustausch.	176
5.3.4	Authentizitätsprüfung.	178
5.3.5	Wichtige asymmetrische Algorithmen.	179
5.4	Hash-Algorithmen	181
5.4.1	Ein digitaler Fingerabdruck	181
5.4.2	Integritätsprüfung mit Hashwerten	182
5.4.3	Wichtige Hash-Algorithmen.	185
5.5	Digitale Signaturen	187
5.5.1	Das Prinzip der digitalen Signatur	187
5.5.2	Wichtige Verfahren der digitalen Signatur	189
5.6	Public-Key-Infrastrukturen (PKI)	189
5.6.1	Das Prinzip von PKI	190
5.6.2	Digitale Zertifikate.	190
5.6.3	Zertifikate und PKI in der Praxis	191
5.6.4	Zertifikatssperrlisten und OCSP	195
5.7	Virtual Private Networks (VPN).	197
5.7.1	IPsec-VPNs.	198
5.7.2	SSL-VPNs	199
5.8	Angriffe auf kryptografische Systeme.	201
5.8.1	Methodologie der Kryptoanalyse.	201
5.8.2	Der Heartbleed-Angriff	203
5.8.3	Des Poodles Kern – der Poodle-Angriff	205
5.9	Kryptotrojaner und Ransomware	206
5.9.1	WannaCry	206
5.9.2	Petya	207
5.9.3	Locky	208
5.9.4	Schutz- und Gegenmaßnahmen	208
5.10	Zusammenfassung und Prüfungstipps	209
5.10.1	Zusammenfassung und Weiterführendes	209
5.10.2	CEH-Prüfungstipps	209
5.10.3	Fragen zur CEH-Prüfungsvorbereitung.	209

Teil II Informationsbeschaffung **213**

6	Informationsbeschaffung – Footprinting & Reconnaissance	217
6.1	Ich will hacken, wozu die langweilige Informationssuche?	218
6.1.1	Worum geht es bei der Informationsbeschaffung?	219
6.1.2	Welche Informationen sind relevant?	219
6.2	Suchmaschinen und Informationsportale nutzen.	221
6.2.1	Reguläre Suchmaschinen	221
6.2.2	Netcraft: Nach öffentlichen und zugriffsbeschränkten Seiten suchen	222

6.2.3	WayBack Machine – das Internet-Archiv	223
6.2.4	Shodan.	224
6.2.5	Map-Anbieter: Mal von oben betrachtet	225
6.2.6	Personen-Suchmaschinen	226
6.2.7	Jobsuchmaschinen als Informationsquelle.	226
6.2.8	Arbeitgeber-Bewertungsportale	227
6.3	Google-Hacking.	227
6.3.1	Was steckt dahinter?	227
6.3.2	Wichtige Suchoperatoren.	228
6.3.3	Die Google Hacking Database (GHDB)	228
6.4	Social-Media-Footprinting	229
6.4.1	Wo suchen wir?	230
6.4.2	Was suchen wir?	230
6.4.3	Wie suchen wir?	230
6.5	Technische Analysen.	231
6.5.1	Whois.	231
6.5.2	DNS – Das Domain Name System	233
6.5.3	E-Mail-Footprinting	237
6.5.4	Website-Footprinting	239
6.5.5	Dokumente analysieren mit Metagoofil	240
6.6	Recon-ng – das Web-Reconnaissance-Framework	241
6.6.1	Die ersten Schritte mit Recon-ng.	241
6.6.2	Ein Modul installieren und laden	243
6.6.3	Wie geht es weiter?	245
6.7	Maltego – Zusammenhänge visualisieren	245
6.7.1	Einführung in Maltego.	245
6.7.2	Maltego starten	246
6.7.3	Mit Maltego arbeiten	247
6.7.4	Der Transform Hub	250
6.8	Gegenmaßnahmen gegen Footprinting	250
6.9	Zusammenfassung und Prüfungstipps.	251
6.9.1	Zusammenfassung und Weiterführendes	251
6.9.2	CEH-Prüfungstipps	252
6.9.3	Fragen zur CEH-Prüfungsvorbereitung	252
7	Scanning – das Netzwerk unter der Lupe	255
7.1	Scanning – Überblick und Methoden	255
7.1.1	Die Scanning-Phase	256
7.1.2	Ziel des Scanning-Prozesses	256
7.1.3	Scanning-Methoden	256
7.2	TCP/IP-Essentials	257
7.2.1	Das OSI-Netzwerk-Referenzmodell.	257
7.2.2	ARP, Switch & Co. – Layer-2-Technologien	259
7.2.3	Das Internet Protocol (IPv4)	259
7.2.4	Das Internet Control Message Protocol (ICMP).	260

7.2.5	Das User Datagram Protocol (UDP)	261
7.2.6	Das Transmission Control Protocol (TCP)	262
7.3	Nmap – DER Portscanner	263
7.3.1	Host Discovery	264
7.3.2	Normale Portscans	267
7.3.3	Zu scannende Ports festlegen	269
7.3.4	Besondere Portscans	270
7.3.5	Dienst- und Versionserkennung	272
7.3.6	Betriebssystem-Erkennung	273
7.3.7	Firewall/IDS-Vermeidung (Evasion)	273
7.3.8	Ausgabe-Optionen	274
7.3.9	Die Nmap Scripting Engine (NSE)	275
7.3.10	Weitere wichtige Optionen	276
7.3.11	Zenmap	277
7.4	Scannen mit Metasploit	278
7.4.1	Was ist Metasploit?	278
7.4.2	Erste Schritte mit Metasploit (MSF)	278
7.4.3	Nmap in Metasploit nutzen	282
7.5	Weitere Tools und Verfahren	284
7.5.1	Paketerstellung und Scanning mit hping3	284
7.5.2	Weitere Packet-Crafting-Tools	286
7.5.3	Banner Grabbing mit Telnet und Netcat	286
7.5.4	Scannen von IPv6-Netzwerken	288
7.6	Gegenmaßnahmen gegen Portscanning und Banner Grabbing	289
7.7	Zusammenfassung und Prüfungstipps	290
7.7.1	Zusammenfassung und Weiterführendes	290
7.7.2	CEH-Prüfungstipps	290
7.7.3	Fragen zur CEH-Prüfungsvorbereitung	291
8	Enumeration – welche Ressourcen sind verfügbar?	295
8.1	Was wollen wir mit Enumeration erreichen?	295
8.2	NetBIOS- und SMB-Enumeration	296
8.2.1	Die Protokolle NetBIOS und SMB	296
8.2.2	Der Enumeration-Prozess	298
8.3	SNMP-Enumeration	303
8.3.1	SNMP-Grundlagen	304
8.3.2	SNMP-Agents identifizieren	306
8.3.3	Enumeration-Tools nutzen	307
8.4	LDAP-Enumeration	312
8.4.1	LDAP- und AD-Grundlagen	312
8.4.2	Der Enumeration-Prozess	314
8.5	SMTP-Enumeration	316
8.5.1	SMTP-Grundlagen	316
8.5.2	Der Enumeration-Prozess	317
8.6	NTP-Enumeration	320

8.6.1	Funktionsweise von NTP	320
8.6.2	Der Enumeration-Prozess	320
8.7	DNS-Enumeration.	322
8.7.1	NFS-Enumeration.	327
8.7.2	Weitere Enumeration-Techniken	328
8.8	Schutzmaßnahmen gegen Enumeration.	328
8.9	Zusammenfassung und Prüfungstipps.	331
8.9.1	Zusammenfassung und Weiterführendes	331
8.9.2	CEH-Prüfungstipps	331
8.9.3	Fragen zur CEH-Prüfungsvorbereitung	332
9	Vulnerability-Scanning und Schwachstellenanalyse.	335
9.1	Was steckt hinter Vulnerability-Scanning?	335
9.1.1	Vulnerabilities und Exploits.	336
9.1.2	Common Vulnerabilities and Exposures (CVE)	336
9.1.3	CVE- und Exploit-Datenbanken.	338
9.1.4	Vulnerability-Scanner.	339
9.2	Vulnerability-Scanning mit Nmap	341
9.2.1	Die Kategorie »vuln«	341
9.2.2	Die passenden Skripts einsetzen	341
9.3	Nessus	344
9.3.1	Installation von Nessus	344
9.3.2	Vulnerability-Scanning mit Nessus	345
9.3.3	Nessus versus OpenVAS	349
9.4	Rapid 7 Nexpose	350
9.5	Vulnerability-Scanning in der Praxis	351
9.5.1	Vulnerability-Assessments	351
9.5.2	Einsatz von Vulnerability-Scannern im Ethical Hacking.	352
9.5.3	Credentialed Scan vs. Remote Scan.	353
9.5.4	Verifizieren der Schwachstelle.	354
9.5.5	Exploits zum Testen von Schwachstellen	354
9.5.6	Spezialisierte Scanner.	355
9.6	Zusammenfassung und Prüfungstipps.	355
9.6.1	Zusammenfassung und Weiterführendes	355
9.6.2	CEH-Prüfungstipps	356
9.6.3	Fragen zur CEH-Prüfungsvorbereitung	356
Teil III	Systeme angreifen	359
10	Password Hacking.	365
10.1	Zugriffsschutz mit Passwörtern und anderen Methoden	365
10.2	Angriffsvektoren auf Passwörter	367
10.2.1	Nicht elektronische Angriffe	367
10.2.2	Aktive Online-Angriffe	367

10.2.3	Passive Online-Angriffe	368
10.2.4	Offline-Angriffe	368
10.3	Password Guessing und Password Recovery	368
10.3.1	Grundlagen des Password Guessings	369
10.3.2	Default-Passwörter	370
10.3.3	Password Recovery unter Windows	372
10.3.4	Password Recovery für Linux	378
10.3.5	Password Recovery auf Cisco-Routern	379
10.4	Die Windows-Authentifizierung	381
10.4.1	Die SAM-Datenbank	381
10.4.2	LM und NTLM	381
10.4.3	Kerberos	382
10.4.4	NTLM-Hashes auslesen mit FGDump	386
10.5	Die Linux-Authentifizierung	388
10.5.1	Speicherorte der Login-Daten	388
10.5.2	Passwort-Hashes unter Linux	389
10.5.3	Der Salt – Passwort-Hashes »salzen«	390
10.5.4	Wie gelangen wir an die Passwort-Hashes?	391
10.6	Passwort-Hashes angreifen	392
10.6.1	Angriffsvektoren auf Passwort-Hashes	392
10.6.2	Pass the Hash (PTH)	396
10.6.3	Wortlisten erstellen	397
10.6.4	L0phtcrack	402
10.6.5	John the Ripper	404
10.6.6	Hashcat	406
10.6.7	Cain & Abel	406
10.7	Online-Angriffe auf Passwörter	407
10.7.1	Grundlegende Problematik	407
10.7.2	Medusa	407
10.7.3	Hydra	409
10.7.4	Ncrack	410
10.8	Distributed Network Attack (DNA)	412
10.8.1	Funktionsweise	412
10.8.2	ElcomSoft Distributed Password Recovery	413
10.9	Schutzmaßnahmen gegen Password Hacking	413
10.10	Zusammenfassung und Prüfungstipps	415
10.10.1	Zusammenfassung und Weiterführendes	415
10.10.2	CEH-Prüfungstipps	415
10.10.3	Fragen zur CEH-Prüfungsvorbereitung	416
11	Shells und Post-Exploitation	417
11.1	Remote-Zugriff mit Shell und Backdoor	417
11.1.1	Einführung in Shells und Backdoors	418
11.1.2	Netcat und Ncat – Einführung	420
11.1.3	Grundlegende Funktionsweise von Netcat und Ncat	421

11.1.4	Eine Bind-Shell bereitstellen	424
11.1.5	Eine Reverse-Shell bereitstellen	426
11.1.6	Wo stehen wir jetzt?	427
11.2	Grundlagen Privilegien-Eskalation	427
11.2.1	Vertikale Rechteerweiterung	428
11.2.2	Horizontale Rechteerweiterung	428
11.2.3	Rechte von Programmen	428
11.3	Mit Privilegien-Eskalation zur Root-Shell	429
11.3.1	Reverse-Shell durch DistCC-Exploit	429
11.3.2	Bereitstellung eines Post-Exploits	430
11.3.3	Mit Metasploit-Multi-Handler zur Root-Shell	434
11.4	Meterpreter – die Luxus-Shell für Hacker	435
11.4.1	Exploits und Payload	435
11.4.2	Einführung in Meterpreter	436
11.4.3	Meterpreter-Shell in der Praxis	438
11.4.4	Eine Meterpreter-Shell für Windows erstellen	440
11.4.5	Externe Module in Meterpreter laden	443
11.5	Privilegien-Eskalation in einer Windows Domäne	444
11.5.1	Das Szenario	445
11.5.2	Ermittlung des Domain-Controllers	445
11.5.3	Privilegien-Eskalation durchführen	446
11.6	Verteidigungsmaßnahmen gegen Privilegien-Eskalation	447
11.7	Zusammenfassung und Prüfungstipps	448
11.7.1	Zusammenfassung und Weiterführendes	448
11.7.2	CEH-Prüfungstipps	449
11.7.3	Fragen zur CEH-Prüfungsvorbereitung	449
12	Mit Malware das System übernehmen	451
12.1	Malware-Grundlagen	452
12.1.1	Typische Malware-Kategorien	452
12.1.2	Wie gelangt Malware auf das Opfer-System?	455
12.1.3	Eine selbst erstellte Malware	456
12.2	Viren und Würmer	457
12.2.1	Was ist ein Computervirus?	457
12.2.2	Was ist ein Computerwurm?	459
12.2.3	Einen Makro-Virus erstellen	460
12.3	Trojanische Pferde in der Praxis	465
12.3.1	Trojaner-Typen	465
12.3.2	Einen Trojaner selbst bauen	467
12.3.3	Viren- und Trojaner-Baukästen	470
12.4	Malware tarnen und vor Entdeckung schützen	472
12.4.1	Grundlagen der Tarnung von Payload	473
12.4.2	Encoder einsetzen	475
12.4.3	Payload mit Hyperion verschlüsseln	478
12.4.4	Das Veil-Framework	479

12.4.5	Shellter AV Evasion	480
12.4.6	Fileless Malware	481
12.5	Rootkits	482
12.5.1	Grundlagen der Rootkits	483
12.5.2	Kernel-Rootkits	484
12.5.3	Userland-Rootkits	484
12.5.4	Rootkit-Beispiele	484
12.5.5	Rootkits entdecken und entfernen	485
12.6	Covert Channel	486
12.6.1	ICMP-Tunneling	486
12.6.2	NTFS Alternate Data Stream (ADS)	489
12.7	Keylogger und Spyware	491
12.7.1	Grundlagen	492
12.7.2	Keylogger und Spyware in der Praxis	492
12.8	Advanced Persistent Threat (APT)	497
12.8.1	Wie funktioniert ein APT?	497
12.8.2	Ablauf eines APT-Angriffs	498
12.8.3	Zielgruppen von APT-Angriffen	498
12.9	Schutzmaßnahmen gegen Malware	499
12.10	Zusammenfassung und Prüfungstipps	499
12.10.1	Zusammenfassung und Weiterführendes	499
12.10.2	CEH-Prüfungstipps	500
12.10.3	Fragen zur CEH-Prüfungsvorbereitung	500
13	Malware-Erkennung und -Analyse	503
13.1	Grundlagen der Malware-Analyse	503
13.1.1	Statische Malware-Analyse	504
13.1.2	Dynamische Malware-Analyse	507
13.2	Verdächtiges Verhalten analysieren	507
13.2.1	Virencheck durchführen	508
13.2.2	Prozesse überprüfen	512
13.2.3	Netzwerkaktivitäten prüfen	515
13.2.4	Die Windows-Registrierung checken	520
13.2.5	Autostart-Einträge unter Kontrolle	524
13.2.6	Windows-Dienste checken	526
13.2.7	Treiber überprüfen	528
13.2.8	Integrität der Systemdateien prüfen	530
13.2.9	Datei-Integrität durch Prüfsummen-Check	531
13.2.10	System-Integrität mit Tripwire sichern	532
13.3	Sheep-Dip-Systeme	533
13.3.1	Einführung	533
13.3.2	Aufbau eines Sheep-Dip-Systems	534
13.4	Schutz durch Sandbox	535
13.4.1	Sandboxie	535
13.4.2	Cuckoo	537

13.5	Allgemeine Schutzmaßnahmen vor Malware-Infektion	538
13.6	Zusammenfassung und Prüfungstipps	539
13.6.1	Zusammenfassung und Weiterführendes	539
13.6.2	CEH-Prüfungstipps	540
13.6.3	Fragen zur CEH-Prüfungsvorbereitung	540
14	Steganografie	543
14.1	Grundlagen der Steganografie	543
14.1.1	Wozu Steganografie?	543
14.1.2	Ein paar einfache Beispiele	544
14.1.3	Klassifikation der Steganografie	545
14.2	Computergestützte Steganografie	549
14.2.1	Daten in Bildern verstecken	549
14.2.2	Daten in Dokumenten verstecken	554
14.2.3	Weitere Cover-Datenformate	555
14.3	Steganalyse und Schutz vor Steganografie	556
14.3.1	Methoden der Steganalyse	556
14.3.2	Steganalyse-Tools	557
14.3.3	Schutz vor Steganografie	557
14.4	Zusammenfassung und Prüfungstipps	558
14.4.1	Zusammenfassung und Weiterführendes	558
14.4.2	CEH-Prüfungstipps	559
14.4.3	Fragen zur CEH-Prüfungsvorbereitung	559
15	Spuren verwischen	561
15.1	Auditing und Logging	561
15.1.1	Die Windows-Protokollierung	562
15.1.2	Die klassische Linux-Protokollierung	564
15.2	Spuren verwischen auf einem Windows-System	567
15.2.1	Das Windows-Auditing deaktivieren	567
15.2.2	Windows-Ereignisprotokolle löschen	569
15.2.3	Most Recently Used (MRU) löschen	571
15.2.4	Zeitstempel manipulieren	573
15.2.5	Clearing-Tools	576
15.3	Spuren verwischen auf einem Linux-System	578
15.3.1	Logfiles manipulieren und löschen	578
15.3.2	Systemd-Logging in Journald.	580
15.3.3	Zeitstempel manipulieren	581
15.3.4	Die Befehlszeilen-Historie löschen	583
15.4	Schutz vor dem Spuren-Verwischen	584
15.5	Zusammenfassung und Prüfungstipps	585
15.5.1	Zusammenfassung und Weiterführendes	585
15.5.2	CEH-Prüfungstipps	586
15.5.3	Fragen zur CEH-Prüfungsvorbereitung	587

Teil IV	Netzwerk- und sonstige Angriffe	589
16	Network Sniffing mit Wireshark & Co.	593
16.1	Grundlagen von Netzwerk-Sniffern	593
16.1.1	Technik der Netzwerk-Sniffer	593
16.1.2	Wireshark und die Pcap-Bibliotheken	595
16.2	Wireshark installieren und starten	595
16.2.1	Installation unter Linux	595
16.2.2	Installation unter Windows	596
16.2.3	Der erste Start	597
16.3	Die ersten Schritte mit Wireshark	598
16.3.1	Grundeinstellungen	598
16.3.2	Ein erster Mitschnitt	600
16.4	Mitschnitt-Filter einsetzen	601
16.4.1	Analyse eines TCP-Handshakes	602
16.4.2	Der Ping in Wireshark	603
16.4.3	Weitere Mitschnittfilter	604
16.5	Anzeigefilter einsetzen	605
16.5.1	Eine HTTP-Sitzung im Detail	606
16.5.2	Weitere Anzeigefilter	608
16.6	Passwörter und andere Daten ausspähen	609
16.6.1	FTP-Zugangsdaten ermitteln	610
16.6.2	Telnet-Zugangsdaten identifizieren	611
16.6.3	SSH – sicherer Schutz gegen Mitlesen	613
16.6.4	Andere Daten ausspähen	615
16.7	Auswertungsfunktionen von Wireshark nutzen	616
16.8	Tcpdump und TShark einsetzen	618
16.8.1	Tcpdump – der Standard-Sniffer für die Konsole	618
16.8.2	TShark – Wireshark auf der Konsole	621
16.9	Zusammenfassung und Prüfungstipps	623
16.9.1	Zusammenfassung und Weiterführendes	623
16.9.2	CEH-Prüfungstipps	623
16.9.3	Fragen zur CEH-Prüfungsvorbereitung	624
17	Lauschangriffe & Man-in-the-Middle	627
17.1	Eavesdropping und Sniffing für Hacker	627
17.1.1	Eavesdropping und Wiretapping	628
17.1.2	Sniffing als Angriffsvektor	628
17.2	Man-in-the-Middle (MITM)	629
17.2.1	Was bedeutet Man-in-the-Middle?	630
17.2.2	Was erreichen wir durch einen MITM-Angriff?	631
17.3	Active Sniffing	631
17.3.1	Mirror-Ports: Ein Kabel mit drei Enden	632
17.3.2	Aus Switch mach Hub – MAC-Flooding	632
17.3.3	Auf dem Silbertablett: WLAN-Sniffing	634

17.3.4	Weitere physische Abhörmöglichkeiten	635
17.4	Die Kommunikation für MITM umleiten	635
17.4.1	Physische Umleitung	635
17.4.2	Umleitung über aktive Netzwerk-Komponenten	636
17.4.3	Umleiten mit ARP-Spoofing	637
17.4.4	ICMP-Typ 5 Redirect	637
17.4.5	DNS-Spoofing oder DNS-Cache-Poisoning	638
17.4.6	Manipulation der hosts-Datei	640
17.4.7	Umleiten via DHCP-Spoofing	641
17.5	Die Dsniff-Toolsammlung	642
17.5.1	Programme der Dsniff-Suite	642
17.5.2	Abhören des Netzwerk-Traffics	643
17.5.3	MITM mit arpspoof	644
17.5.4	Die ARP-Tabelle des Switches mit macof überfluten	647
17.5.5	DNS-Spoofing mit dnspooft	647
17.5.6	Dsniff	650
17.6	Man-in-the-Middle-Angriffe mit Ettercap	651
17.6.1	Einführung in Ettercap	651
17.6.2	DNS-Spoofing mit Ettercap	653
17.7	Schutz vor Lauschangriffen & MITM	661
17.8	Zusammenfassung und Prüfungstipps	663
17.8.1	Zusammenfassung und Weiterführendes	663
17.8.2	CEH-Prüfungstipps	664
17.8.3	Fragen zur CEH-Prüfungsvorbereitung	664
18	Session Hijacking	667
18.1	Grundlagen des Session Hijackings	667
18.1.1	Wie funktioniert Session Hijacking grundsätzlich?	668
18.1.2	Session-Hijacking-Varianten	668
18.2	Network Level Session Hijacking	669
18.2.1	Die TCP-Session im Detail	670
18.2.2	Entführen von TCP-Sessions	672
18.2.3	Weitere Hijacking-Varianten auf Netzwerk-Ebene	674
18.3	Application Level Session Hijacking	675
18.3.1	Die Session-IDs	676
18.3.2	Die Session-ID ermitteln	677
18.3.3	Sniffing/Man-in-the-Middle	677
18.3.4	Die Session-ID erraten – das Prinzip	678
18.3.5	WebGoat bereitstellen	678
18.3.6	Die Burp Suite – Grundlagen und Installation	681
18.3.7	Burp Suite als Intercepting Proxy	683
18.3.8	Der Burp Sequencer – Session-IDs analysieren	686
18.3.9	Entführen der Session mithilfe der Session-ID	690
18.3.10	Man-in-the-Browser-Angriff	696

18.3.11	Weitere Angriffsformen	698
18.4	Gegenmaßnahmen gegen Session Hijacking	700
18.4.1	Session Hijacking entdecken	700
18.4.2	Schutzmaßnahmen	701
18.5	Zusammenfassung und Prüfungstipps	703
18.5.1	Zusammenfassung und Weiterführendes	703
18.5.2	CEH-Prüfungstipps	704
18.5.3	Fragen zur CEH-Prüfungsvorbereitung	704
19	Firewalls, IDS/IPS und Honeypots einsetzen und umgehen	707
19.1	Firewall-Technologien	707
19.1.1	Netzwerk- und Personal-Firewalls	708
19.1.2	Filtertechniken und Kategorisierung der Netzwerk-Firewalls	709
19.2	Firewall-Szenarien	713
19.2.1	DMZ-Szenarien	713
19.2.2	Failover-Szenarien	715
19.3	Firewalls umgehen	716
19.3.1	Identifikation von Firewalls	716
19.3.2	IP-Adress-Spoofing	717
19.3.3	Was wirklich funktioniert	718
19.4	Intrusion-Detection- und -Prevention-Systeme	719
19.4.1	Grundlagen und Unterschiede zwischen IDS und IPS	719
19.4.2	Einführung in Snort	722
19.5	Intrusion-Detection-Systeme umgehen	726
19.5.1	Injection/Insertion	726
19.5.2	Evasion	727
19.5.3	Denial-of-Service-Angriff (DoS)	728
19.5.4	Obfuscation	728
19.5.5	Generieren von False Positives	728
19.5.6	Fragmentation	729
19.5.7	TCP Session Splicing	730
19.5.8	Weitere Evasion-Techniken	730
19.6	Network Access Control (NAC)	731
19.6.1	NAC-Lösungen - Grundlagen	731
19.6.2	Angriffsvektoren auf NAC-Lösungen	732
19.7	Honeypots	733
19.7.1	Grundlagen und Begriffsklärung	734
19.7.2	Kategorisierung der Honeypots	734
19.7.3	Valhala – ein Honeypot in der Praxis	737
19.7.4	Honeypots identifizieren und umgehen	740
19.7.5	Rechtliche Aspekte beim Einsatz von Honeypots	742
19.8	Zusammenfassung und Prüfungstipps	742
19.8.1	Zusammenfassung und Weiterführendes	742
19.8.2	CEH-Prüfungstipps	744
19.8.3	Fragen zur CEH-Prüfungsvorbereitung	744

20	Social Engineering	747
20.1	Einführung in das Social Engineering.	747
20.1.1	Welche Gefahren birgt Social Engineering?	748
20.1.2	Verlustangst, Neugier, Eitelkeit – die Schwachstellen des Systems Mensch	748
20.1.3	Varianten des Social Engineerings	751
20.1.4	Allgemeine Vorgehensweise beim Social Engineering	753
20.2	Human Based Social Engineering	754
20.2.1	Vortäuschen einer anderen Identität.	754
20.2.2	Shoulder Surfing & Co.	756
20.2.3	Piggybacking und Tailgaiting	757
20.3	Computer Based Social Engineering	758
20.3.1	Phishing	758
20.3.2	Pharming.	758
20.3.3	Spear Phishing	759
20.3.4	Drive-by-Downloads	760
20.3.5	Gefälschte Viren-Warnungen	761
20.4	Das Social-Engineer Toolkit (SET)	762
20.4.1	Einführung in SET	762
20.4.2	Praxisdemonstration: Credential Harvester	764
20.4.3	Weitere Angriffe mit SET.	767
20.5	So schützen Sie sich gegen Social-Engineering-Angriffe.	768
20.6	Zusammenfassung und Prüfungstipps.	770
20.6.1	Zusammenfassung und Weiterführendes	770
20.6.2	CEH-Prüfungstipps	771
20.6.3	Fragen zur CEH-Prüfungsvorbereitung	771
21	Hacking-Hardware	773
21.1	Allgemeines und rechtliche Hinweise zu Spionage-Hardware	774
21.2	Angriffsvektor USB-Schnittstelle	774
21.2.1	Hardware Keylogger	775
21.2.2	USB Rubber Ducky	776
21.2.3	Bash Bunny	779
21.2.4	Digispark	781
21.2.5	USBNinja	782
21.2.6	Mouse Jiggler	783
21.3	Weitere Hacking-Gadgets	783
21.3.1	VideoGhost	783
21.3.2	Packet Squirrel	784
21.3.3	LAN Turtle	785
21.3.4	Throwing Star LAN Tap	785
21.3.5	Software Defined Radio	786
21.3.6	Crazyradio PA	786
21.3.7	WiFi Pinapple	787

21.3.8	Proxmark 3	788
21.3.9	ChameleonMini	788
21.4	Raspberry Pi als Hacking-Kit.	788
21.5	Gegenmaßnahmen	790
21.6	Zusammenfassung und Prüfungstipps	792
21.6.1	Zusammenfassung und Weiterführendes	792
21.6.2	CEH-Prüfungstipps	793
21.6.3	Fragen zur CEH-Prüfungsvorbereitung.	793
22	DoS- und DDoS-Angriffe.	795
22.1	DoS- und DDoS-Grundlagen.	795
22.1.1	Was ist ein Denial-of-Service-Angriff?	796
22.1.2	Warum werden DoS- und DDoS-Angriffe durchgeführt?	796
22.1.3	Kategorien der DoS/DDoS-Angriffe.	797
22.2	DoS- und DDoS-Angriffstechniken	797
22.2.1	UDP-Flood-Angriff	798
22.2.2	ICMP-Flood-Angriff.	798
22.2.3	Smurf-Angriff	799
22.2.4	Syn-Flood-Angriff	800
22.2.5	Fragmentation-Angriff	803
22.2.6	Slowloris-Angriff	804
22.2.7	Permanenter Denial-of-Service (PDoS)	805
22.2.8	Distributed-Reflected-Denial-of-Service-Angriff (DRDoS)	806
22.3	Botnetze – Funktionsweise und Betrieb.	807
22.3.1	Bots und deren Einsatzmöglichkeiten	808
22.3.2	Aufbau eines Botnetzes.	808
22.3.3	Wie gelangen Bots auf die Opfer-Systeme?	810
22.3.4	Mobile Systeme und IoT.	811
22.3.5	Botnetze in der Praxis	811
22.3.6	Verteidigung gegen Botnetze und DDoS-Angriffe	812
22.4	DoS-Angriffe in der Praxis.	814
22.4.1	SYN- und ICMP-Flood-Angriff mit hping3	815
22.4.2	DoS-Angriff mit Metasploit.	817
22.4.3	DoS-Angriff mit SlowHTTPTest	819
22.4.4	Low Orbit Ion Cannon (LOIC)	821
22.5	Verteidigung gegen DoS- und DDoS-Angriffe	822
22.5.1	Allgemeiner Grundschutz.	822
22.5.2	Schutz vor volumetrischen DDoS-Angriffen.	823
22.6	Zusammenfassung und Prüfungstipps	824
22.6.1	Zusammenfassung und Weiterführendes	824
22.6.2	CEH-Prüfungstipps	825
22.6.3	Fragen zur CEH-Prüfungsvorbereitung.	825

Teil V	Web-Hacking	827
23	Web-Hacking – Grundlagen	831
23.1	Was ist Web-Hacking?	831
23.2	Architektur von Webanwendungen	832
23.2.1	Die Schichten-Architektur	832
23.2.2	Die URL-Codierung	833
23.2.3	Das Hypertext Transfer Protocol (HTTP)	834
23.2.4	Cookies	837
23.2.5	HTTP vs. HTTPS	837
23.2.6	Webservices und -technologien	838
23.3	Die gängigsten Webserver: Apache, IIS, nginx	843
23.3.1	Apache HTTP Server	843
23.3.2	Internet Information Services (IIS)	845
23.3.3	nginx	847
23.4	Typische Schwachstellen von Webservern und -anwendungen	848
23.4.1	Schwachstellen in Webserver-Plattformen	848
23.4.2	Schwachstellen in der Webanwendung	849
23.5	Reconnaissance für Web-Hacking-Angriffe	850
23.5.1	Footprinting und Scanning	850
23.5.2	Web-Firewalls und Proxys entlarven	852
23.5.3	Hidden Content Discovery	852
23.5.4	Website-Mirroring	855
23.5.5	Security-Scanner	855
23.6	Praxis-Szenario: Einen Apache-Webserver mit Shellshock hacken	858
23.6.1	Die Laborumgebung präparieren	858
23.6.2	Den Angriff durchführen	860
23.7	Praxis-Szenario 2: Angriff auf WordPress	861
23.7.1	WordPress-VM bereitstellen	862
23.7.2	WordPress scannen und Enumeration	866
23.7.3	User-Hacking	868
23.8	Zusammenfassung und Prüfungstipps	868
23.8.1	Zusammenfassung und Weiterführendes	868
23.8.2	CEH-Prüfungstipps	869
23.8.3	Fragen zur CEH-Prüfungsvorbereitung	869
24	Web-Hacking – OWASP Top 10	871
24.1	Einführung in die OWASP-Projekte	871
24.1.1	OWASP Juice Shop	872
24.1.2	OWASP ModSecurity Core Rule Set (CRS)	873
24.1.3	OWASP Web Security Testing Guide	873
24.1.4	OWASP Top 10	873
24.2	WebGoat & Co – virtuelle Sandsäcke für das Web-Hacking-Training	874
24.2.1	WebGoat	875
24.2.2	Mutillidae II	875

24.2.3	bWAPP	876
24.2.4	DVWA	877
24.2.5	Web Security Dojo	878
24.2.6	Vulnhub und Pentesterlab	879
24.3	Die OWASP Top 10 in der Übersicht	879
24.4	A01 – Broken Access Control	880
24.4.1	Unsichere direkte Objektreferenzen	880
24.4.2	Fehlerhafte Autorisierung auf Anwendungsebene	882
24.4.3	Schutzmaßnahmen	885
24.5	A02 – Cryptographic Failures	886
24.5.1	Welche Daten sind betroffen?	886
24.5.2	Angriffsszenarien	887
24.5.3	Schutzmaßnahmen	888
24.6	A03 – Injection	889
24.6.1	Kategorien von Injection-Angriffen	889
24.6.2	Beispiel für einen Injection-Angriff	889
24.6.3	Cross-Site-Scripting (XSS)	892
24.6.4	Wie funktioniert XSS?	892
24.6.5	Ein einfaches XSS-Beispiel	893
24.6.6	XSS-Varianten	895
24.6.7	Ein Beispiel für Stored XSS	897
24.6.8	Exkurs: Cross-Site-Request-Forgery (CSRF)	898
24.6.9	Schutzmaßnahmen gegen XSS-Angriffe	900
24.7	A04 – Insecure Design	901
24.7.1	Was bedeutet unsicheres Design?	901
24.7.2	Sichere Webentwicklung	902
24.7.3	Schutzmaßnahmen	902
24.8	A05 – Security Misconfiguration	903
24.8.1	Typische Fehlkonfigurationen	903
24.8.2	Directory Browsing	903
24.8.3	Allgemeine Schutzmaßnahmen	905
24.8.4	A4 – XML External Entities (XXE)	906
24.8.5	XML-Entities	906
24.8.6	Ein Beispiel für einen XXE-Angriff	907
24.8.7	Schutzmaßnahmen	908
24.9	A06 – Vulnerable and Outdated Components	909
24.9.1	Worin liegt die Gefahr und wer ist gefährdet?	909
24.9.2	Verwundbare JavaScript-Bibliotheken aufdecken mit Retire.js	909
24.9.3	Schutzmaßnahmen	910
24.10	A07 – Identification and Authentication Failures	911
24.10.1	Grundlagen	911
24.10.2	Identitätsdiebstahl durch Token-Manipulation	911
24.10.3	Schutzmaßnahmen	914
24.11	A08 – Software and Data Integrity Failures	914

24.11.1	Was bedeutet Integritätsverletzung?	915
24.11.2	Unsichere Deserialisierung	915
24.11.3	Was bedeutet Serialisierung von Daten?	915
24.11.4	Wie wird die Deserialisierung zum Problem?	916
24.11.5	Schutzmaßnahmen	916
24.12	A09 – Security Logging and Monitoring Failures.	917
24.12.1	Herausforderungen beim Logging & Monitoring	917
24.12.2	Sind unserer Systeme gefährdet?	918
24.13	A10 – Server-Side Request Forgery (SSRF)	919
24.13.1	Wie funktioniert SSRF?	919
24.13.2	Ein SSRF-Beispiel.	920
24.14	Zusammenfassung und Prüfungstipps.	923
24.14.1	Zusammenfassung und Weiterführendes	923
24.14.2	CEH-Prüfungstipps	923
24.14.3	Fragen zur CEH-Prüfungsvorbereitung	924
25	SQL-Injection.	925
25.1	Mit SQL-Injection das Login austricksen	926
25.1.1	Der grundlegende Ansatz.	926
25.1.2	Anmeldung als gewünschter Benutzer	930
25.1.3	Clientseitige Sicherheit.	930
25.2	Daten auslesen mit SQL-Injection	932
25.2.1	Manipulation eines GET-Requests	933
25.2.2	Informationen über die Datenbank auslesen	934
25.2.3	Die Datenbank-Tabellen identifizieren	936
25.2.4	Spalten und Passwörter auslesen	938
25.3	Fortgeschrittene SQL-Injection-Techniken	939
25.3.1	Einführung in Blind SQL-Injection.	940
25.3.2	Codieren des Injection-Strings	942
25.3.3	Blind SQLi: Eins oder null?	945
25.3.4	Time based SQL-Injection	946
25.4	SQLMap – automatische Schwachstellensuche	948
25.4.1	SQLi-CheatSheets	948
25.4.2	Einführung in SQLMap	949
25.4.3	Weitere Analysen mit SQLMap	954
25.5	Schutzmaßnahmen vor SQLi-Angriffen	956
25.6	Zusammenfassung und Prüfungstipps.	957
25.6.1	Zusammenfassung und Weiterführendes	957
25.6.2	CEH-Prüfungstipps	957
25.6.3	Fragen zur CEH-Prüfungsvorbereitung	958
26	Web-Hacking – sonstige Injection-Angriffe	961
26.1	Command-Injection	961
26.1.1	Einführung in Command-Injection-Angriffe	962
26.1.2	Command-Injection in der Praxis	962

26.1.3	Schutzmaßnahmen vor Command-Injection-Angriffen	964
26.2	File-Injection	965
26.2.1	Directory-Traversal-Angriffe	965
26.2.2	File-Upload-Angriffe	967
26.2.3	Local File Inclusion versus Remote File Inclusion	970
26.3	Zusammenfassung und Prüfungstipps	973
26.3.1	Zusammenfassung und Weiterführendes	973
26.3.2	CEH-Prüfungstipps	973
26.3.3	Fragen zur CEH-Prüfungsvorbereitung	974
27	Buffer-Overflow-Angriffe	977
27.1	Wie funktioniert ein Buffer-Overflow-Angriff?	978
27.1.1	Das Grundprinzip	978
27.1.2	Welche Anwendungen sind verwundbar?	978
27.1.3	Funktionsweise des Stacks	979
27.1.4	Register	980
27.2	Ein Buffer-Overflow-Angriff in der Praxis	981
27.2.1	SLmail-Exploit	981
27.2.2	Die Laborumgebung	981
27.2.3	Der Immunity Debugger	984
27.2.4	Fuzzing	986
27.2.5	Einen eindeutigen String erstellen	990
27.2.6	Den EIP lokalisieren	992
27.2.7	Den Shellcode platzieren	992
27.2.8	Bad Characters identifizieren	994
27.2.9	Grundüberlegung: Wohin soll der EIP zeigen?	996
27.2.10	Mona und die Module	996
27.2.11	Die Anweisung JMP ESP auffinden	997
27.2.12	Den Programmablauf über den EIP steuern	999
27.2.13	Den Shellcode erstellen und ausführen	1001
27.3	Heap-Overflow-Angriffe	1005
27.3.1	Der Heap	1005
27.3.2	Heap Overflow versus Stack Overflow	1006
27.3.3	Use-after-free	1006
27.3.4	Heap Spraying	1006
27.4	Schutzmaßnahmen gegen Buffer-Overflow-Angriffe	1007
27.4.1	Address Space Layout Randomization (ASLR)	1007
27.4.2	Data Execution Prevention (DEP)	1008
27.4.3	SEHOP und SafeSEH	1008
27.4.4	Stack Canary	1008
27.4.5	Wie sicher sind die Schutzmaßnahmen?	1009
27.5	Zusammenfassung und Prüfungstipps	1010
27.5.1	Zusammenfassung und Weiterführendes	1010
27.5.2	CEH-Prüfungstipps	1011
27.5.3	Fragen zur CEH-Prüfungsvorbereitung	1011

Teil VI	Angriffe auf WLAN und Next-Gen-Technologien	1013
28	WLAN-Hacking	1017
28.1	WLAN-Grundlagen	1017
28.1.1	Frequenzen und Kanäle	1018
28.1.2	Der IEEE-802.11-Standard	1019
28.1.3	Infrastruktur	1020
28.1.4	Verbindungsaufbau	1023
28.1.5	Verschlüsselungsmethoden	1026
28.2	Setup für das WLAN-Hacking	1029
28.2.1	Die WLAN-Hacking-Plattform	1029
28.2.2	Der richtige WLAN-Adapter	1030
28.2.3	Den Monitor Mode aktivieren	1031
28.3	WLAN-Scanning und -Sniffing	1032
28.3.1	Scanning	1033
28.3.2	WLAN-Sniffing	1033
28.3.3	Hidden SSIDs aufspüren	1035
28.4	Angriffe auf WLAN	1037
28.4.1	Denial of Service durch Störsender	1037
28.4.2	Deauthentication-Angriff	1037
28.4.3	Angriff auf WEP	1039
28.4.4	Angriff auf WPA/WPA2	1043
28.4.5	Angriff auf WPA3	1045
28.4.6	Angriff auf WPS	1046
28.4.7	MAC-Filter umgehen	1049
28.4.8	WLAN-Passwörter auslesen	1052
28.4.9	Standard-Passwörter	1054
28.4.10	Captive Portals umgehen	1055
28.5	Rogue Access Points	1057
28.5.1	Fake-Access-Point bereitstellen	1057
28.5.2	WLAN-Phishing	1060
28.6	Schutzmaßnahmen	1062
28.6.1	Allgemeine Maßnahmen	1062
28.6.2	Fortgeschrittene Sicherheitsmechanismen	1063
28.7	Zusammenfassung und Prüfungstipps	1064
28.7.1	Zusammenfassung und Weiterführendes	1064
28.7.2	CEH-Prüfungstipps	1065
28.7.3	Fragen zur CEH-Prüfungsvorbereitung	1065
29	Mobile Hacking	1067
29.1	Grundlagen	1067
29.1.1	Mobile Betriebssysteme	1067
29.1.2	Apps und App-Stores	1069
29.2	Angriffe auf mobile Geräte	1071
29.2.1	Schutzziele	1071

29.2.2	Angriffsvektoren	1072
29.2.3	OWASP Mobile Top 10	1074
29.3	Mobile Hacking in der Praxis	1075
29.3.1	Android über den PC	1075
29.3.2	Android-Rooting	1079
29.3.3	Jailbreaking iOS	1084
29.3.4	SIM-Unlock	1085
29.3.5	Hacking-Tools für Android	1086
29.3.6	Android-Tojaner erstellen	1088
29.3.7	Angriffe auf iOS	1093
29.3.8	Spyware für mobile Geräte	1094
29.4	Bring Your Own Device (BYOD)	1095
29.4.1	BYOD-Vorteile	1095
29.4.2	BYOD-Risiken	1095
29.4.3	BYOD-Sicherheit	1096
29.5	Mobile Device Management (MDM)	1097
29.6	Schutzmaßnahmen	1098
29.7	Zusammenfassung und Prüfungstipps	1100
29.7.1	Zusammenfassung und Weiterführendes	1100
29.7.2	CEH-Prüfungstipps	1101
29.7.3	Fragen zur CEH-Prüfungsvorbereitung	1102
30	IoT- und OT-Hacking und -Security	1105
30.1	Das Internet of Things	1105
30.1.1	Was ist das Internet of Things?	1106
30.1.2	Was umfasst das Internet of Things?	1106
30.1.3	Die grundlegende Sicherheitsproblematik von IoT-Geräten	1107
30.2	IoT-Technik – Konzepte und Protokolle	1107
30.2.1	IoT-Betriebssysteme	1108
30.2.2	IoT-Kommunikationsmodelle	1108
30.2.3	IoT-Übertragungstechnologien	1110
30.2.4	IoT-Kommunikationsprotokolle	1112
30.3	Schwachstellen von IoT-Systemen	1113
30.3.1	OWASP Top 10 IoT 2018	1113
30.3.2	Angriffsvektoren auf IoT-Systeme	1116
30.4	IoT-Angriffszenarien	1118
30.4.1	Rolling-Code-Angriff	1118
30.4.2	Mirai – Botnet und DDoS-Angriffe	1120
30.4.3	Lokale Angriffe über die UART-Schnittstelle	1121
30.4.4	Command-Injection via Web-Frontend	1122
30.4.5	Der BlueBorne-Angriff	1123
30.4.6	Angriffe auf ZigBee-Geräte mit Killerbee	1124
30.4.7	Angriffe auf Firmware	1125
30.5	Weitere Angriffsformen auf IoT-Ökosysteme	1126
30.5.1	Exploit Kits	1126

	30.5.2	IoT-Suchmaschinen	1126
30.6		OT-Hacking	1128
	30.6.1	OT-Grundlagen und -Konzepte	1128
	30.6.2	Konvergenz von IT und OT	1129
	30.6.3	Das Purdue-Modell	1130
	30.6.4	OT-Sicherheitsherausforderungen	1131
	30.6.5	OT-Schwachstellen und Bedrohungen	1132
	30.6.6	OT-Malware	1133
	30.6.7	OT-Hackingtools und -Enumeration	1134
	30.6.8	Schutzmaßnahmen vor OT-Angriffen	1135
30.7		Schutzmaßnahmen vor IoT-Angriffen	1136
30.8		Zusammenfassung und Prüfungstipps	1138
	30.8.1	Zusammenfassung und Weiterführendes	1138
	30.8.2	CEH-Prüfungstipps	1138
	30.8.3	Fragen zur CEH-Prüfungsvorbereitung	1138
31		Angriffe auf die Cloud	1141
31.1		Grundlagen des Cloud Computings	1141
	31.1.1	Was ist eigentlich »die Cloud?«	1142
	31.1.2	Cloud-Service-Modelle	1143
	31.1.3	Deployment-Modelle für die Cloud	1144
	31.1.4	Besondere Computing-Varianten	1146
	31.1.5	Große Cloud-Anbieter	1147
31.2		Wichtige Cloud-Technologien	1148
	31.2.1	Virtualisierung	1148
	31.2.2	Container-Technologien	1149
	31.2.3	Docker	1152
	31.2.4	Kubernetes	1154
	31.2.5	Schwachstellen von Container-Technologien	1155
	31.2.6	Serverless Computing	1156
	31.2.7	Schwachstellen von Serverless Computing	1157
	31.2.8	Weitere Cloud-Dienstleistungen	1158
31.3		Bedrohungen der Sicherheit und Integrität in der Cloud	1158
	31.3.1	Kontrollverlust	1158
	31.3.2	Unsichere Cloud-Infrastruktur	1159
	31.3.3	Missbrauchs-Risiken beim Cloud-Anbieter	1160
	31.3.4	Unsichere Kommunikation mit der Cloud	1161
	31.3.5	Unzureichende Zugangskontrolle	1163
	31.3.6	Cloud Computing für Hacker	1163
	31.3.7	Übersicht und Zusammenfassung	1164
31.4		Angriffe auf Cloud-Infrastrukturen	1164
	31.4.1	Zugangsdaten ermitteln	1164
	31.4.2	Persistenter Zugang sichern	1165
	31.4.3	Malware einschleusen	1166
	31.4.4	Unsichere Voreinstellungen ausnutzen	1166

31.4.5	Cryptojacking	1167
31.4.6	Zugang über Federation Services	1167
31.4.7	Angriffsvektor Webanwendung	1168
31.5	Cloud-Security-Tools	1169
31.5.1	Security-Tools des Cloud-Anbieters	1169
31.5.2	Drittanbieter-Security-Software	1169
31.5.3	Pentest-Simulation mit CloudGoat und Pacu	1170
31.6	Zusammenfassung und Prüfungstipps	1171
31.6.1	Zusammenfassung und Weiterführendes	1171
31.6.2	CEH-Prüfungstipps	1172
31.6.3	Fragen zur CEH-Prüfungsvorbereitung.	1173
32	Durchführen von Penetrationstests	1175
32.1	Begriffsbestimmung Penetrationstest	1175
32.1.1	Was bedeutet »Penetrationstest« eigentlich?	1176
32.1.2	Wozu einen Penetrationstest durchführen?	1176
32.1.3	Penetrationstest vs. Security Audit vs. Vulnerability Assessment	1177
32.1.4	Arten des Penetrationstests.	1178
32.2	Rechtliche Bestimmungen	1179
32.2.1	In Deutschland geltendes Recht.	1180
32.2.2	US-amerikanisches und internationales Recht	1181
32.3	Vorbereitung und praktische Durchführung des Penetrationstests.	1183
32.3.1	Die Beauftragung.	1183
32.3.2	Methodik der Durchführung	1185
32.3.3	Praxistipps	1188
32.4	Der Pentest-Report.	1191
32.4.1	Dokumentation während des Pentests.	1191
32.4.2	Was umfasst der Pentest-Report?	1192
32.4.3	Aufbau des Pentest-Reports	1193
32.5	Abschluss und Weiterführendes.	1195
32.5.1	Das Abschluss-Meeting.	1196
32.5.2	Weiterführende Tätigkeiten	1196
32.6	Zusammenfassung und Prüfungstipps	1196
32.6.1	Zusammenfassung und Weiterführendes	1196
32.6.2	CEH-Prüfungstipps	1197
32.6.3	Fragen zur CEH-Prüfungsvorbereitung.	1198
A	Lösungen.	1201
	Stichwortverzeichnis	1215

Einleitung

Sie suchen nach einem strukturierten, umfassenden Praxishandbuch zum Thema »Ethical Hacking und Penetration Testing«? Prima, dann sind Sie hier genau richtig! In diesem Buch lernen Sie die Vorgehensweisen und Techniken professioneller Hacker und Penetration-Tester kennen und erlernen das Handwerk von der Pike auf. Durch viele Schritt-für-Schritt-Anleitungen, die Sie selbst in Ihrem Hacking-Labor nachvollziehen können, erleben Sie die Hacking-Techniken quasi live und in der Praxis. Hier ist Mitmachen angesagt!

Dieses Buch versteht sich zum einen als Praxisleitfaden für einen fundierten Einstieg in die Welt der Hacker und Penetration-Tester. Zum anderen sind die Inhalte an das Curriculum des Certified-Ethical-Hacker-Examens (CEHv12) des EC-Council angelehnt, sodass Sie dieses Werk als zusätzliche Ressource für die Prüfungsvorbereitung nutzen können. Bitte beachten Sie hierzu, dass es bestimmte Voraussetzungen für die Prüfungszulassung gibt, die wir Ihnen im ersten Kapitel erläutern.

Das CEH-Examen unterliegt ständigen Aktualisierungen, die naturgemäß nicht im bereits gedruckten Buch berücksichtigt werden können. Im Buch-Memberbereich auf www.hacking-akademie.de/buch/member versuchen wir aber, immer zeitnah aktualisierte Informationen bereitzustellen. Die Zugangsdaten zum Memberbereich finden Sie am Ende dieser Einleitung.

Für wen ist dieses Buch geeignet?

Dieses Buch ist für Sie geeignet, wenn Sie sich praxisorientiert und umfassend mit den Themen Hacking und Penetration Testing beschäftigen möchten. Die Zielgruppe umfasst insbesondere:

- Angehende Ethical Hacker und Penetration-Tester
- System- und Netzwerkadministratoren mit Fokus auf IT-Sicherheit
- Verantwortliche im Bereich IT-Security
- Interessierte Power-User

Auch wenn Sie sich durch einfaches Durchlesen des Buches bereits einen guten Überblick über das Thema verschaffen können, ist der Inhalt eher dazu konzipiert, tief in die Materie einzutauchen, und fordert Sie mit konkreten praktischen Beispielen zum Mitmachen auf. Dies erfordert bei Ihnen auf diesem Level auch ein ordentliches Maß an Engagement und Eigeninitiative. Aber genau so lernen Sie die Methoden nicht nur in der Theorie, sondern direkt in der praktischen Umsetzung.

Die Inhalte bauen an einigen Stellen aufeinander auf, sodass das Buch für ein umfassendes Verständnis Kapitel für Kapitel durchgearbeitet werden sollte. Natürlich eignet es sich darüber hinaus auch als Nachschlagewerk, da zu allen Inhalten, die für das Verständnis eines Themas benötigt werden, entsprechende Verweise zu den jeweiligen Stellen im Buch vorhanden sind.

Für wen ist dieses Buch nicht geeignet?

Auch wenn Sie in diesem Buch sehr viele Hacking-Tools kennenlernen werden, so möchten wir an dieser Stelle doch klar betonen, dass das Buch nicht für Scriptkiddies gedacht ist, die mit ein paar wenigen Klicks coole Hacks zaubern und ihre Freunde beeindrucken wollen. Leser, die ohne viel Hintergrundwissen und Engagement ein paar oberflächliche Tricks lernen wollen, finden sicher andere Literatur interessanter.

Andersherum geht es hier auch nicht darum, versierten Profis, die bereits tief in den Themen stecken, den letzten Schliff zu geben. Zu jedem Thema, das das Buch aufgreift, lassen sich eigene Bücher schreiben. Auch wenn die Seitenzahl sehr groß ist, können wir zu vielen Themen nicht mehr als einen fundierten, praxisnahen Einstieg bieten.

Was werden Sie hier lernen?

In diesem Buch geht es um Ethical Hacking und Penetration Testing. Wir werden diese Begriffe noch detaillierter beschreiben. Vom Grundsatz handelt es sich darum, die Perspektive des Angreifers einzunehmen, um die Schwachstellen von Computersystemen und -netzwerken aufzudecken. Dabei haben wir unter dem Strich das Ziel, die IT-Systeme sicherer zu machen. Es geht also nicht darum, die gefundenen Schwachstellen für die eigene Bereicherung zu nutzen, sondern darum, dem Auftraggeber die Möglichkeit zu geben, diese zu beseitigen. Anders ausgedrückt, bilden wir Sie hier zu einem »gutartigen« Hacker aus. Die Vorgehensweise, Technologien und eingesetzten Tools sind jedoch weitgehend dieselben, wie sie von bössartigen Hackern verwendet werden. Diese lernen Sie damit also ebenfalls kennen. Es ist wie so oft: Nicht die Werkzeuge bestimmen darüber, ob sie etwas verbessern oder Schaden anrichten, sondern derjenige, der sich diese Werkzeuge zunutze macht und einsetzt.

Hacking ist einerseits sehr kreativ und individuell, andererseits gibt es aber auch eine sinnvolle Vorgehensweise mit verschiedenen Phasen, die in fast jedem professionellen Hacking-Angriff enthalten sind. Sie erfahren, welche das sind und wie die einzelnen Phasen ablaufen. Viele Hacking-Tätigkeiten bauen aufeinander auf, andere kommen nur in bestimmten Szenarien zum Tragen. Wir haben in diesem Buch fast alle relevanten und gängigen Bereiche abgedeckt: angefangen vom simplen Passwort-Hacking über diverse Web-Hacking-Szenarien bis hin zu Mobile- und IoT-Hacking. Für alle Angriffsformen werden effektive Verteidigungsmaßnahmen aufgelistet, so dass Sie Ihre Kunden dabei unterstützen können, die gefundenen Schwachstellen zu beheben.

Der Fokus in diesem Buch liegt allerdings auf den Angriffstechniken. Sie erhalten zum einen fundierte Hintergrundinformationen zur Vorgehensweise und zu den Hacking-Techniken und zum anderen viele Praxissszenarien, in denen Sie Ihr neues Wissen praktisch einsetzen können. Nachdem Sie dieses Buch durchgearbeitet und die Szenarien praktisch nachvollzogen haben, sind Sie auf dem besten Weg zu einem fähigen Ethical Hacker und Penetration-Tester. Im Anschluss sind Sie in der Lage, Ihre Fähigkeiten eigenständig weiterzuentwickeln und mit zusätzlichen Informationsquellen Ihr Know-how zu vertiefen. Zudem erhalten Sie eine wertvolle Ressource für die Vorbereitung auf das CEHv12-Examen, mit dem Sie Ihre Karriere als Ethical Hacker effektiv voranbringen können.

Inhaltsübersicht

Das Buch ist in sechs Teile untergliedert. Nachfolgend stellen wir Ihnen den Inhalt kurz vor, damit Sie sich ein Bild verschaffen können.

Teil I – Grundlagen und Arbeitsumgebung

Hier erfahren Sie zunächst in Kapitel 1, welche Hacker-Typen es gibt und welche Ziele diese verfolgen. Wichtig ist dabei auch der rechtliche Aspekt, den wir natürlich ebenfalls betrachten. In Kapitel 2 bauen wir gemeinsam die Arbeitsumgebung für unser Hacking-Labor auf, das Sie im Laufe des gesamten Buches nutzen können. In Kapitel 3 lernen Sie Ihr wichtigstes Arbeitsgerät namens Kali Linux kennen.

Kapitel 4 widmet sich der Anonymität im Internet und der Methoden, deren sich die Hacker bedienen, um anonym zu bleiben. In Kapitel 5 betrachten wir mit der Kryptografie eines der wichtigsten Konzepte im Rahmen der IT-Sicherheit, wobei kryptografische Systeme in der Praxis auch immer wieder Angriffen ausgesetzt sind.

Teil II – Informationsbeschaffung

Im zweiten Teil beschäftigen wir uns mit der Informationsbeschaffung. Zunächst lernen Sie in Kapitel 6 die passive Datensammlung. In Kapitel 7 nehmen wir das Netzwerk unter die Lupe mithilfe von Netzwerk-Scannern wie z.B. Nmap. Kapitel 8 enthält Techniken und Wege für den Enumeration-Prozess, bei dem wir versuchen, aus verschiedenen Netzwerk-Diensten so viele Informationen zu extrahieren wie möglich.

Mit dem Vulnerability-Scanning in Kapitel 9 werden wir dann bereits aggressiver und suchen gezielt nach Schwachstellen. Die Schwachstellenanalyse behandeln wir ebenfalls in diesem Kapitel.

Teil III – Systeme angreifen

Nun geht es daran, Systeme konkret zu hacken. Wir beginnen in Kapitel 10 mit dem klassischen Password-Hacking und betrachten diverse Wege, um an Login-Daten zu gelangen. Mit der Privilegien-Eskalation in Kapitel 11 zielen wir darauf ab, unserer Rechte zu erweitern, wenn wir einen nicht-privilegierten Zugang zu den Zielsystemen erlangt haben.

Die Kapitel 12 und 13 beschäftigen sich mit Malware. Zum einen lernen Sie, wie Malware Computersysteme angreift, und erfahren dabei auch, wie Sie selbst Trojaner und ähnliche bösartige Software erstellen können. Zum anderen betrachten wir die Malware-Analyse, also Wege, um Malware aufzuspüren und zu beseitigen.

In Kapitel 14 erfahren Sie, wie Sie mithilfe von Steganografie Dateien und Informationen unentdeckt und versteckt transportieren können. Kapitel 15 befasst sich mit dem Verwischen von Spuren. Dies ist ein elementarer Bestandteil eines Hacking-Prozesses, wenn der Angreifer unentdeckt bleiben möchte.

Teil IV – Netzwerk- und sonstige Angriffe

Der Übergang zu diesem Teil ist fließend. In Kapitel 16 schauen wir mit Wireshark & Co. hinter die Kulissen der Netzwerk-Kommunikation. Hier lernen Sie, wie Sie Passwörter und Login-Vorgänge mitschneiden und ganze Sessions analysieren können. Dies führt wie von selbst zu Kapitel 17, in dem es um Lauschangriffe und Man-in-the-Middle-Angriffe geht.

Mit Session-Hijacking kann ein Angreifer eine etablierte und authentifizierte Session von ahnungslosen Benutzern übernehmen und spart sich so die Eingabe von Zugangsdaten. Wie das geht, erfahren Sie in Kapitel 18.

In Kapitel 19 lernen Sie die wichtigsten Security-Systeme kennen, denen sich ein Angreifer gegenüber sieht. Hierzu gehören neben Firewalls insbesondere Intrusion-Detection- bzw. -Prevention-Systeme sowie Honeybots.

Den Abschluss dieses vierten Teils bilden drei eher anders geartete Angriffsmethoden. In Kapitel 20 werfen wir einen Blick hinter die Kulissen des Social Engineerings. Mit dieser Technik greifen wir nicht die Computersysteme selbst an, sondern bedienen uns psychologischer Tricks, um die Benutzer der Systeme auszutricksen und an Informationen zu gelangen. Kapitel 21 präsentiert Ihnen gängige Hacking-Hardware. Hier lernen Sie zum Beispiel, wie Sie einen Keylogger installieren oder ein Hacking-Kit für die Hosentasche auf einem Raspberry Pi einrichten können. Last, but not least beschäftigen wir uns in Kapitel 22 mit DoS- und DDoS-Angriffen. Diese destruktive Angriffsform ist im Internet weit verbreitet und kann auch im Rahmen von größer angelegten Angriffen nützlich sein, um bestimmte Systeme außer Gefecht zu setzen, die den Angriff evtl. verhindern könnten.

Teil V – Web-Hacking

Einer der wichtigsten Angriffsvektoren ist der Angriff auf Webanwendungen. Daher haben wir diesem Thema einen breiten Raum eingeräumt. In Kapitel 23 lernen Sie zunächst die Grundlagen der Web-Kommunikation und -Technologien und erfahren, wie Angriffe auf Webserver und -anwendungen grundsätzlich funktionieren.

Kapitel 24 führt Sie in die Welt der OWASP Top 10 ein, OWASP steht für *Open Web Application Security Project*. Dabei handelt es sich um die zehn gängigsten Angriffsvektoren auf Webanwendungen. In diesem Kapitel erfahren Sie die daraus resultierenden Angriffe in Theorie und Praxis. Kapitel 25 greift den wichtigsten Punkt der OWASP Top 10 heraus und betrachtet den Angriffsvektor SQL-Injection von allen Seiten. In Kapitel 26 ergänzen Sie Ihr Wissen zu Injection-Angriffen und wir betrachten weitere Formen wie Command-Injection, Code-Injection oder LFI und RFI.

Den Abschluss dieses Teils bildet eine sehr gängige Form des Angriffs auf Software, die zwar häufig bei Webanwendungen zum Einsatz kommt, aber nicht auf diese beschränkt ist. Die Rede ist von Buffer-Overflow-Angriffen, die Sie in Kapitel 27 kennenlernen. Dort gehen wir ein umfassendes Praxisbeispiel durch, sodass Sie Ihren eigenen Buffer-Overflow-Angriff durchführen können.

Teil VI – Angriffe auf WLAN und Next-Gen-Technologien

Nun kommen wir zum letzten Teil des Buches, in dem wir uns zunächst mit der Thematik der mobilen Geräte beschäftigen. Im Kapitel 28 lernen Sie alles rund um WLAN-Hacking. Welchen Angriffsvektoren Smartphones und Tablets ausgesetzt sind, erfahren Sie in Kapitel 29. Kapitel 30 führt Sie in die Welt des IoT-Hackings ein, das immer wichtiger wird, da das Internet of Things seinen Siegeszug unaufhaltsam fortsetzt und die internetfähigen Alltagsgegenstände oft angreifbar sind. Mit dem Thema Cloud-Security schließen wir das Themenspektrum dieses Buches in Kapitel 31 ab.

An dieser Stelle haben Sie ein fundiertes Verständnis für Hacking-Methoden und -Technologien sowie für gängige Hacking-Tools. Zudem haben Sie zu allen Angriffsmethoden und -vektoren die effektivsten Gegenmaßnahmen kennengelernt und sind in der Lage, Kunden bzw. Auftraggeber hinsichtlich der Absicherung ihrer Systeme fundiert zu beraten.

Um dieser Tätigkeit einen Rahmen zu geben, existieren Penetrationstests. Das letzte Kapitel dieses Buches erläutert detailliert die Vorgehensweise bei einem Penetrationstest und gibt viele Tipps und Hinweise für angehende Penetration-Tester.

Aktualität der Inhalte

Als wir dieses Buch vor über sechs Jahren begonnen hatten, war uns nicht einmal im Ansatz klar, auf was wir uns einlassen würden! Es sollte unser bisher umfangreichstes Buchprojekt werden, da der Inhalt ständigen Änderungen und Anpassungen unterworfen ist. Als wir das Buch inhaltlich einmal fertiggestellt hatten, konnten wir sozusagen von vorn anfangen und mussten viele Stellen überarbeiten, vieles ergänzen und einiges streichen, da es keine Gültigkeit mehr hatte. Fast die Hälfte des Buches wurde in der Zwischenzeit inhaltlich überarbeitet, um es an den aktuellen Stand anzupassen.

Mittlerweile wurde das Buch für die 3. Auflage erneut an vielen Stellen überarbeitet, um es unter anderem für die aktuelle Zertifizierung zum CEHv12 zu aktualisieren. Und auch hier mussten wir an diversen Stellen veraltete Tools und Beschreibungen anpassen.

Aufgrund dieser Erfahrung haben wir einen wichtigen Hinweis an Sie als Leser: Wir haben viel Herzblut in dieses Buch investiert. Alle Anleitungen wurden mit größtmöglicher Sorgfalt erstellt und mehrfach getestet. Leider können die Anleitungen jedoch immer nur den Stand zum Zeitpunkt der Erstellung darstellen. Programme, Webseiten und Prozesse unterliegen in der IT-Welt ständiger Weiterentwicklung und Veränderung. Daher kann und wird es passieren, dass vereinzelt Programme nicht mehr so funktionieren wie beschrieben, Webseiten anders aussehen als im Buch abgedruckt und Inhalte unter Umständen nicht mehr in der Form zur Verfügung stehen wie beschrieben. Wir bitten hierfür um Verständnis und motivieren Sie, in derartigen Fällen selbstständig nach Lösungen zu suchen.

Denn das ist Hacking: neue Wege gehen, Dinge anders machen, um zu neuen Ergebnissen zu gelangen. Hacking erfordert Kreativität, Neugier und eine gute Portion Beharrlichkeit, da Hacker die Computersysteme und Software nicht in der vom Hersteller oder Entwickler erwarteten Art und Weise nutzen und daher mit dem Unerwarteten umgehen müssen.

Die Webseite zum Buch

Obwohl dieses Buch bereits sehr umfangreich ist, mussten wir aus Platzgründen diverse Inhalte auslagern. An vielen Stellen im Buch verweisen wir auf die jeweiligen Dokumente mit ergänzenden Informationen, die unter www.hacking-akademie.de/buch/member verfügbar sind. Sie stehen exklusiv für Sie als Leser zur Verfügung und sind Zugangsgeschützt. Geben Sie das Passwort **h4ckm3mber** ein, um in den Buch-Memberbereich zu gelangen und hier auf alle zusätzlichen Inhalte zugreifen zu können. In diesem Zusammenhang stellen wir auch eine Errata-Seite bereit, in der alle bekannten Fehler bzw. Updates zu den Inhalten erfasst sind. Falls Sie Fehler melden oder anderweitiges Feedback geben wollen, freuen wir uns darüber. Dies können Sie an buch@hacking-akademie.de schicken.

Noch ein Hinweis zur Online-Learning-Plattform Hacking-Akademie: Hier bieten wir als Ergänzung zum Buch eine umfassende Ausbildung zum Ethical Hacker und Penetration-Tester an. Mit praxisorientierten Videolektionen und eigener Laborumgebung erhalten Sie hier die Möglichkeit, Ihre Hacking- und Security-Skills systematisch auf- und auszubauen.

Worauf warten Sie noch?

Jetzt liegt es an Ihnen! Haben Sie das Zeug zu einem fähigen Hacker? Sie benötigen ein hohes Maß an Motivation und Neugier, Disziplin und Geduld. Hacking lernt man nicht von heute auf morgen. Hacking umfasst grundsätzlich die gesamte Palette der IT-Systeme und -Anwendungen.

Wer hier jenseits des Scriptkiddie-Niveaus erfolgreich sein möchte, beschreitet einen langen, spannenden Weg, auf dem er sehr viel lernen, aber auch immer wieder an seine Grenzen stoßen wird. Wir freuen uns, wenn wir Sie bei Ihrem Einstieg in die spannende Welt des Hackings und Penetration Testings ein Stück weit begleiten und unterstützen können.

Jetzt bleibt nur eins: Gehen Sie den ersten Schritt, beginnen Sie Ihren Weg! Bauen Sie noch heute Ihr Hacking-Labor auf und starten Sie Ihre Karriere als Ethical Hacker!

Herzliche Grüße,
Eric Amberg und Daniel Schmid

Über die Autoren



Eric Amberg ist selbstständiger Experte für IT-Netzwerke und -Sicherheit und hat in den letzten 20 Jahren zahlreiche Projekte aller Größenordnungen durchgeführt. Seine große Leidenschaft ist die Wissensvermittlung, die er in Büchern, Magazinen und insbesondere Videotrainings stets praxisnah und lebendig präsentiert. Mit der Hacking-Akademie hat Eric eine Online-Plattform zum Lernen von Ethical Hacking und Penetration Testing in deutscher Sprache entwickelt: <https://hacking-akademie.de>



Daniel Schmid ist bei einem großen Energiekonzern im Bereich Netzwerke und Security tätig. Als Projektleiter für diverse große, teils internationale Projekte hat er in über 10 Jahren viel Erfahrung in der Planung und Implementation sicherheitskritischer Infrastruktur gesammelt und hat dabei seine Leidenschaft für das Thema »Hacking und Penetration Testing« entdeckt.

Eric und Daniel haben bereits viele gemeinsame Projekte erfolgreich umgesetzt und sind die Gründer der Hacking-Akademie: <https://hacking-akademie.de>

Die perfekte Ergänzung zu diesem Buch



Nur für die Leser unseres Buches:

Exklusiver 50% Rabattcode für die Hacking-Akademie!

Werden Sie Teilnehmer der Hacking-Akademie!

Vielen Dank, dass Sie sich für dieses Buch entschieden haben. Als Dankeschön bieten wir Ihnen einen 50% günstigeren Zugang zur videobasierten Online-Learning-Plattform **Hacking-Akademie**.

Erweitern Sie Ihre Fähigkeiten mit unserem hochwertigen Lernangebot:

Das erwartet Sie:

- **Grundkurs Hacking & Security:** Das solide Fundament für den Einstieg
- **Video Lektionen:** Einfaches Lernen durch Zuschauen und Mitmachen
- **Online-Laborumgebungen:** Praxistraining in cloudbasierten HackLabs
- **CTF-Challenges:** Hacking-Herausforderungen in der Praxis
- **Community-Forum:** Fragen stellen und Mitmachen in der Community
- **Eigene Zertifikate:** Steigern der Jobchancen durch zertifiziertes Wissen

Ihr exklusiver Rabattcode:

Nutzen Sie diesen Code bei der Anmeldung auf unserer Website und erhalten Sie auf die Anmeldung **50% Rabatt:**

<https://hacking-akademie.de>

Bereit für die Herausforderung?

Dann starten Sie jetzt Ihre Ausbildung zum Ethical Hacker in der Hacking-Akademie!



Danksagung

Dieses Buch war ein echtes Mammut-Projekt, das ohne die Unterstützung von vielen Menschen nicht zu diesem bemerkenswerten Ergebnis geführt hätte. Daher möchten sich die Autoren Eric und Daniel bei allen Beteiligten herzlich für den großartigen Einsatz und die fantastische Unterstützung bedanken.

Unser besonderer Dank gilt unseren unermüdlichen Testlesern Anton Perchermeier, Martin Meinl, Markus Bauer und Timo Scheidemantel. Mit euren umfassenden, kritischen und fundierten Rückmeldungen habt ihr die hohe Qualität dieses Buchs erst ermöglicht. Wir schätzen uns glücklich, Profis aus dem IT-Security-Umfeld wie euch als engagierte Testleser zu haben. Dank euch ist der Inhalt des Buchs noch einmal deutlich aufgewertet worden.

Auch an Sabine Schulz vom mitp-Verlag geht ein herzliches Dankeschön! Liebe Sabine, du hast während der langen Entstehungszeit dieses Buchs stets zu uns gehalten und trotz vieler Verzögerungen immer mit Verständnis reagiert – das ist alles andere als selbstverständlich, hat aber auch dazu beigetragen, dass wir uns noch mehr Mühe mit dem Buch gegeben haben, damit sich die Wartezeit auch wirklich gelohnt hat.

Man sagt, hinter jedem erfolgreichen Mann steht eine starke Frau. Ob der Spruch allgemein noch zeitgemäß ist, sei dahingestellt – auf uns trifft er auf jeden Fall zu. Ohne dass unsere Partnerinnen uns den Rücken freigehalten hätten und sehr tolerant mit der vielen Zeit umgegangen wären, in der wir am Buch-Manuskript gesessen haben, wäre dieses Buchprojekt nicht realisierbar gewesen. Unser ganz besonderer Dank gilt daher unseren Ehefrauen Kati und Rocío. Ihr habt uns dabei so großartig unterstützt und mit viel Verständnis und Geduld in den letzten Jahren auf die zusätzliche Arbeitslast reagiert, die uns das Buch auferlegt hat. Nur mit eurer Hilfe konnte dieses Buch entstehen!

Berlin und Stuttgart, 16. Februar 2024

Eric und Daniel

Grundlagen Hacking und Penetration Testing

Hacker sind die Bösen! Hacker sind darauf aus, möglichst viel Schaden anzurichten und bedrohen das Internet und jeden Rechner, der daran angeschlossen ist! Also gilt es, Hackern möglichst schnell und nachhaltig das Handwerk zu legen ...

Okay, Schluss damit! Die obige Aussage ist natürlich Unsinn! Tatsache ist, dass wir Hackern diverse geniale Programme und Tools verdanken. Kennen Sie Linux? Nun, wer nicht? Wissen Sie, wer es entwickelt hat? Linus Torvalds, ein finnischer Student, der sich nicht damit abfinden wollte, dass AT&T den Quellcode zu UNIX nicht freigeben wollte und ein System benötigte, das besser auf seine Anforderungen zugeschnitten war. Daraus entstand Linux (Linus+X). Und auch wenn die meisten »Rechtschaffenen« unter uns Torvalds einen »Entwickler« nennen würden, so versteht er sich selbst doch als »Hacker«.

Es gibt also jede Menge Begrifflichkeiten zu unterscheiden. In diesem Kapitel legen wir die Grundlagen für Ihr Verständnis von Hacking und Penetration Testing. Sie lernen insbesondere Folgendes:

- Was ist Hacking?
- Verschiedene Hacker-Typen
- Motive und Absichten eines Hackers
- Was bedeutet Ethical Hacking?
- Die Zertifizierung zum Ethical Hacker (CEH)
- Die Schutzziele
- Wie funktioniert ein Penetrationstest?
- Hacking-Beispiele

In diesem ersten Kapitel beschäftigen wir uns mit den Grundlagen des Hackings. Damit Sie verstehen, was ein Hacker überhaupt ist und wo das Wort Hacking herkommt. Sie werden zudem erfahren, welche verschiedenen Hacker-Typen es gibt und wie die Ziele der Hacker aussehen. Sie lernen, was sich hinter dem *Ethical Hacking* verbirgt und warum Sie sich diesen Ehrencodex zu Eigen machen sollten.

Darüber hinaus betrachten wir auch die andere Seite. Die Schutzziele geben Aufschluss darüber, gegen welche Gefahren wir uns schützen wollen. Letztlich geht es darum, Computersysteme und -netzwerke sicherer zu machen. Der Weg ist also das Hacking, das Ziel jedoch, die IT-Sicherheit zu erhöhen. Daher werden wir ein großes Augenmerk auf den Schutz der gefundenen Schwachstellen und Angriffsvektoren legen.

Ein *Ethical Hacker* betreibt seine Tätigkeit regelmäßig im Rahmen eines beauftragten Penetrationstests. Sie lernen, wie ein solcher Test aufgebaut ist, welchen Klärungsbedarf es mit dem Auftraggeber gibt und wie ein Hacker bzw. Penetrationstester vorgeht.

Den Abschluss dieses Kapitels liefern einige bekannte Hacking-Beispiele, die Ihnen schon einmal einen gewissen Bezug zur Realität zeigen. Im Laufe dieses Buches lernen Sie noch viele weitere Möglichkeiten kennen, wie Computersysteme angegriffen werden können. Dabei gehen wir auch immer wieder auf bereits bekannte Angriffe ein und beschreiben diese.

1.1 Was ist Hacking?

In der heutigen Zeit von Informationstechnologien und Vernetzung spricht man von einem »Hacker«, wenn es um eine Person geht, die sich Zugriffe zu Netzwerken, Systemen und Anwendungen verschafft. Ohne dass der Besitzer der jeweiligen Einrichtungen das beabsichtigt hat. Doch das war nicht schon immer so.

Wo kommt denn dieses Wort überhaupt her und was ist denn Hacking eigentlich? Der Begriff »Hacking« kommt aus einer Zeit, in der nicht Netzwerke und Computersysteme im Fokus standen. Denn damit hatte der Begriff erst mal gar nichts zu tun. Es ging vielmehr darum, sich so intensiv mit einer bestimmten Technik zu beschäftigen, dass man einen Weg findet, scheinbar Unmögliches machbar zu machen. Auf Deutsch hätte man das Wort »Tüftler« verwendet.

Ein Hacker war jemand, der mithilfe von ein paar Streichhölzern, einem Gummi und einem Bleistift einen Fernseher bauen kann. Oder war das MacGyver? :) Spaß beiseite. Tatsächlich war ein Hacker ursprünglich einfach nur jemand, der sich sehr intensiv mit einer Technologie auseinandergesetzt hat, um sie zu begreifen, für sich nutzbar zu machen und ggf. zu verbessern. Ein Hacker ist nichts Bedrohliches oder Böses an sich. Dieser Ruf kam erst später durch die Medien und als es die ersten Einbrüche in fremde Systeme gab. Heutzutage hat ein Hacker in der Öffentlichkeit kein gutes Ansehen, man verbindet den Begriff in der Regel mit einem Verbrecher, der gegen das Gesetz handelt. Doch das stimmt so nicht zwangsläufig.

Aber wie kommt denn nun dieses Bild vom Hacker, der in fremde Computersysteme eindringt und allerlei Schaden anrichtet, zustande? Nun, zweifelsfrei haben Hacker eines gemeinsam: Sie sind neugierige Menschen, die neue Wege suchen, insbesondere mit Computersystemen zu arbeiten! Und einige von ihnen sind scharf auf Informationen. Dabei ist es zunächst einmal zweitrangig, ob ein Computersystem diese Informationen freiwillig bereitstellt oder nicht. Im Gegenteil versprechen gut geschützte Computer und Netzwerke sogar interessantere Informationen – proportional steigend zu den Schutzmaßnahmen.

Und so waren es natürlich auch gerade die Hacker mit ihrem tiefgreifenden Wissen über Computersysteme und -netzwerke, die, oftmals aus purer Neugier, Wege in diese Systeme gesucht und gefunden haben. In vielen Fällen wurden die gefundenen Schwachstellen dem jeweiligen Eigentümer bekannt gemacht und die möglicherweise gefundenen Daten und Informationen gar nicht verwendet – es ging nur um die Machbarkeit eines Einbruchs.

Aber wie es so ist, nutzen nicht alle ihr außerordentliches Wissen, um Gutes zu tun, diese Welt sicherer zu machen oder interessante Software unentgeltlich zur Verfügung zu stellen. Stattdessen unterliegen sie der Verlockung, ihr Expertenwissen für sich selbst zu nutzen, um sich zu bereichern.

Und genau hier grenzen sich die einzelnen Hacker-Typen voneinander ab. Denn der traditionelle Hacker im oben beschriebenen Sinne möchte keinesfalls in einen Topf mit diesen Kriminellen geworfen werden. Daher wird der »böse« Hacker auch generell als »Cracker« bezeichnet. Doch dies ist nur eine sehr globale Kategorisierung. Für eine fundierte Unterscheidung derjenigen, die sich mit dem Thema »Hacking« intensiver beschäftigen, müssen wir etwas weiter in die Tiefe gehen und neben der Motivation auch die Qualität der Tätigkeit betrachten.

1.2 Die verschiedenen Hacker-Typen

Bestimmt kennen Sie aus diversen Blockbustern die schwarzen Gestalten, die hinter einer Wand von Bildschirmen sitzen und nur von den kryptischen, grünen Zeichen beleuchtet werden, die über die Monitore rasen. Auch wenn dieses gängige Klischee tatsächlich durchaus vereinzelt bedient wird und einige Zeitgenossen auf diese Art arbeiten, gibt es doch auch ganz andere Inkarnationen der Hacker-Zunft.

Es finden sich nämlich genauso Hacker, die mit Anzug und Krawatte bei namhaften Firmen ein- und ausgehen, um deren Sicherheit zu testen. Diese Leute haben auch eine Hacking-Ausbildung, nutzen ihr Wissen allerdings nicht, um Schaden anzurichten, sondern um genau davor zu schützen – man nennt sie auch Penetrationstester bzw. kurz: Pentester. Tatsächlich gibt es aber auch böse Jungs, die Anzug und Krawatte tragen. In bestimmten Situationen gilt: Kleider machen Leute. Und wer z.B. in einer Bank ein Computer-Terminal hacken möchte, tut gut daran, optisch nicht aufzufallen. Auch für das *Social Engineering*, bei dem Informationen über Menschen anstatt über Technik gewonnen werden, ist das Auftreten oft ein wichtiger Aspekt. Näheres hierzu finden Sie in Kapitel 20 *Social Engineering*.

Nachfolgend eine Übersicht über die wichtigsten Hacker-Klassifikationen.

Scriptkiddies

Sie haben wenig Grundwissen und versuchen, mithilfe von Tools in fremde Systeme einzudringen. Dabei sind diese Tools meist sehr einfach über eine Oberfläche zu bedienen. Die Motivation ist meistens Spaß und die Absichten sind oft krimineller Natur. Oftmals möchten Scriptkiddies mit ihren Aktionen Unruhe stiften. Die Angriffe sind meist ohne System und Strategie. Viele Hacker starten ihre Karriere als Scriptkiddie, nutzen die Tools zunächst mit wenig Erfahrung, lernen aus dem Probieren, entwickeln sich weiter und finden dadurch einen Einstieg in die Szene.

Black Hats

Diese Gattung Hacker beschreibt am ehesten die Hacker, die man aus den Medien kennt. Hier redet man von Hackern mit bösen Absichten. Sie haben sehr gute Kenntnisse und greifen bewusst und strukturiert Unternehmen, Organisationen oder Einzelpersonen an, um diesen Schaden zuzufügen. Die Ziele der Black Hats sind vielfältig und reichen vom einfachen Zerstören von Daten bis hin zum Diebstahl von wertvollen Informationen, wie Kontodaten oder Unternehmensgeheimnissen. In manchen Fällen reicht es den Black Hats auch, wenn sie erfolgreich die Server ihres Opfers lahmlegen und damit Sabotage verüben.

White Hats

Einen *White Hat Hacker* nennt man oft auch einen *Ethical Hacker*. Er nutzt das Wissen und die Tools eines Hackers, um zu verstehen, wie Black Hats bei ihren Angriffen vorgehen. Im Gegensatz zum Black Hat will der White Hat jedoch die betreffenden Systeme letztlich vor Angriffen besser schützen und testet daher die Schwachstellen aktiv aus. Damit hat ein White Hat Hacker grundsätzlich keine bösen Absichten, im Gegenteil, er unterstützt die Security-Verantwortlichen der jeweiligen Organisation. White Hat Hacker oder Ethical Hacker versuchen im Anschluss an ihre Hacking-Tätigkeit, herauszufinden, welche Sicherheitslücken es gibt, und geben eine Anleitung dazu, diese möglichst effizient zu schließen.

Penetrationstester (Pentester)

Zu den White Hat Hackern gehören auch die sogenannten Penetrationstester. Hier steht grundsätzlich ein Auftrag im Hintergrund eines Angriffs. Pentester werden angeheuert, um ein bestimmtes System auf Herz und Nieren zu testen. Hier wird sehr systematisch nach Schwachstellen gesucht. Ein Penetrationstester hat eine ausdrückliche Genehmigung für sein Tun. Am Ende seiner Arbeit steht ein Bericht zur Verfügung, in dem alle gefundenen Schwachstellen dem Auftraggeber aufgezeigt werden. Dieser hat dann die Möglichkeit, die Lücken zu schließen, bevor die Black Hats ihr Glück versuchen ...

Grey Hats

Genauso wie die Farbe Grau zwischen Schwarz und Weiß liegt, so liegen die Grey Hats zwischen den Black und den White Hat Hackern. Mal haben sie gute, mal schlechte Absichten. Je nachdem was ihnen gerade lukrativ erscheint. Ein Grey Hat ist nicht grundsätzlich böse, nimmt es mit der Ethik aber auch nicht unbedingt so genau.

Cyber-Terroristen

Dies sind organisierte Gruppen, die sich gegen bestimmte Dinge auflehnen und mithilfe des Internets und seiner Technologien Angriffe durchführen. Dabei versuchen sie, möglichst viel Schaden anzurichten. In vielen Fällen ist ihr Tun politisch oder auch religiös motiviert.

Staatlich unterstützte Hacker

Hierbei handelt es sich um Hacker, die im Auftrag einer Regierung agieren. Sie wurden speziell ausgebildet und versuchen, als Agenten beispielsweise an geheime Informationen zu kommen. Das Einsatzgebiet kann der Kampf gegen den Terror sein oder auch das Sammeln von Informationen über einen Gegner in Konfliktsituationen. Insbesondere die USA, Russland und China sind hier sehr aktiv.

Suicide Hacker

Der CEH (Certified Ethical Hacker) beschreibt hier eine Ausprägung des Hackings, bei dem der Angreifer ohne Rücksicht auf Verluste vorgeht und dabei auch sich selbst der Gefahr aussetzt, entdeckt zu werden. Dabei handelt es sich ggf. nicht wirklich um Profis, sondern eher um Verzweiflungstäter, die jedoch aufgrund ihrer Kompromisslosigkeit kurzfristig hocheffektiv ihre Ziele erreichen können.

Haktivisten

Werden Systeme, insbesondere Webserver, im Internet gehackt, um auf politische Inhalte hinzuweisen und zu protestieren, sprechen wir von *Hackivismus* oder *Haktivisten*. Dabei werden in der Regel die originalen Webinhalte durch eigene Inhalte ersetzt. Diesen Prozess nennt man auch *defacen* (von engl. *Face* = Gesicht). Weitere Methoden der Haktivisten sind *Denial-of-Service-Angriffe* und *E-Mail-Spamming*. Die bekannteste Haktivist-Gruppe kennen Sie vielleicht sogar schon, die Rede ist von *Anonymous*.

Oft ist es nicht einfach, zwischen den verschiedenen Typen zu unterscheiden. Ein Black Hat Hacker kann genauso auch ab und zu ein Haktivist sein und ein White Hat arbeitet oft auch als Penetrationstester. Wichtig ist, zu wissen, dass nicht alle Hacker dieselben Absichten haben und es Hacker mit unterschiedlichsten Motiven gibt. Gutes Stichwort ...

1.3 Motive und Absichten eines Hackers

Egal, ob White oder Black Hat Hacker: Die Tools, die Techniken, die Vorgehensweise und auch das Wissen ist annähernd dasselbe. Unterschieden wird darin, welche Motive und Absichten ein Hacker hat.

1.3.1 Das Motiv

Fragen Sie einen Hacker (oder Cracker) danach, könnten Sie typischerweise folgende Antworten erhalten:

Ich möchte mich an jemandem rächen!

Rache ist kein seltenes Motiv, ob es der alte Arbeitgeber ist, der einen entlassen hat, eine Firma, mit der man Probleme hatte, oder gar die/der Ex-Partnerin/Partner. Das Ziel des Hacking-Angriffs besteht darin, jemandem Schaden zuzufügen, dem man nicht wohlgesonnen ist.

Ich möchte damit Geld verdienen!

Wer das Hacking beherrscht, dem stehen viele Türen offen. Gute White Hat Hacker sind gefragt – egal, ob sie als Security-Spezialist um die Sicherheit eines Unternehmens bemüht sind oder großen Organisationen Penetrationstests anbieten. Das White Hat Hacking ist durchaus lukrativ. Aber auch Black Hat Hacker kommen an ihr Geld, meistens allerdings durch illegale Weise wie Erpressung oder Datendiebstahl. Im Zweifel werden sie für ihre Aktivitäten von anderen bezahlt, in deren Auftrag sie ein bestimmtes Ziel verfolgen.

Ich möchte Spaß haben!

Keine Frage, Hacking macht Spaß, das werden Sie noch früh genug merken. Diese Mischung von Nervenkitzel und Erfolgserlebnis nach einem gelungenen Angriff ist sehr reizvoll. Daher gibt es viele Menschen, die sich das Hacking zum Hobby gemacht haben, eben weil es Spaß macht. Auch hier kann die Waage zur einen oder zur anderen Seite ausschlagen: Entweder nutzen Sie Ihr Wissen, um anderen zu helfen oder ihnen zu schaden ...

Ich möchte jemanden ausspionieren!

Nicht gerade die feine Art, aber es finden sich immer wieder gute Gründe, um einen Menschen, ein Unternehmen oder eine Institution auszuspionieren. Den klassischen Job eines Privat-Detektivs übernimmt in diesem Fall der Hacker. Die umfangreichsten Informationen finden sich heutzutage nicht mehr in Aktenschränken, sondern auf den Festplatten der Computer einer Person oder Institution. Daher ist der Einsatz von Hacking-Methoden sehr vielversprechend, um an sensible Informationen zu gelangen.

Ich möchte etwas bewegen!

Auch Aktivismus ist oft ein Motiv zum Hacken – daher der bereits oben beschriebene Begriff *Hack-tivismus*. Es gibt eine Vielzahl von Angriffen auf politische Parteien bzw. Länder, Bewegungen und Firmen. Man muss hierzu heutzutage nicht mehr auf die Straße gehen, der Protest kann auch virtuell stattfinden, wie wir bereits weiter oben dargelegt haben.

Ich möchte im Mittelpunkt stehen!

Meldungen über Hacking-Angriffe sind aus den Medien kaum noch wegzudenken. Möchten Sie auch mal in der Zeitung stehen? Dazu ist nur ein richtiger Angriff an der richtigen Stelle notwendig. Natürlich wäre es nicht gut, wenn Sie Ihren Namen unter einem Fahndungsfoto sehen sehen. Meist verbergen sich Hacker daher hinter Pseudonymen oder Gruppen. Bekannte Hacking-Gruppen sind zum Beispiel *Anonymous*, *AntiSec* oder *LulzSec*.

1.3.2 Ziel des Angriffs

Warum ein Hacker einen Angriff ausführt, haben wir also geklärt; stellt sich noch die Frage, was er genau vorhat. Welche Absichten können also hinter einem Hacking-Angriff stecken? Betrachten wir die wichtigsten:

Datendiebstahl

Der Angreifer ist auf geheime Daten seiner Opfer aus, er möchte an Informationen kommen. Daher geht er gezielt auf die Suche nach bestimmten Dateien oder Datensätzen. Die Daten können dann gewinnbringend weiterverkauft, gegen das Opfer verwendet oder erst gegen ein Lösegeld wieder freigegeben werden.

Manipulation

Auch hier sucht der Angreifer nach Daten, aber nicht, um diese an sich zu bringen, sondern um sie zu verändern. Das kann insbesondere bei finanziellen Transaktionen teilweise gravierende Folgen haben. Stellen Sie sich einmal vor, das Komma auf Ihrem monatlichen Gehaltszettel wäre um eine Stelle nach rechts verschoben ... und nun stellen Sie sich Ihren Arbeitgeber vor. Wo es Gewinner gibt, existieren immer auch Verlierer!

Erpressung

Mit gestohlenen oder manipulierten Daten kann der Angreifer das Opfer natürlich auch erpressen: Zahlt der Betroffene nicht die geforderte Summe, so werden z.B. Firmen-Internas veröffentlicht oder ein zentrales System lahmgelegt.

Eine Variante hierzu ist der Einsatz von *Ransomware*. Dabei werden die Daten des Opfers verschlüsselt und der Schlüssel nur gegen Zahlung eines Geldbetrags (engl. Ransom) übermittelt.

Rechte erweitern

In den meisten Fällen steckt dahinter die Absicht, den Angriff effektiv fortzuführen. Es wird versucht, an möglichst viele Rechte und Privilegien zu gelangen, um damit eine möglichst umfassende Kontrolle über das Zielsystem zu bekommen. Stellen Sie sich vor, Sie melden sich als normaler Benutzer an einem System an und erlangen durch Hacking-Methoden Administrator-Privilegien. Von diesem Moment an stehen Ihnen alle Türen offen, sodass Sie z.B. neue Software installieren oder die Systemkonfiguration ändern können. Somit ist die Rechte-Erweiterung (auch als *Privilegien-Eskalation* bzw. gängiger *Privilege Escalation* bekannt) selten Selbstzweck, sondern in der Regel Mittel zum Zweck.

Unerlaubt etwas steuern

Viele Systeme haben die Aufgabe, etwas zu steuern. Denken Sie hierbei an Verkehrsleitrechner, Sicherheitszentralen, Maschinensteuerungen usw. Hat man sich einmal in die Sicherheitszentrale ein-

gehackt, spart man sich das Brecheisen. Ist es z.B. einem Hacker möglich, sich in die Kontrollsysteme eines Kernkraftwerks zu hacken, kann das fatale Folgen bis hin zum Super-GAU haben. Sie halten das für weit hergeholt? Dann warten Sie mal ab, bis Sie die perfiden Methoden von *Stuxnet* kennengelernt haben, einer Wurmsoftware, die wir Ihnen in Abschnitt 1.8.2 dieses Kapitels vorstellen.

Geld stehlen

Viele Angriffe finden auch auf Banken und Geldautomaten statt. Das Ziel der Begierde ist der schnöde Mammon – also Geld. Mal ehrlich: Haben Sie nicht auch schon davon geträumt, einen Geldautomaten so zu manipulieren, dass er unbegrenzt Geld ausspuckt? Wir zeigen Ihnen ... NICHT, wie es geht! Aber es gibt Techniken und Methoden, um sich zu bereichern, auch ohne den Bankautomaten aus dem Fundament zu reißen. In einigen Fällen werden Bankautomaten mit veralteter (und damit anfälliger) Software, wie z.B. Windows XP betrieben. Über Remote-Zugriff ist es möglich, entsprechende Schadsoftware zu installieren, um damit die Bankautomaten zu manipulieren.

Darüber hinaus ist es natürlich auch durch die Manipulation von Kontenbewegungen und Finanzsoftware möglich, Geld auf das eigene Konto auf den Bahamas transferieren zu lassen. Wie Sie feststellen, ist dieses Hacking-Ziel in der Regel durch Manipulation zu erreichen, die wir weiter oben bereits grundlegend als übergeordnetes Hacking-Ziel ausgemacht haben.

Ruf ruinieren

Wie Sie schon wissen, können die Motive für Hacking auch Rache oder Aktivismus ein. Die Absicht, einen Ruf zu ruinieren, kann auf verschiedene Art und Weise umgesetzt werden. Eine Möglichkeit besteht darin, einen erfolgreichen Angriff bekannt werden zu lassen. Stellen Sie sich z.B. vor, in den Medien wird von einem erfolgreichen Hacking-Angriff auf eine Bank berichtet. Das richtet großen Image-Schaden an.

Zugang/Service blockieren

Eine der häufigsten Angriffsformen ist der *Denial-of-Service-Angriff* (DoS). Dabei versucht der Angreifer, das Opfer-System oder -Netzwerk derartig zu überlasten, dass der angebotene Dienst (in der Regel Webanwendungen) nicht mehr für reguläre Anfragen oder Zugriffe erreichbar ist. DoS-Angriffe kommen in ganz verschiedenen Varianten vor. Im Internet wird häufig ein *Distributed-Denial-of-Service-Angriff* (DDoS) durchgeführt, wobei Hunderte oder sogar Tausende Systeme zentral gesteuert werden und synchronisiert einen Angriff starten (sogenannte Botnetze).

1.4 Ethical Hacking

Sie lernen in diesem Buch eine ganze Menge über das Hacking. Dieses Wissen können Sie für die verschiedensten Zwecke einsetzen. An dieser Stelle möchten wir jedoch noch einmal ganz ausdrücklich an Ihren ethischen Kompass appellieren!

Was du nicht willst, das man dir tu' ...

Das Ziel dieses Buches ist *offensive IT-Sicherheit*. Das bedeutet, dass Sie als jemand, der sich mit den Methoden und Techniken der bösen Jungs (und Mädels) auskennt, Ihr Wissen nutzen, um die Sicherheit von Computersystemen zu erhöhen, indem Sie deren Schwachstellen aufdecken und helfen, diese zu beseitigen. Dies wird als *Ethical Hacking* bezeichnet. Es dient ausschließlich der Sicherheit von Computersystemen und bezeichnet den verantwortungsvollen Umgang mit dem Know-how des Hackings.

Als Ethical Hacker verpflichten Sie sich, Schaden von Computersystemen abzuwenden und niemals absichtlich zu verursachen. Sie handeln nach dem Motto: »Was du nicht willst, das man dir tu‘, das füg‘ auch keinem anderen zu!«

Lernen Sie so viel über das Hacking wie möglich und seien Sie immer neugierig – doch die Freiheit des einen hört dort auf, wo die Freiheit des anderen eingeschränkt wird! Greifen Sie niemals ohne schriftliche Genehmigung und eindeutige Auftragsklärung fremde Systeme an. Das Wissen über theoretische und praktische Hacking-Technologien verpflichtet. So wie ein Kampfsportler seine Fähigkeiten nur im Ring bzw. auf der Matte und nicht auf der Straße anwenden darf, so bleibt ein Ethical Hacker immer im ethischen und rechtlichen Rahmen des Erlaubten. Gutes Stichwort, dazu gibt es noch etwas Wichtiges zu erläutern.

Der Hacker-Paragraf

Im Jahr 2007 wurde im Rahmen der »Strafvorschriften zur Bekämpfung der Computerkriminalität« der Paragraf 202c des Strafgesetzbuches (StGB) eingeführt. Er lautet folgendermaßen:

(1) Wer eine Straftat nach § 202a oder § 202b vorbereitet, indem er

1. Passwörter oder sonstige Sicherungscodes, die den Zugang zu Daten (§ 202a Abs. 2) ermöglichen, oder

2. Computerprogramme, deren Zweck die Begehung einer solchen Tat ist,

herstellt, sich oder einem anderen verschafft, verkauft, einem anderen überlässt, verbreitet oder sonst zugänglich macht, wird mit Freiheitsstrafe bis zu zwei Jahren oder mit Geldstrafe bestraft.

(2) § 149 Abs. 2 und 3 gilt entsprechend.

Das umfasst grundsätzlich auch die Hacker-Tools, deren sich nicht nur die bösen Jungs, sondern auch Administratoren und Sicherheitsbeauftragte bedienen, um die Sicherheit von Computersystemen und -netzwerken zu erhöhen. Bevor Sie jetzt jedoch aus rechtlichen Bedenken dieses Buch zuschlagen und sich dem Fernsehprogramm widmen, dürfen wir Sie beruhigen: Auch wenn der Wortlaut hier leider sehr schwammig ist und eine weitgefaste Auslegung zulassen würde, so dient der Paragraf seinem Inhalt nach nur der Vereitelung von Straftaten.

Die bisherige Rechtsprechung zeigt, dass die Verwendung dieser Tools zur Erhöhung der Sicherheit von IT-Infrastrukturen keine Strafverfolgung nach sich zieht. Dennoch bleibt eine gewisse rechtliche Unsicherheit. Der entsprechende Wikipedia-Artikel ist sehr aufschlussreich und einen Blick wert: https://de.wikipedia.org/wiki/Vorbereiten_des_Ausspähens_und_Abfangens_von_Daten. Sichern Sie sich beim Hacking bzw. Penetration Testing in fremden Umgebungen immer schriftlich und umfangreich ab, indem Sie Art und Umfang Ihrer Tätigkeit (bzw. des Penetrations-tests) ganz genau beschreiben und anschließend auch ausführlich dokumentieren.

1.5 Der Certified Ethical Hacker (CEHv12)

Dieses Buch versteht sich als eine fundierte, praxisorientierte Einführung in das Thema »Ethical Hacking«. Es ist an die Inhalte der Prüfung zum *Certified Ethical Hacker* (CEHv12) angepasst und stellt somit eine wertvolle Ressource für Ihre Vorbereitung auf das Examen dar. Auch wenn der Fokus nicht primär auf der Prüfungsvorbereitung liegt, werden wir im Laufe des Buches immer wieder Hinweise zur Prüfung geben. An dieser Stelle möchten wir Ihnen einmal kurz den CEH vorstellen.

1.5.1 Was steckt dahinter?

Der *Certified Ethical Hacker* ist eine herstellerunabhängige Zertifizierung, die vom EC-Council (www.eccouncil.org) entwickelt und angeboten wird. Dahinter verbirgt sich eine Organisation, die sich auf Zertifizierungen im Hacking- und Security-Bereich spezialisiert hat.

Der CEH ist mittlerweile in der Version 12 verfügbar (siehe hierfür <https://www.eccouncil.org/cybersecurity-exchange/ethical-hacking/cehv12-new-learning-framework/>). Er stellt eine anspruchsvolle Basiszertifizierung für angehende Ethical Hacker und Penetrationstester dar, die durch weitergehende Zertifizierungen ergänzt wird. So steht seit dem CEHv10 optional eine ergänzende CEH-Practical-Zertifizierung zur Verfügung. Dabei handelt es sich um eine praktische Prüfung, bei der der Kandidat seine Hacking-Kenntnisse in einer praxisnahen Laborumgebung unter Beweis stellen muss. Inzwischen führen diese beiden Prüfungen zusammen zum *CEH Master*, um den Mehrwert hervorzuheben (<https://www.eccouncil.org/train-certify/ceh-master/>).

Wer sich darüber hinaus noch weiter in den professionellen Bereich begeben möchte, kann über den *EC-Council Certified Penetration Testing Professional* (CPENT) den nächsten Schritt gehen und auch die Expert-Level-Zertifizierung zum *Licensed Penetration Tester* (LPT) absolvieren, der allerdings hohe Einstiegshürden aufweist. Mittlerweile bietet das EC-Council eine Vielzahl von Zertifizierungen und Zertifizierungspfaden an.

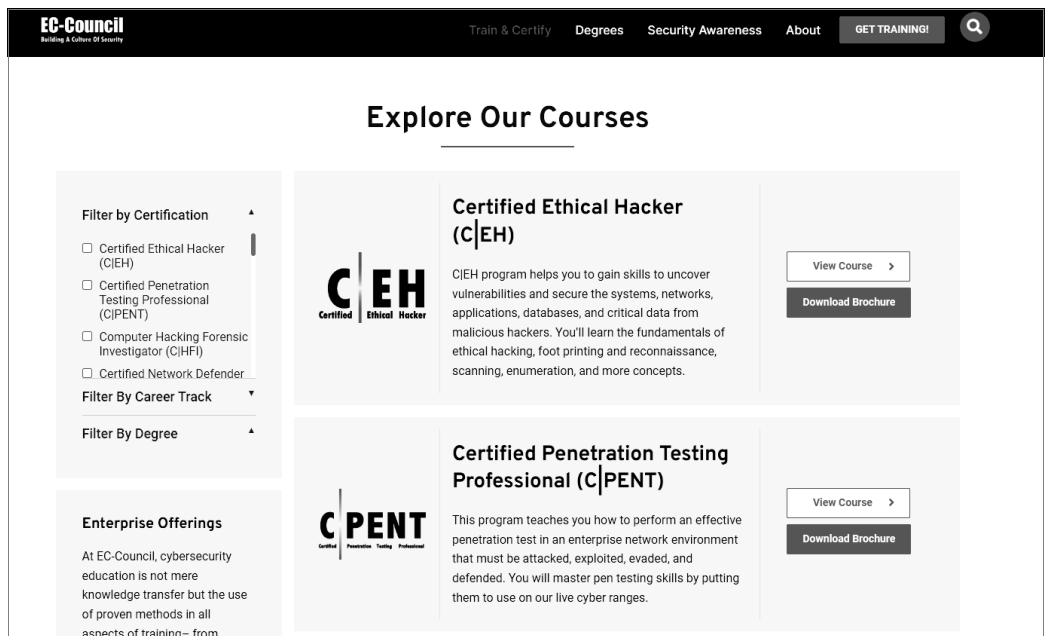


Abb. 1.1: Zahlreiche Kurse und Zertifizierungen sind beim EC-Council verfügbar.

Das Curriculum des CEHv12 umfasst insgesamt 20 Module, deren Inhalte in diesem Buch abgedeckt sind. Es wird ein breites Themen-Spektrum mit diversen Konzepten und unzähligen Tools abgearbeitet, wobei es hauptsächlich um Konzepte und Technologien geht und weniger darum, alle der vorgestellten Tools bis ins Detail zu beherrschen. Den Prüfling erwartet ein intensives Studium,

das ein hohes Engagement und intensive Einarbeitung voraussetzt, um alle behandelten Themen in ausreichender Tiefe zu beherrschen.

Neu im Angebot des CEHv12 sind eine höhere Praxisorientierung und Unterstützung nach der eigentlichen Prüfung. ECCouncil nennt das »Learning Framework« und unterteilt das Lernsystem in vier Stufen:

- **Learn:** Der Teilnehmer absolviert den Kurs oder lernt im Rahmen des Online-Kurses.
- **Certify:** Der Teilnehmer absolviert die Prüfung.
- **Engage:** Der Teilnehmer kann seine Skills in Capture-The-Flag-Umgebungen (CTF) praktisch trainieren
- **Compete:** Im sogenannten »Hackerverse« werden monatliche CTF-Challenges bereitgestellt, in denen die Kandidaten gegeneinander antreten und Punkte im Leaderboard sammeln können.

Insgesamt wurde das Angebot damit deutlich aufgewertet.

1.5.2 Die CEHv12-Prüfung im Detail

Zur CEHv12-Prüfung werden Sie unter einer der folgenden Bedingungen zugelassen:

1. Sie absolvieren einen der offiziellen (und nicht gerade günstigen!) CEH-Kurse. Damit sind Sie automatisch qualifiziert für die Prüfung.
2. Sie reichen ein »Egibility Form« (ein Formular für die Zulassung zur Prüfung) ein und weisen nach, dass Sie mindestens zwei Jahre Erfahrung auf dem Gebiet der IT-Sicherheit haben. Diese Zulassungsprüfung kostet Sie derzeit 100 Dollar – unabhängig vom Ausgang der Prüfung.

Im Gegensatz zum Themenspektrum und dem Inhalt des CEH-Curriculums ist die Prüfung derzeit eher geradlinig gehalten:

- Anzahl der Fragen: 125
- Maximale Testdauer: vier Stunden
- Test-Format: Multiple Choice mit nur einer richtigen Antwort
- Test wird angeboten über: VUE-Testcenter oder ECC-Online-Examen
- Test-Nummer: 312-50

Es gibt eine Aufschlüsselung in Themenkomplexe und deren Schwerpunkte, aber diese wird in regelmäßigen Abständen geändert. Die Prüfung wirkte in der Vergangenheit mitunter unausgeglich. Ein bisher überdimensionierter Schwerpunkt lag auf Nmap-Befehlen und auf kryptografischen Konzepten. Dies ist jedoch keine Garantie für Ihren Prüfungszeitpunkt. Von daher empfehlen wir Ihnen, sich im Internet in einschlägigen Foren Informationen zur Prüfung einzuholen, wenn Ihr Prüfungszeitpunkt konkret wird.

Unter dem Strich ist die Zertifizierung zum CEH eine gute Ergänzung zur Schärfung Ihres Profils und kann Ihre Karrierechancen deutlich verbessern. Sie ist allerdings mit derzeit 950 bzw. 1200 Dollar sehr teuer. Der Preis ist abhängig davon, ob Sie die Prüfung im ECC Exam Center oder in einem VUE-Prüfungcenter absolvieren möchten.

Sie sollten insbesondere in folgenden Szenarien über eine CEH-Zertifizierung nachdenken:

- Sie möchten zukünftig als Penetrationstester arbeiten und benötigen einen Nachweis Ihrer Qualifikation.

- Ihre Tätigkeit liegt im IT-Security-Bereich und Sie möchten Ihr Einsatzgebiet erweitern.
- Sie arbeiten als Security Analyst und möchten Ihr Wissen zertifizieren.

Wir halten die Zertifizierung für ein gutes Fundament für den Einstieg in eine Karriere als Ethical Hacker und Penetrationstester. Um aus diesem Buch das Maximum herauszuholen, ist jedoch die Prüfung zum CEH keine Voraussetzung. Trotzdem werden wir immer wieder auf die CEH-Prüfung zurückkommen und Tipps und Prüfungshinweise geben.

1.6 Die Schutzziele: Was wird angegriffen?

Distanzieren wir uns für einen Moment von unserer Hacker-Rolle und setzen die Brille derjenigen auf, die Computersysteme und deren Daten schützen müssen. Denn Hacking und Penetration Testing dient aus Sicht der Offensive Security zur Absicherung der Systeme. Betrachten wir also den Blickwinkel des Security-Verantwortlichen einer Organisation.

Die IT-Sicherheit definiert drei grundlegende Schutzziele, die durch Angriffe auf IT-Systeme bedroht werden. Sie werden mit **C I A** abgekürzt. Dies steht in diesem Fall nicht für Central Intelligence Agency, sondern ist eine Abkürzung für:

- **Confidentiality** = Vertraulichkeit
- **Integrity** = Integrität
- **Availability** = Verfügbarkeit

Manchmal wird ein viertes Schutzziel, die **Authenticity** (= Authentizität) definiert. Diese dient auch der **Non-Repudiation**, was etwas hölzern als *Nicht-Abstreitbarkeit* übersetzt wird. Dieses Thema wird aber oft im Schutzziel **Integrität** enthalten gesehen.

Tip: Kompromittierte Systeme sind per se nicht mehr sicher

Unter dem Strich möchten die Sicherheitsverantwortlichen hauptsächlich sicherstellen, dass die Daten und Systeme nicht *kompromittiert* werden. Bei einem kompromittierten System kann der Eigentümer sich nicht mehr sicher sein, dass die darauf enthaltenen Daten unverändert bzw. nach wie vor vertraulich sind und die korrekte Funktion der Dienste noch gegeben ist. Ein kompromittiertes System sollte meistens von Grund auf neu aufgesetzt werden.

Umgekehrt ist es also das Ziel von Hackern, Computersysteme zu kompromittieren und damit ganz oder teilweise unter ihre Kontrolle zu bringen. Eine Ausnahme stellen die destruktiven *Denial-of-Service-Angriffe* dar, bei denen es nur darum geht, dass das gesamte System oder Teile des Systems nicht mehr funktionieren.

Kaum zu glauben, dass sich der Schutzbedarf von Computersystemen auf die oben genannten drei bzw. vier Schutzziele herunterbrechen lässt. Sehen wir uns daher die einzelnen Schutzziele aus Sicht der IT-Sicherheit einmal im Detail an:

1.6.1 Vertraulichkeit

Es gibt Daten, bei denen ist es dem Eigentümer egal, ob sie öffentlich zugänglich sind oder nicht. Oftmals ist es aus Sicht des Eigentümers sogar wünschenswert, wenn diese Daten Beachtung finden. Hierzu zählen zum Beispiel:

- **Unternehmensadresse(n):** Zumindest die meisten Unternehmen leben davon, gefunden zu werden.
- **Marketing-Materialien:** Stellen Sie sich vor, ein Unternehmen erstellt Werbespots, veröffentlicht diese aber nicht ... das ginge dann ziemlich am Sinn vorbei.
- **Produkt-Beschreibungen:** Soll das Produkt verkauft werden, müssen potenzielle Käufer einen Einblick in die Eigenschaften des Produkts erhalten können, z.B. in Form eines Downloads von PDF-Dateien von der Website.
- **White-Paper:** Diese Übersichtsdokumente enthalten Erläuterungen zu Technologien, Fallstudien und Ansätze für Problemlösungen. Sie dienen der Öffentlichkeitsarbeit.
- **Give-Aways:** Kleine Geschenke erhalten die Freundschaft. Kostenlose Downloads oder klassische Geschenke, wie Kugelschreiber oder Tassen, erhöhen die Kundenbindung.

Die obige Aufzählung ist nur exemplarisch. Es gibt noch jede Menge weiterer Informationen, die öffentlich zugänglich sind und es aus der Sicht des Eigentümers auch sein sollen.

Andererseits sind die meisten Daten und Informationen von Personen, Unternehmen und Organisationen schützenswert und sollten oder dürfen der Öffentlichkeit nicht zugänglich gemacht werden. Eine Veröffentlichung bedeutet im besten Falle Image-Schaden und im schlimmsten Fall den Untergang des Unternehmens.

Stellen Sie sich vor, ein Unternehmen entwickelt ein neues, hoch-innovatives Produkt, mit dem es eine Alleinstellung auf dem Markt anstrebt. Alle finanziellen Ressourcen werden in diese Entwicklung gesteckt. Leider gelingt es einem Hacker, die Pläne und alle Detailinformationen des Produkts zu stehlen und einem anderen Unternehmen zukommen zu lassen, das das Produkt schneller fertigstellt und auf den Markt bringen kann. Da kann unser Unternehmen dann vermutlich dichtmachen. Übrigens fällt dieser Vorfall unter die Rubrik *Wirtschaftsspionage* und ist eine der am weitesten verbreiteten und lukrativsten Tätigkeiten von Black Hats und staatlich unterstützten Hackern.

Die Vertraulichkeit von Daten kann auch aus Datenschutzgründen notwendig sein. So müssen personenbezogene Daten von Kunden eines Unternehmens unbedingt vor unbefugtem Zugriff geschützt werden. Eine Veröffentlichung von Kundendaten geht in der Regel mit einem enormen Image-Schaden einher und kann auch für jeden einzelnen Kunden sehr teuer werden, wenn diese Daten dazu geeignet sind, der jeweiligen Person oder Organisation zu schaden. Dies ist z.B. bei Kreditkartendaten der Fall. (So geschehen 2011 bei Sonys Playstation Network.) Auch die Veröffentlichung von Patientendaten ist hochkritisch.

Die Vertraulichkeit ist also für viele Daten essenziell. Da nicht alle Daten den gleichen Schutzbedarf haben, werden oftmals Schutzklassen bzw. Sicherheitsstufen (z.B. *öffentlich*, *sensibel*, *geheim*, *Top Secret*) definiert, denen die jeweiligen Daten zugeordnet werden. In Deutschland existiert hierzu mit DIN 66399 sogar eine Norm.

Je nach Schutzklasse und Sicherheitsstufe wird in diesem Zusammenhang der jeweilige Sicherheitsbedarf festgelegt. Je höher, desto mehr und umfangreichere Sicherheitsmechanismen werden zum Schutz der Daten bereitgestellt und desto strenger sind die Kontrollen. Dies erklärt andererseits auch, warum (böartige) Hacker insbesondere von den besonders geschützten Daten angezogen werden wie die Motten vom Licht.

Auf der anderen Seite gibt es für alle relevanten Daten immer auch Personen, die auf die jeweiligen Daten zugreifen müssen. Es ist also zum einen notwendig, die autorisierten Zugriffe festzulegen, und zum anderen, dafür zu sorgen, dass nicht-autorisierte Zugriffe unterbunden werden. Dabei erhält ein Benutzer oder eine Benutzergruppe in der Regel eine eindeutige Kennung (ID) und eine

Möglichkeit, sich zu authentisieren. Ist seine *Authentizität* festgestellt, erhält er Zugriff auf diejenigen Daten, für die er *autorisiert* ist. In Abschnitt 1.6.4 gehen wir weiter in die Details der Authentisierung.

Schutzmaßnahmen

Die Maßnahmen zur Sicherstellung der Vertraulichkeit können ganz unterschiedlich aussehen und auf unterschiedlichen Ebenen ansetzen. Typische Sicherheitssysteme in Computernetzwerken sind:

- **Firewalls:** Klassisches Instrument zur Steuerung von Netzwerk-Traffic und Verhinderung von unerwünschter Kommunikation.
- **Virenschutzsysteme:** Auch Antivirus-Systeme (kurz: AV) genannt. Dienen zum Verhindern von *Malware* (böartiger Software).
- **Intrusion-Detection/Prevention-Systeme:** Kurz: IDS/IPS, dienen der Erkennung von Angriffsmustern und – im Falle von IPS – der automatischen Abwehr des Angriffs.
- **Application Gateways:** Analysieren die Kommunikation auf Protokollebene bis in die Details und können fehlerhafte und unerwünschte Kommunikation erkennen und blockieren.
- **Zugangskontrollsysteme:** Sowohl physische als auch logische Systeme dienen dazu, den Zugriff auf zu schützende Daten auf die autorisierten Personen zu beschränken.

Die wohl wichtigste Maßnahme zur Sicherstellung der Vertraulichkeit im Rahmen der Netzwerk-Kommunikation ist die *Verschlüsselung*. Sie stellt sicher, dass ein Angreifer den Inhalt einer Kommunikation nicht erkennen kann.

Vorsicht: Verschlüsselung verhindert nicht Veränderung

Bei einem *Man-in-the-Middle-Angriff* positioniert sich der Angreifer zwischen den Kommunikationspartnern und übernimmt unbemerkt jeweils stellvertretend für den anderen die Kommunikation. Beide Kommunikationspartner glauben, dass sie mit dem jeweils anderen kommunizieren, während der Angreifer jedes Datenpaket abfangen, analysieren, ggfs. verändern und dann an den echten Empfänger weiterleiten kann. Die Verschlüsselung verhindert, dass der Angreifer die Daten entziffern kann, jedoch nicht, dass sie verändert weitergeleitet werden.

Um sicherzustellen, dass die gesendeten Daten unverändert beim Empfänger ankommen oder auf einem Datenträger abgelegte Daten zwischenzeitlich nicht verändert wurden, müssen wir die *Integrität* der Daten wahren.

1.6.2 Integrität

Es war einmal ein Mitarbeiter, dem von seinem Unternehmen gekündigt wurde. Dieser war ob der Kündigung erzürnt und wollte sich an seinem Unternehmen rächen. Zu diesem Zwecke erlernte er das Hacking und führte eine *Man-in-the-Middle-Attacke* aus, indem er ausgehende Angebotsmails des Unternehmens abfing und verändert an den Adressaten weiterleitete. Immer, wenn das Unternehmen ein Dienstleistungsangebot mit einem guten Preis an einen Interessenten aussendete, veränderte er den Preis derart, dass die Dienstleistung viel zu teuer wäre – statt 1500 Euro las der Interessent nun 15.000 Euro als Gesamtpreis, lachte kurz und wandte sich von diesem Unternehmen ab, um die Dienstleistung bei einem anderen Unternehmen einzukaufen ...

Dem Unternehmen ging viel Geld dadurch verloren und der ehemalige Mitarbeiter erhielt seine Rache. Ende der Geschichte.

Tatsächlich ist die Frage, ob gesendete Daten beim Empfänger unverändert ankommen, oftmals essenziell – dabei geht es nicht immer um Geld. Es gibt populäre Fälle, in denen eine renommierte Software auf dem Server so manipuliert wurde, dass sie auf dem Opfer-System eine sogenannte »Backdoor« installierte, um Angreifern einen unbemerkten Remote-Zugang zum System zu ermöglichen.

Angriffe der oben beschriebenen Art können verhindert werden, wenn es gelingt, die Integrität der Daten sicherzustellen. Wir betrachten also die »Echtheit« der Daten. Das Ziel ist es, Daten vor Manipulationen zu schützen.

Wie bereits dargelegt, können das Dateien sein, die auf einem Server liegen und unbemerkt gegen eine manipulierte Version ausgetauscht, oder Informationen, die bei der Übermittlung manipuliert werden, wie in unserem Eingangsbeispiel.

Es muss sichergestellt werden, dass die Daten, die den Sender verlassen, auch genauso beim Empfänger ankommen und unterwegs nicht verändert oder ausgetauscht werden. Neben veränderten Inhalten kann aber auch der Absender eines Datenpakets manipuliert werden. Hierbei geht es dann um Authentizität, die ebenfalls mit Mitteln der Integrität sichergestellt werden kann.

Schutzmaßnahmen

Um die Integrität von Daten zu gewährleisten, kommt oft ein sogenannter *Hashwert* zum Einsatz. Das ist eine mathematische Funktion, die auf eine Nachricht oder eine Datei angewendet werden kann. Dabei wird die Original-Nachricht als Eingangswert von der Hash-Funktion verarbeitet. Daraus entsteht eine immer gleich lange Kombination aus Zeichen, das ist der Hashwert. Von diesem lässt sich nicht auf den Inhalt der Nachricht zurückschließen, aber er identifiziert diese ganz genau.

Wie der Fingerabdruck eines Menschen eine Person identifiziert, aber keinerlei Informationen zu Größe, Gewicht oder Haarfarbe preisgibt, so verschickt der Sender seine Nachricht inklusive Hashwert an den Empfänger. Dabei muss er den Hashwert so schützen, dass der Angreifer diesen nicht unerkannt ändern kann. Dies geschieht z.B. mittels digitaler Signatur.

Der Empfänger wendet dieselbe Hash-Funktion auf die Nachricht an und vergleicht den ermittelten Hashwert mit dem des Senders. Wurde an der Nachricht nur ein einziges Zeichen verändert, stimmt der Hashwert nicht überein. Damit kann der Empfänger die Echtheit der empfangenen Daten überprüfen.

Vorsicht: Die Integritätsprüfung verhindert nicht die Manipulation der Daten!

»Moment mal!«, werden Sie vielleicht sagen: »Mit der Integritätsprüfung will ich doch die Echtheit der Daten sicherstellen?« Jupp! Das können Sie auch – was Sie aber *nicht* können, ist, zu *verhindern*, dass die Daten manipuliert werden. Sie können es lediglich erkennen und entsprechend reagieren. Mehr kann die Integritätsprüfung nicht leisten. Ein kleiner, aber feiner und wichtiger Unterschied.

Was also tun, wenn wir bemerken, dass die Integrität von Daten nicht gewahrt werden konnte? In diesem Fall muss die Nachricht oder Datei verworfen werden, sie ist nicht mehr vertrauenswürdig. Im Fall einer Netzwerk-Kommunikation muss der Absender seine Informationen erneut senden. Dumm nur, wenn die dazu notwendigen Systeme aufgrund eines Angriffs den Dienst versagen.

Dieser Punkt betrifft das dritte Sicherheitsziel, die Verfügbarkeit von Daten in der gewünschten Art und zum gewünschten Zeitpunkt.

Auf das Thema Kryptografie gehen wir aufgrund seiner Bedeutung noch einmal gesondert ein. In Kapitel 5 erfahren Sie viele Details über Verschlüsselungsvarianten, -algorithmen und -verfahren.

1.6.3 Verfügbarkeit

Vielleicht erinnern Sie sich noch an Weihnachten 2014, als die Netzwerke der Spielekonsolen von Sony und Microsoft lahmgelegt wurden? Die neuen Spiele, die zum Fest verschenkt wurden, konnten erst einmal nur begrenzt zum Einsatz kommen, was den Herstellern viel Ärger einbrachte.

Ursache dafür war ein sogenannter *DoS-Angriff* (Denial-of-Service). Dabei versuchen Angreifer, ein System in die Knie zu zwingen, bis es seinen Dienst quittiert. Dies geschieht zum Beispiel durch eine Flut von Anfragen an das Zielsystem oder durch Ausnutzen einer bekannten Schwachstelle, die das System zum Absturz bringt. In diesem Fall reicht manchmal schon ein einziges, entsprechend manipuliertes Datenpaket.

Angreifer versuchen mittels der oben beschriebenen Denial-of-Service-Angriffe (DoS), die Verfügbarkeit von Systemen im Netzwerk und im Internet zu untergraben. Oftmals geschieht dies mit der Brechstange, indem die Opfer-Systeme mit so vielen Anfragen überhäuft werden, dass sie diese nicht mehr verarbeiten können.

Um die Wirksamkeit dieser Angriffe zu erhöhen, werden *Distributed-Denial-of-Service-Angriffe* (DDoS, sprich: Di-Dos) gefahren, bei denen der Angriff von Hunderten oder Tausenden Systemen aus dem Internet stattfindet. Hierzu dienen sogenannte »Botnetze«, bei denen eigentlich harmlose Computer zu einem früheren Zeitpunkt mit einer Software infiziert wurden, die ferngesteuert einen Angriff zu einem gewünschten Zeitpunkt initiiert.

Schutzmaßnahmen

Sich gegen einen DoS- oder DDoS-Angriff zu schützen, ist eine der schwierigsten Angelegenheiten der IT-Sicherheit. Im März 2013 fand aus Rache am Blacklist-Anbieter *Spamhaus* ein DDoS-Angriff statt, der eine Woche dauerte. Initiiert wurde er vom niederländischen Provider Cyberbunker, der sich dagegen wehren wollte, dass Spamhaus diverse seiner Kunden auf die schwarze Liste (Blacklist) gesetzt hatte, weil diese Spam und anderen unerwünschten Traffic erzeugt hatten. Der DDoS-Angriff war derart heftig, dass ein nicht unerheblicher Teil des Internets davon betroffen war und es auch andernorts zu Leistungseinbußen kam.

Für viele Unternehmen und Organisationen ist die Verfügbarkeit des Computernetzwerks und seiner Systeme essenziell. Daher werden diverse Maßnahmen ergriffen, um dies sicherzustellen. Hierbei können verschiedene Technologien zum Einsatz kommen, zum Beispiel:

- **High Availability (HA):** Auch hierbei werden redundante Systeme bereitgestellt, die entweder parallel aktiv oder im Aktiv/Passiv-Modus arbeiten, also die Funktion sofort übernehmen können, wenn das Hauptsystem ausfällt. Bei HA ist es nicht unbedingt erforderlich, dass die Systeme als Cluster arbeiten.
- **Clustering:** Dabei werden mehrere gleichartige Systeme zu einem Verbund zusammengeschlossen. Fällt eines oder sogar mehrere dieser Verbundsysteme aus, können die anderen die Funktion trotzdem aufrechterhalten. Clustering unterscheidet sich von High Availability insofern, als es die Bereitstellung eines gemeinsamen Speichers erfordert, *Quorum* genannt.

- **Loadbalancing:** Dahinter versteckt sich das Konzept, die Anfragen von Client-Systemen automatisch nach bestimmten Kriterien auf verschiedene, gleichartige Systeme zu verteilen, um die Last aufzuteilen.

Es existieren diverse weitere Technologien speziell zur Vermeidung von DDoS-Angriffen, wie z.B. Scrubbing-Center und Content-Delivery-Netzwerke. Im Internet existieren Dienstanbieter, die sich auf die Erhaltung der Verfügbarkeit der Systeme spezialisiert haben. Wir kommen in Kapitel 22 *DoS- und DDoS-Angriffe* darauf zurück.

1.6.4 Authentizität und Nicht-Abstreitbarkeit

Was passiert hinter den Kulissen, wenn Sie sich an einem Computer anmelden? Sie geben Ihren Benutzernamen an, tippen Ihr Kennwort ein und bestätigen diese Eingabe. Im Hintergrund prüft der Computer nun, ob er Sie kennt. Das ermittelt er anhand der Benutzer-ID, in diesem Fall Ihrem Benutzernamen. Dazu existiert in Windows-Systemen ein sogenanntes Benutzerkonto. Anschließend vergleicht er das für Ihr Benutzerkonto hinterlegte Passwort mit dem eingegebenen (in der Regel vergleicht er die Hashwerte, da das Passwort aus Sicherheitsgründen nicht direkt hinterlegt ist).

Passt alles zusammen, sind Sie *authentifiziert*. Das bedeutet nichts anderes, als dass der Computer Ihnen Ihre Identität glaubt und Sie für diejenige Person hält, für die Sie sich ausgeben. An dieser Stelle kommt immer auch die *Autorisierung* ins Spiel: Durch die Vergabe von Zugriffs- und Systemrechten erhalten Sie nun die Möglichkeit, in einer festgelegten Art auf bestimmte Daten zuzugreifen, z.B. nur lesend (*read-only*) oder lesend oder schreibend. Auch die Verwendung von Programmen und der Zugriff auf die Systemkonfiguration sind von Ihren Rechten abhängig. Ein Administrator darf hier deutlich mehr (im Zweifel alles) als ein nicht-privilegierter Benutzer.

Neben der Autorisierung dient die Authentizität bzw. Authentisierung in bestimmten Situationen auch der *Nicht-Abstreitbarkeit* (engl. *Non-Repudiation*). Geben Sie z.B. über das Internet eine Bestellung auf und behaupten später, dass Sie das gar nicht getan hätten, so streiten Sie die Bestellung ab und der Auftragnehmer hat das Beweisproblem. Gerade bei Geschäftsbeziehungen, die über das Internet laufen, spielt dies eine große Rolle.

Ziel der Nicht-Abstreitbarkeit ist der Nachweis, dass eine Nachricht mit einem bestimmten Inhalt tatsächlich von der Person gekommen ist, die als Absender angegeben ist. Dies wird durch ähnliche Methoden erreicht, wie sie bei der Sicherstellung der Integrität eingesetzt werden.

Schutzmaßnahmen

Eine große Rolle spielen hier Hashwerte als Prüfsummen und ein Konzept namens *digitale Signatur* oder *elektronische Unterschrift*. Durch die digitale Signatur kann eindeutig nachgewiesen werden, dass eine Nachricht von einem bestimmten Absender stammt. Im Zusammenspiel mit der Integritätsprüfung kann auch der Inhalt verifiziert werden, sodass eine Nicht-Abstreitbarkeit erreicht wird. Dadurch werden Geschäftsbeziehungen im Internet glaubwürdig. Gelingt es einem Angreifer, diese digitale Signatur oder die Hashwerte zur Integritätsprüfung zu fälschen, wiegt sich der Empfänger einer Nachricht in falscher Sicherheit. Im Rahmen von Kapitel 5 *Kryptografie und ihre Schwachstellen* nennen wir Ihnen effektive Methoden, Ihre Integrität und Authentizität zu schützen.

1.6.5 Die Quadratur des Kreises

Sind Sie verantwortlich für die IT-Sicherheit, sollten Sie immer die oben genannten Schutzziele im Auge behalten und sich entsprechend schützen.

Bei allem Sicherheitsbewusstsein, das wir bei Ihnen im Laufe dieses Buches verstärken möchten, dürfen Sie allerdings nie das Verhältnis zwischen Sicherheit, Funktionalität und Bedienbarkeit außer Acht lassen.

Je nachdem, wo Sie Schwerpunkte setzen, verlagert sich die Balance Ihrer Computersysteme. Natürlich können Sie die Sicherheit zu 100 % sicherstellen – indem Sie die Systeme abschalten und niemandem zugänglich machen. In diesem Fall würden Funktionalität und Bedienbarkeit auf 0 % reduziert. Und dies ist sicherlich nicht zielführend.

Die anderen Extreme bringen jedoch auch Probleme mit sich: Die Bedienbarkeit zu maximieren, führt in jedem Fall zu vermehrten Sicherheitslücken. So könnten Sie z.B. auf Zugangskontrolle verzichten und jedem Vollzugriff auf alle Systeme und Daten geben. Dass das ebenfalls nicht zum gewünschten Gesamtergebnis führt, müssen wir nicht weiter ausführen.

Das bedeutet letztlich, dass Sie als Sicherheitsbeauftragte(r) manchmal Kompromisse eingehen müssen, die gegen das Sicherheitsziel sprechen. Wenn die Funktionen zu sehr eingeschränkt sind oder sich Ihr System nicht mehr effizient bedienen lässt, haben Sie auch nichts gewonnen. Versuchen Sie, einen gesunden Mittelpunkt im Inneren des Dreiecks zu finden.

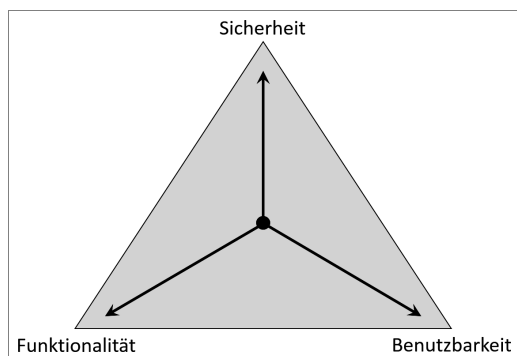


Abb. 1.2: Immer auf das Verhältnis achten

Welche Balance das Optimum in der jeweiligen Umgebung darstellt, lässt sich pauschal nicht beantworten. So wird eine Bank z.B. naturgemäß sehr viel mehr Wert auf Sicherheit legen – zur Not eben auch auf Kosten der Bedienbarkeit (Usability) und Funktionalität. Mittlerweile ist ja das Einloggen in den Online-Bankaccount oft schon ein dreistufiger Authentifizierungsprozess und teilweise recht nervig für den Kunden.

Auf der anderen Seite gibt es Unternehmen, die von der Kreativität und Individualität ihrer Mitarbeiter leben. Hier könnte es notwendig sein, vielen Mitarbeitern weitgehende Rechte bis hin zu Administratorprivilegien einzuräumen, damit diese ihre Jobs optimal ausfüllen können. Dies ist zwar ein Horrorszenario für jeden Security-Beauftragten, aber wenn die Alternative lautet, dass das Unternehmen pleitegeht, weil die Mitarbeiter nicht vernünftig arbeiten können, müssen entsprechende, aus Security-Sicht manchmal schmerzhaft, Kompromisse gefunden werden.

Tipp: Das Prinzip der Least Privileges und das Vier-Augen-Prinzip

Grundsätzlich gilt: Jeder Benutzer erhält so viel Rechte wie nötig und so wenig wie möglich, um seine Tätigkeit ausüben zu können! Führt ein Recht zu einem Sicherheitsproblem, suchen Sie

nach Alternativen: Ist es z.B. möglich, bestimmte, sicherheitskritische Prozesse durch nur einen oder wenige Mitarbeiter ausführen zu lassen, anstatt durch jeden einzelnen Benutzer? Sorgen Sie im Zweifel auch immer für ein Vier-Augen-Prinzip: Ein Mitarbeiter beantragt einen Prozess, ein zweiter genehmigt diesen und der dritte führt ihn schließlich aus. Das reduziert den Missbrauch von privilegierten Funktionen, wie z.B. das Ändern von Firewall-Regeln.

1.7 Systematischer Ablauf eines Hacking-Angriffs

Einer der Haupt-Unterschiede zwischen Scriptkiddies und echten Hackern oder auch Pentestern ist das systematische Vorgehen, das bei den Scriptkiddies fehlt. Ein professioneller Hacking-Angriff umfasst eine Reihe von Phasen, die aufeinander aufbauen. Es gibt verschiedene Ansätze, die leicht voneinander abweichen, aber inhaltlich weitgehend denselben Weg verfolgen. Abbildung 1.3 zeigt eine Übersicht über die einzelnen Etappen, wie sie vom CEH-Curriculum unterschieden werden.

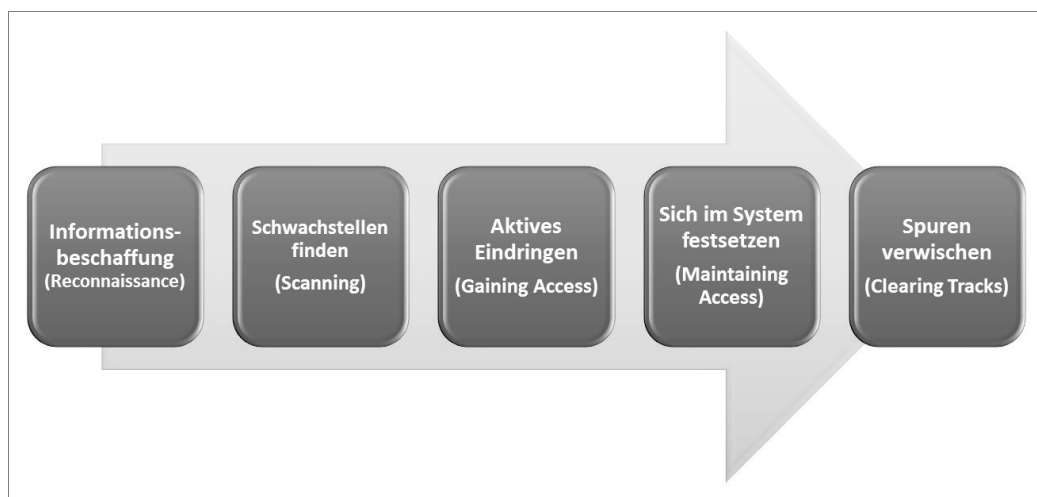


Abb. 1.3: Prozess-Schritte eines Hacking-Angriffs

Hierbei ergibt sich jedoch eine Begriffsüberschneidung, da die zweite Phase, das *Scanning*, in den meisten Quellen zur aktiven *Reconnaissance-Phase* hinzugerechnet wird. An dieser Stelle gibt es diverse Begrifflichkeiten zu unterscheiden. Wir werden das gleich noch etwas genauer erläutern.

Auch wenn die Vorgehensweise von Black Hat Hackern und White Hat Hackern grundsätzlich gleich ist, so sind die Phasen bei einem realen Angriff noch etwas umfangreicher und aggressiver. Schauen wir uns das einmal an.

1.7.1 Phasen eines echten Angriffs

Im Rahmen eines professionellen Hacking-Angriffs versucht der Angreifer, sein Ziel systematisch und nachhaltig zu erreichen. So hat er z.B. nichts gewonnen, wenn er zwar die gesuchten Daten findet und stehlen kann, dabei aber erwischt wird. Daher ist es notwendig, mit Bedacht vorzugehen und möglichst wenig Spuren zu hinterlassen. Zudem kann der Angreifer die Chance nutzen, im

Rahmen eines erfolgreichen Angriffs eine Hintertür einzubauen, die ihm auch zukünftig Zugang zu dem betreffenden System sichert.

Für einen erfolgreichen Angriff wird der Hacker in der Regel eine bestimmte Reihenfolge seiner Handlungen verfolgen, um sich seinem Ziel schrittweise zu nähern und nach erfolgreichem Angriff auch wieder unbemerkt abtauchen zu können. Betrachten wir die einzelnen Schritte einmal genauer:

Informationsbeschaffung (Reconnaissance)

Dies ist der erste Schritt für die Vorbereitung auf einen Angriff. Sammeln Sie möglichst viele Informationen über Ihr Ziel. Je mehr Informationen Sie haben, umso gezielter können die nächsten Schritte gewählt werden. Das spart nicht nur Zeit, sondern erhöht auch die Chance, Schwachstellen zu finden. Wir unterscheiden zwischen zwei Phasen:

- **Passive Discovery:** In dieser Phase versuchen Sie, Informationen über Ihr Ziel (also die Person oder das Unternehmen) zu erlangen, ohne direkt mit ihm in Kontakt zu treten. Dies umfasst z.B. Google-Suchen, Social-Media-Analysen und andere Recherchen über das Ziel, kann aber auch bedeuten, dass Sie das Gebäude des betreffenden Unternehmens beobachten, um die Verhaltensweisen und Gewohnheiten der Mitarbeiter und des Wachpersonals zu erkunden. Passive Discovery umfasst damit auch einen Teil des *Social Engineerings* (grob ausgedrückt ist das alles, was primär mit Menschen statt Computern zu tun hat, genauer wird dieses Thema in Kapitel 20 *Social Engineering* behandelt) sowie das sogenannte *Dumpster Diving*, bei dem der Angreifer versucht, aus dem Müll des Opfers relevante Informationen zu erlangen. Dies kann z.B. erfolgreich sein, wenn wichtige Dokumente nicht sachgerecht entsorgt werden.
- **Active Discovery:** Jetzt werden Sie als Angreifer konkreter und prüfen die Systeme durch aktives »Anklopfen«. Das heißt, Sie treten bereits mit den Systemen des Opfers in Kontakt. In dieser Phase setzen Sie sich erstmalig der Gefahr aus, entdeckt zu werden. Andererseits können Sie aber auch wichtige Informationen zu den Zielsystemen erlangen, die weitere Angriffsvorbereitungen ermöglichen.

Wichtig: Verschiedene Perspektiven unterscheiden!

Der CEH sieht in der Active-Discovery-Phase noch keine Scanning-Aktivitäten, sondern die Verbindungsaufnahme mit dem Ziel auf anderen Ebenen, z.B. einem Telefonanruf beim Help Desk oder in der IT-Abteilung. Wir betrachten daher die Scanning-Phase formal auch von der Reconnaissance-Phase getrennt, sehen aber inhaltlich das Scanning als Bestandteil der Active-Discovery-Phase.

Schwachstellen finden (Scanning)

Somit geht die Active-Discovery-Phase sozusagen fließend in die Scanning-Phase über. In dieser Phase werden die Zielsysteme genau unter die Lupe genommen. Dabei nutzen Sie als Angreifer die Informationen, die Sie im Rahmen des ersten Schrittes der (passiven) Informationsbeschaffung (Reconnaissance) erlangt haben. Hier kommen Netzwerk-Scanner und -Mapper sowie Vulnerability-Scanner zum Einsatz. Tatsächlich erhöht sich der Grad der Aggressivität des Scans gegenüber dem Active Discovery.

In dieser Phase ermittelt der Angreifer die Architektur des Netzwerks, offene Ports und Dienste, die Art der Dienste, Betriebssysteme, Patchstände, scannt auf bekannte Schwachstellen und Sicher-

heitslücken etc. In dieser Phase steigt die Entdeckungsgefahr weiter, da der Angreifer sehr aktiv und teilweise aggressiv mit den Zielsystemen kommuniziert.

Aktives Eindringen (Gaining Access)

Hier geht es richtig los, denn jetzt versuchen Sie, die gefundenen Lücken auszunutzen und sich mittels entsprechender Exploits unerlaubten Zugriff zu verschaffen. Angriffe gibt es in allen möglichen Varianten, wie Webserver-Attacken, SQL-Injection, Session Hijacking, Buffer Overflow etc. Diese werden wir ausführlich vorstellen und natürlich auch praktisch demonstrieren.

Sich im System festsetzen (Maintaining Access)

Hat der Angreifer sich erst einmal Zugang verschafft, versucht er, den Zugriff auszubauen. Er bemüht sich mittels *Privilege Escalation* um noch mehr Rechte und versucht, das System weitestgehend einzunehmen. Mittlerweile hat er nicht nur Zugang zum System, sondern bestenfalls sogar Administrator-Privilegien. Damit gibt sich ein professioneller Angreifer jedoch nicht zufrieden. Denn an dieser Stelle nutzen Black Hats die Gunst der Stunde, weitere Sicherheitslücken zu schaffen und über entsprechende »Backdoors« dafür zu sorgen, dass sie das Opfer-System jederzeit wieder »besuchen« können.

Das kann auch hilfreich sein, sollte die Lücke, durch die der Angreifer hineingekommen ist, geschlossen werden. Jetzt wird Ihnen vermutlich auch klar, warum Sie einem einmal kompromittierten System nicht mehr trauen können: Als Administrator eines einmal kompromittierten Systems werden Sie keine ruhige Nacht mehr haben, mit dem Hintergedanken, dass der Angreifer evtl. weitere Einfallstore und Zugänge installiert hat.

Spuren verwischen (Clearing Tracks)

In den meisten Fällen entstehen bei einem Hacking-Angriff Spuren, die durch Methoden der Computer-Forensik ausgewertet werden können. Ist der Angriff auf den Hacker zurückzuführen, so ist dessen Karriere schnell vorbei.

In dieser Phase geht es also darum, die Spuren seines (unerlaubten) Tuns möglichst nachhaltig und umfangreich zu verwischen. Hierzu werden Logging-Einträge manipuliert oder gelöscht, Rootkits installiert, die sehr tief im Kernel operieren und das System und dessen Wahrnehmung der Ereignisse manipulieren können, sowie Kommunikationsprotokolle und -wege eingesetzt, die eine Nachverfolgung erschweren.

Nicht immer müssen die Angriffe strikt in dieser Reihenfolge ablaufen. So kann es durchaus sein, dass Sie einen Scan auf ein System laufen lassen, während Sie in der Zwischenzeit in ein anderes einbrechen. Auch macht es Sinn, zwischen den einzelnen Schritten seine Spuren immer wieder zu verwischen, obwohl diese Phase generell erst am Ende der Kette steht. Um allerdings den grundlegenden Ablauf zu verstehen und zu verinnerlichen, ist es wichtig, die Phasen und ihre Reihenfolge zu kennen und ständig im Blick zu haben.

1.7.2 Unterschied zum Penetration Testing

Sie haben vielleicht bemerkt, dass die im vorigen Abschnitt vorgestellten Phasen – gerade die letzten beiden – doch recht »dunkel« anmuten. Und auch wenn das beschriebene Vorgehen weitgehend sowohl für White Hats als auch für Black Hats gilt, so ist der Vorgang beim Penetration Testing im Allgemeinen doch noch ein wenig modifiziert. Dies betrifft insbesondere folgende Punkte:

Vorbereitung

Vor einem Penetrationstest wird sehr genau festgelegt, was die Ziele des Audits sind und in welchem Rahmen der Pentester sich bewegt. Es wird die Aggressivität des Tests festgelegt und die Kommunikation zwischen dem Pentester und dem Auftraggeber geklärt.

Der Auftraggeber wird während des Tests in der Regel in Intervallen über den aktuellen Stand aufgeklärt und über einzelne, geplante Schritte hinsichtlich Zeitraum und Umfang informiert. Dies wird ebenfalls in der Vorbereitungsphase geklärt. Das umfasst auch ggf. gesetzliche Regelungen. Wird das Audit im Rahmen einer *Compliance-Prüfung* durchgeführt, so müssen weitere Rahmenbedingungen und formale Anforderungen erfüllt werden, die vorab zu klären sind. »Compliance« bedeutet Regelkonformität und umfasst die Einhaltung von Gesetzen und Richtlinien. Diverse Unternehmen und Organisationen sind bestimmten Gesetzen unterworfen, die eine entsprechende regelmäßige Prüfung erfordern.

Abschluss und Dokumentation

Während ein echter Angreifer zufrieden ist, wenn er das System kompromittiert und seine Ziele (Datendiebstahl, Sabotage etc.) erreicht hat, muss der Pentester den Auftraggeber bestmöglich unterstützen, um die gefundenen Schwachstellen zu erkennen und zu beseitigen. Daher wird ein umfangreicher Bericht über die Sicherheitslücken, Gefährdungen und Risiken erstellt und ein Maßnahmenkatalog erarbeitet, der dem Auftraggeber die mögliche Beseitigung der Schwachstellen aufzeigt.

Dabei wird auch die Vorgehensweise des Pentesters detailliert beschrieben, um dem Auftraggeber darzulegen, wie die Informationsbeschaffung und Ausnutzung der Sicherheitslücken erfolgt ist. Zur Dokumentation eines Penetrationstests existieren diverse Tools und Hilfsmittel, die eine Datenbank-gestützte Auswertung ermöglichen. Auf die Details hierzu gehen wir in Kapitel 32 *Durchführen von Penetrationstests* am Ende des Buches ein.

Was ein Pentester nicht macht

Im Rahmen eines Audits wird ein Pentester in der Regel nicht versuchen, sich im System festzusetzen, um zu einem späteren Zeitpunkt erneut in das System einzubrechen. Andererseits ist es natürlich durchaus sinnvoll, zu testen, wie weit der Angreifer kommen würde, um *Backdoors* und andere Schwachstellen zu platzieren. Diese werden jedoch im Rahmen eines Audits in der Regel nicht installiert, um sie später tatsächlich zu nutzen – es bleibt meistens beim »Proof-of-Concept«, also beim Ausloten der Möglichkeiten.

Darüber hinaus wird ein Pentester in der Regel auch keine aggressiven Techniken einsetzen, um seine Spuren zu verwischen. Dies erfordert eine Manipulation diverser wichtiger Subsysteme von Produktivsystemen, einschließlich des Einsatzes von Rootkits, die es ermöglichen, auf Kernel-Ebene elementare Prozesse und Dateien zu manipulieren und zu verstecken.

Dahinter steckt nicht zuletzt die Philosophie, dass die Systeme des Auftraggebers getestet und anschließend *gehärtet* (also sicherer gemacht) werden sollen, nicht jedoch als Spielwiese eines Hackers dienen sollen, um zu schauen, was alles geht. Das gezielte Schwächen eines Produktiv-Systems führt unter Umständen zur Notwendigkeit einer Neuinstallation und ist ein »No-Go« für einen Pentester.

Tipp: Bleiben Sie neugierig und testen Sie Ihre Grenzen aus!

Damit wir uns nicht falsch verstehen: Wir fordern Sie geradezu auf, an die Grenze Ihrer Fähigkeiten zu gehen! Innerhalb Ihres Labornetzes sollten Sie alles, was irgendwie möglich erscheint,

umsetzen und ausprobieren – hier sind Ihnen keine Grenzen gesetzt – virtuelle Maschinen und Snapshots machen es möglich.

Stellen Sie jedoch sicher, dass die von Ihnen angegriffenen Systeme vollständig unter Ihrer eigenen Kontrolle sind und keinerlei Produktivzwecken dienen! In Ihrem abgeschotteten Labor können Sie so viel herumexperimentieren, wie Sie wollen. Aber halten Sie strikt die Regeln ein, wenn Sie ein anderes Netzwerk oder Computersystem im Rahmen eines beauftragten Penetrationstests hacken.

Grundsätzlich gibt es auch spezielle Szenarien, in denen ein Pentester aggressiver vorgeht und bestimmte Black-Hat-Methoden anwendet, wie beispielsweise die Installation einer Backdoor. Dies hängt immer von der Zielstellung bzw. Auftragsformulierung ab. Unter dem Strich muss dies jedoch abgesprochen sein und dem Gesamtziel der Verbesserung der IT-Sicherheit dienen.

1.8 Praktische Hacking-Beispiele

In diesem letzten Abschnitt des Kapitels möchten wir Ihnen noch drei erfolgreiche Hacking-Angriffe vorstellen, um gleich einmal etwas »Praxis« einzubringen und Ihnen eine Vorstellung von »Real-World-Hacks« zu geben.

1.8.1 Angriff auf den Deutschen Bundestag

Am 13. April 2015 wurde ein Angriff auf das Netzwerk des Bundestages bekannt, bei dem diverse, teilweise als *Top Secret* eingestufte, Dokumente gestohlen wurden. Offensichtlich haben sich die Hacker Zugang zu einem Großteil der Systeme des Bundestages verschaffen können, sodass zum einen nicht im Detail nachvollziehbar ist, welche Informationen entwendet und welche Systeme kompromittiert wurden. Zum anderen wurde es dadurch notwendig, einen erheblichen Teil der IT-Infrastruktur neu aufzusetzen, um wieder Vertrauen in die Systeme haben zu können.

Nach den Analysen ist zunächst ein einzelner Computer eines Abgeordneten durch eine E-Mail mit entsprechendem Malware-Anhang oder einem *Drive-by-Download* (ein Schadcode wird automatisch beim Besuch einer bestimmten Website unbemerkt im Hintergrund heruntergeladen) infiziert worden. So hatten die Angreifer vermutlich eine *Backdoor* (also eine Hintertür im System) installiert, über die sie Zugang zum Opfer-System erlangten.

Von dort aus gelang es den Angreifern mittels gängiger Open-Source-Software (namentlich *mimikatz*, siehe Kapitel 10 *Password Hacking*), Zugriff auf Administrator-Accounts zu erlangen, die ihnen wiederum Zugang zu diversen Systemen des Netzwerks ermöglichten und dazu führten, dass sich die Angreifer frei im Netzwerk des Bundestages bewegen konnten.

Interessant hierbei ist, dass bis zu diesem Zeitpunkt niemand wirklich reagierte: Obwohl sich einige Systeme merkwürdig verhielten, nahm man die Situation noch nicht so richtig ernst. Erst als ausländische Geheimdienste mitteilten, dass ein derartiger Angriffsplan entdeckt wurde, sind die entsprechenden Stellen, unter anderem das *Bundesamt für Sicherheit in der Informationstechnik* (BSI) involviert worden, um die Sachverhalte aufzuklären.

Das Verblüffende hierbei ist, dass die Angreifer bereits bekannte Schwachstellen und Hacking-Tools eingesetzt haben. Es muss sich also keineswegs um versierte Hacker gehandelt haben – stattdessen wäre es erschreckenderweise auch denkbar, dass hier Scriptkiddies (zugegebenermaßen mit deutlich erweiterten Kenntnissen) am Werk waren!

Unter dem Strich bleibt die Erkenntnis, dass das Netzwerk des Bundestages zum einen unzureichend geschützt war und zum anderen das Sicherheitsbewusstsein der Administratoren ganz offen-

sichtlich nicht ausreichte, um die (durchaus vorhandenen) Symptome des Angriffs rechtzeitig zu erkennen und entsprechend zu handeln. Aufgrund dieser Umstände war es sogar mit relativ einfachen Mitteln und Open-Source-Standard-Tools möglich, derart tief in das Netzwerk des Bundestages einzudringen und sich dort festzusetzen.

1.8.2 Stuxnet – der genialste Wurm aller Zeiten

Im krassen Gegensatz zum Angriff auf den Bundestag wurde 2010 ein Computerwurm entdeckt, der als *Stuxnet* bekannt wurde. Es handelt sich um den höchstentwickelten Wurm, der jemals gefunden wurde. Er nutzt eine Vielzahl von Schwachstellen und kann sogar, wie ein normales Programm, automatisch über das Internet aktualisiert werden.

Stuxnet wurde speziell für den Angriff auf *Simatic S7* entwickelt. Dabei handelt es sich um ein Steuerungssystem der Firma Siemens, das vielfach in verschiedenen Industrieanlagen, wie z.B. Wasserwerken, Pipelines oder aber auch Urananreicherungsanlagen eingesetzt wird.

Letztere schienen auch das Ziel von Stuxnet zu sein, da zunächst der Iran den größten Anteil an infizierten Computern besaß und die Anlagen des iranischen Atomprogramms von Störungen betroffen waren. Durch die Störung der Leittechnik dieser Anlagen sollte wohl die Entwicklung des Atomprogramms gestört und verzögert werden.

Die Entwickler und Auftraggeber von Stuxnet sind bis heute nicht bekannt – selbstverständlich gibt es diverse Gerüchte und Indizien, die an dieser Stelle aber nicht von Belang sind. Entscheidend ist, dass hier kein einzelner Hobbyprogrammierer oder Scriptkiddie am Werk war, sondern eine hochversierte Gruppe professioneller Entwickler. Die Komplexität von Stuxnet legt die Vermutung nahe, dass hier hochspezialisierte Experten an der Arbeit waren und die Entwicklung des Wurms mehrere Monate professioneller Projektarbeit erforderte.

Hinweis: Zusatzmaterial zum Buch online

Mehr Informationen über Stuxnet haben wir in einem Dokument zusammengefasst und zum Download unter www.hacking-akademie.de/buch/member bereitgestellt. Bitte nutzen Sie das Passwort `h4ckm3mber` für den exklusiven Zugang zum Mitglieder-Bereich unserer Leser.

1.8.3 Angriff auf heise.de mittels Emotet

Auch Malware entwickelt sich weiter und ein neuer Meilenstein in der Evolution war *Emotet*. Dabei handelt es sich um einen sogenannten Banking-Trojaner. Derartige Schadsoftware ist darauf spezialisiert, Zugangsdaten von Online-Banking-Diensten auszuspähen. Emotet ist jedoch erheblich vielseitiger und leistungsfähiger als die meisten derartigen Schadprogramme und wird zudem aktiv weiterentwickelt.

Seit 2018 ist Emotet in der Lage, auch lokale E-Mails auszulesen und somit selbst Mails zu generieren, die scheinbar von bekannten Absendern kommen, mit denen das Opfer kürzlich bereits in Kontakt stand. Durch glaubwürdige Inhalte wird der Benutzer dazu verführt, schädliche Dateianhänge zu öffnen oder auf Links zu klicken, die zu infizierten Servern führen, wodurch sogenannte *Drive-by-Downloads* initiiert werden. Diese automatischen Downloads nutzen Browserlücken aus und platzieren Schadcode auf dem Computer des Opfers.

Im Mai 2019 wurde das bekannte Online-Magazin heise.de Opfer von Emotet. Es handelte sich um einen ausgeklügelten, mehrstufigen Angriff, der von heise vorbildlich und transparent aufgearbeitet wurde. Die detaillierten Untersuchungsergebnisse wurden veröffentlicht. Sie können unter

www.heise.de/ct/artikel/Trojaner-Befall-Emotet-bei-Heise-4437807.html den gesamten Vorfall in allen Details nachlesen.

1.9 Zusammenfassung und Prüfungstipps

Werfen wir einen kurzen Blick zurück: Was haben Sie gelernt, wo stehen Sie und wie geht es weiter?

1.9.1 Zusammenfassung und Weiterführendes

Sie haben in diesem Kapitel gelernt, was es mit dem Begriff »hacking« bzw. »Hacker« auf sich hat, und haben festgestellt, dass wir hier durchaus genau unterscheiden müssen, z.B. zwischen *Script-kiddie*, *White Hat*, *Grey Hat* und *Black Hat* bzw. dem *Cracker*. Weiterhin haben wir Motive und Ziele von Hacking-Angriffen beleuchtet.

Ein ganz elementares Konzept, das Sie sich unbedingt zu Eigen machen sollten, ist das »Ethical Hacking«. Hierbei geht es darum, als *White Hat Hacker* die Kunst des Hackings einzusetzen, um die Sicherheit von Computersystemen und -netzwerken zu verbessern. Wenn Sie die Zukunft Ihrer Karriere im Ethical Hacking sehen, dann sollten Sie sich überlegen, die Prüfung zum *Certified Ethical Hacker* zu absolvieren.

Es ist wichtig, beide Seiten zu berücksichtigen. Daher haben wir vorübergehend einen Perspektiv-Wechsel vorgenommen und betrachtet, welche Schutzziele es gibt und wie sie von den IT-Sicherheitsbeauftragten verfolgt werden. Der Abkürzung *CIA* stehen die englischen Begriffe *Confidentiality* (Vertraulichkeit), *Integrity* (Integrität) und *Availability* (Verfügbarkeit) gegenüber. Dazu kommt in manchen Betrachtungen noch die *Authenticity* (Authentizität) bzw. die *Non Repudiation* (Nichtabstreitbarkeit). Beides wird aber häufig auch unter der Integrität zusammengefasst. Die Herausforderung für einen IT-Sicherheitsbeauftragten ist die Sicherstellung der Schutzziele einerseits, ohne andererseits die Benutzerfreundlichkeit und die Funktionalität zu stark einzuschränken – sonst heißt es am Ende: »Operation gelungen, Patient tot!«

Wird das *White Hat Hacking* im Rahmen eines abgesprochenen Audits durchgeführt, so nennt sich dieser Prozess *Penetrationstest*, oder in der englischen Form: *Penetration Test* bzw. kurz: *Pentest*. Dabei werden die Computersysteme und/oder das Netzwerk des Auftraggebers nach detaillierter Absprache systematisch auf Schwachstellen untersucht. Hierzu bedient sich der Pentester professioneller Hacking-Methoden.

In diesem Zusammenhang haben Sie die Phasen eines Hacking-Angriffs kennengelernt, die aus dem *Ausspähen* (Reconnaissance), dem *Finden von Schwachstellen* (Scanning), dem *aktiven Eindringen* (Gaining Access), dem *Festsetzen im Opfer-System* (Maintaining Access) sowie der *Verwischung der Einbruchsspuren* (Clearing Tracks) besteht. Im Rahmen eines Pentests werden einige der Phasen angepasst, da es hier insbesondere um das Aufzeigen und Dokumentieren von Schwachstellen geht.

1.9.2 CEH-Prüfungstipps

In diesem ersten Kapitel sind schon einige wichtige Begriffe und Konzepte enthalten, die in der Prüfung abgefragt werden können. Hierzu zählen die unterschiedlichen Hackertypen, die Schutzziele und die Phasen eines Hacking-Angriffs. Stellen Sie sicher, dass Sie Hacking-Aktivitäten den einzelnen Phasen zuordnen können und dass Sie verstanden haben, welche Schutzziele durch bestimmte Maßnahmen sichergestellt bzw. bedroht werden. Letzteres werden Sie im Laufe dieses Buches immer wieder gegenüberstellen können.

1.9.3 Fragen zur CEH-Prüfungsvorbereitung

Mit den nachfolgenden Fragen können Sie Ihr Wissen überprüfen. Die Fragestellungen sind teilweise ähnlich zum CEH-Examen und können daher gut zur ergänzenden Vorbereitung auf das Examen genutzt werden. Die Lösungen zu den Fragen finden Sie in Anhang A.

1. Welcher Hacker-Typ hat beschränkte oder kaum Kenntnisse im Security-Bereich und weiß lediglich, wie einige einschlägige Hacking-Tools verwendet werden?
 - a) Black Hat Hacker
 - b) White Hat Hacker
 - c) Scriptkiddie
 - d) Grey Hat Hacker
 - e) Cracker

2. Welche der im Folgenden genannten Phasen ist die wichtigste Phase im Ethical Hacking, die häufig die längste Zeitspanne in Anspruch nimmt?
 - a) Gaining Access
 - b) Network Mapping
 - c) Privilege Escalation
 - d) Footprinting
 - e) Clearing Tracks

3. Ein CEH-zertifizierter Ethical Hacker wird von einer Freundin angesprochen. Sie erklärt ihm, dass sie befürchtet, ihr Ehemann würde sie betrügen. Sie bietet dem Ethical Hacker eine Bezahlung an, damit er in den E-Mail-Account des Freundes einbricht, um Beweise zu finden. Was wird er ihr antworten?
 - a) Er lehnt ab, da der Account nicht der Freundin gehört.
 - b) Er sagt zu, da der Ehemann unethisch handelt und die Freundin Hilfe benötigt.
 - c) Er sagt zu, lehnt aber die Bezahlung ab, da es sich um einen Freundschaftsdienst handelt.
 - d) Er lehnt ab und erklärt der Freundin, welcher Gefahr sie ihn damit aussetzt.

4. Die Sicherheitsrichtlinie (Security Policy) definiert die Grundsätze der IT-Security in der Organisation. Für einige Bereiche gibt es ggf. Sub-Policys, wie z.B. Computer-Sicherheitsrichtlinie, Netzwerk-Sicherheitsrichtlinie, Remote-Access-Richtlinie etc. Welche drei der im Folgenden genannten Ziele sollen damit sichergestellt werden?
 - a) Availability, Non-repudiation, Confidentiality
 - b) Authenticity, Integrity, Non-repudiation
 - c) Confidentiality, Integrity, Availability
 - d) Authenticity, Confidentiality, Integrity

5. Welcher Phase eines Hacking-Angriffs kann die Installation eines Rootkits zugerechnet werden?
 - a) Reconnaissance
 - b) Scanning
 - c) Gaining Access
 - d) Maintaining Access
 - e) Clearing Tracks

Stichwortverzeichnis

6LoWPAN 1111
802.1x 731

A

Access Control List (ACL) 709
Active Directory (AD) 312
Active Discovery 221, 256
Acunetix 856
Address Resolution Protocol (ARP) 259, 637
Address Space Layout Randomization (ASLR) 1007
Ad-hoc-Netzwerk (WLAN) 1020
ADS 489
Advanced Message Queuing Protocol (AMQT) 1112
AdwCleaner 522
airbase-ng 1059, 1060
aircrack-ng 1029, 1041
AirDroid 1075
aireplay-ng 1038, 1040
Airedon 1061
airodump-ng 1033, 1040, 1043, 1047
Ajax 842
Alternate Data Stream 489
Amplifying Attack 800
Android 1068
Android Debug Bridge (ADB) 1081
Android x86 1076
Angler 472
Angriffsphasen 56
Anonymizer 137
Anonymous 44
Antivirus-System (AV) 473
Anydesk 1075
Any Run 507
apache2 468
Apache-Webserver 843
App 1069
ARP-Cache-Poisoning 637
ARP-Inspection 663
ARP-Spoofing 637
 arpspoof 644
ASP.net 842
Asymmetrische Verschlüsselung 175
 Authentizitätsprüfung 178
 Diffie-Hellman-Schlüsselaustausch 179
 Digital Signature Algorithm (DSA) 181
 Elgamal 180
 Private Key 176

 Public Key 176
 Public-Key-Authentifizierung 178
 Rivest Shamir Adleman (RSA) 180
 Schlüsselaustausch 176
auditpol 563, 568
Audit Policies (Windows) 562
Ausführen-Recht (x) 105
Autoruns 525
Autostart-Eintrag 524
AV-Signatur 473
AWS 1147
Azure (Microsoft) 1148

B

Backdoor 419, 453
BackTrack 71
Bad Character 994
Baseband-Hack 1072
Bash Bunny 779
Beacon Frame (WLAN) 1024
Best(er) Keylogger 492
Bettercap 661
Bildschirm Auflösung 102
Bind-Shell 424
Black-Box-Test 1178
Black Hat 41
Blackhole Exploit Kit 472
Blind Hijacking 675
BlueBorne 1123
Bluebugging 1074
Bluejacking 1074
Bluesnarfing 1074
BlueStacks 1076
Blue Teaming 1179
Bluetooth Low Energy (BLE) 1110
Boot-Sektor-Virus 458
Botnet 454
Botnetz 807
Bricking 806
Bring Your Own Device (BYOD) 1095
Brute-Force-Angriff 392
BSS (Basic Service Set) 1020
BSSID (Basic Service Set Identifier) 1024
btmtp 580
Buffer Overflow (Pufferüberlauf) 977
Bug-Bounty-Programm 874
BulkFileChanger 573

bully (WPS-Cracking) 1048
 Burp Suite 681
 Proxy 683
 Sequencer 686

C

c99 (Webshell) 969
 C/C++ (Buffer Overflow) 979
 Cain & Abel 406
 Capsa 518
 Captive Portal (WLAN) 1055
 Capture 594
 Cavity Virus 459
 CCleaner 158, 455, 522, 576
 CEHv12-Prüfung 48
 CeWL 400
 CGI 842
 ChameleonMini 788
 chmod 106
 chntpw 375
 CIFS 297
 Clear_Event_Viewer_Logs.bat 571
 Cloud 1141
 CloudGoat 1170
 Clustering 53
 cmd.exe 419
 Colasoft Packet Builder 286
 Command-Injection 961
 Community Cloud 1145
 Community-String 306
 Companion-Virus 459
 Compliance 1177
 Computervirus 452, 453
 Computerwurm 453, 459
 Config-Register (Cisco) 379
 Constrained Application Protocol (CoAP) 1112
 Contentfilter 711
 Contiki 1108
 Cookies 837
 Covert Channel 486
 Crazyradio PA 786
 Credentialed Scan 353
 Credential Stuffing 911, 918
 Cross-Site-Scripting (XSS) 698, 892
 Crunch 398, 1036
 Crypter 507
 Cryptojacking 1167
 Crypto-Mining 1167
 CrypTool 166
 CSRF (Cross-Site-Request-Forgery) 898
 CSS 842
 Cuckoo 537
 CurrPorts 516
 Custom-Recovery 1082
 Custom-ROM (Android) 1079
 CVE 336
 Cyber-Terrorist 42

D

Dander Spritz 569
 Darknet 147
 Data Execution Prevention (DEP) 1008
 Datei
 anzeigen 112
 finden 114
 Dateimanager 99
 Dateisignaturverifizierung 530
 Datei-Virus 458
 Deauthentication Attack (WLAN) 788, 1037
 Debugger 980
 Decompiler 504
 Deep Web 147
 Defacing 42
 Default-Passwörter 370
 Denial-of-Service-Angriff (DoS-Angriff) 796
 DHCP-Snooping 663
 DHCP-Spoofing 641
 Dictionary-Angriffe 393
 Dienst
 prüfen 526
 verwalten 117
 Diffie-Hellman-Schlüsselaustausch *siehe* Asymmetrische Verschlüsselung
 Digispark Development Board 781
 Digitale Signatur 54
 DirBuster 853
 Directory-Traversal-Angriff 845, 965
 Disassembler 504
 diskpart 373
 DistCC (Schwachstelle) 430
 Distributed-Denial-of-Service-Angriff (DDoS-Attacke) 515, 796
 Distributed-Reflected-DoS-Angriff (DRDoS) 806
 DMZ 713
 DNS-Amplification-Angriff 806
 DNS-Cache-Poisoning 638
 DNS-Footprinting 233
 DNS-Hijacking 639
 DNS-Injection 639
 DNS over TLS 663
 DNSQuerySniffer 519
 dnsrecon 323
 DNSSEC 663
 dnsspoof 647
 DNS-Spoofing 638
 Domain Name System (DNS) 322, 638
 DOM-Interface 697
 Drive-by-Download 455, 760
 DriverView 528
 Dropbox 1142
 Dropper 452
 dsniff (Tool) 642, 650
 Dumpster Diving 367
 DVWA 877

E

Eavesdropping 628
 EAX, EBX, ECX und EDX (Stack Register) 980
 EBP (Stack Pointer) 980
 EICAR 510
 EIP (Stack Pointer) 980
 Elektronische Unterschrift 54
 E-Mail-Footprinting 237
 Encoder 475
 Encryption Code 459
 Entropie 689
 Entry Point 962
 enum4linux 301
 Enumeration 218, 295
 NetBIOS 296
 SMB 297
 Ereignisanzeige 562
 ESS (Extended Service Set) 1021
 ESSID (Extended Service Set Identifier) 1024
 Etcher 790
 Ethereum 595
 Ethical Hacking 1176
 Ettercap 651, 1057, 1060
 Evasion (IDS/IPS) 726
 eventlogedit 569
 eventvwr.exe 562
 evilginx2 768
 Evil Twin (WLAN) 1060
 Exploit 336, 354, 435
 Exploit-Database 229
 Exploit Kit 472
 Exposure *siehe* Vulnerability
 Extensible Markup Language (XML) 839

F

False Positives 354
 Fastboot 1081, 1082
 Federation Services 1167
 FGDump 386
 Fingerabdruck-Scan 366
 Firewalking 716
 Firewall 707
 Application Layer Gateway 711
 Contentfilter 711
 Deep Packet Inspection 712
 Failover/Cluster 715
 iptables 710
 Netzwerk-Firewall 708
 Paketfilter-Firewall 709
 Perimeterschutz 709
 Personal-Firewall 708
 Proxy-System 711
 Stateful Inspection 710
 UTM-Lösung 713
 FISMA 1182
 Footprinting 218
 FoxyProxy 136
 FQDN 234

Fragmentation-Angriff 803
 Fragmentierung 729
 Framegrabber 783
 Freenet-Netzwerk 153
 fsutil 573
 FTP-Zugangsdaten ermitteln 610
 Fuzzing 986

G

Gerätetreiber prüfen 528
 Gesichtsscanner 366
 GHDB *siehe* Google Hacking Database
 Golden Ticket 1168
 Google Cloud Platform 1148
 Googledork 228
 Google-Hacking 227
 Google Hacking Database 229
 gpedit.msc 563
 Gqrx 1120
 Greenshot 1192
 Grey-Box-Test 1178
 Grey Hat 42
 Gruppenrichtlinienverwaltungs-Editor 563
 G-Zapper 159

H

Hacker-Paragraf 46
 HackRF One 1119
 Hacktivist 42
 Handler 696
 Hard Brick 1079
 Hash-Algorithmen 181
 Bcrypt und Scrypt 187
 Integritätsprüfung 182
 Kryptologische Hashfunktionen 185
 Message Digest 5 (MD5) 186
 Passwort-Hashfunktionen 185
 PBKDF2 186
 Prüfsummen 185
 Secure Hash Algorithm (SHA) 186
 Hash Injection Attack 396
 Hash Suite 404
 Hashwert 52
 Haveibeenpwned (Website) 394
 Heap-Buffer-Overflow-Angriff 1005
 Heap Spraying (Heap Overflow) 1006
 Heartbleed-Angriff 203
 High Availability 53
 HijackThis 523
 HIPAA 1181
 Honeypot 733
 hosts (Datei) 518, 523, 640
 Hotspot 1017
 hping3 284, 815
 HTML 842
 HTTP 834
 CONNECT 837
 DELETE 837

- GET 836
- HEAD 837
- Host-Header-Wert 835, 844
- PATCH 837
- POST 836
- PUT 837
- User-Agent 835
- HTTPprint 850
- HTTrack 240, 855
- Hub 594, 632
- Hub-Modus (Switch) 633
- Human Hacker 748
- Hybrid Cloud 1146
- Hydra 409
- Hyperion 478
- Hypertext Transfer Protocol (HTTP) 834
- Hyper-V 67

I

- IBSS (Independent Basic Service Set) 1020
- ICMP 260, 637
- ICMP-Flood-Angriff 798
- ICMP-Tunneling 486
- Identity and Access Management (IAM) 1166
- Identity Services Engine 731
- IDOR (Insecure Direct Object References) 880
- IDS (Intrusion-Detection-System)
 - Hostbasiertes IDS (HIDS) 719
 - Netzwerkbasierendes IDS (NIDS) 719
- IEEE 802.11 1019
- IEEE 802.15.4 1111
- IIS 845
- Immunity Debugger 984
- IMSI-Catcher 1074
- Informationsbeschaffung 57
- Infrared Data Association (IrDA) 1110
- Infrastructure as a Service (IaaS) 1143
- Injection-Angriff 925
- Internes Netzwerk 88
- Internet Information Services (IIS) 845
- Internet of Everything 1107
- Internet of Things (IoT) 1105
- Internet Protocol (IPv4) 259
- Intrusion-Detection-System (IDS) 532
- iOS (Apple) 1068
- IPS (Intrusion-Prevention-System) 720
- IPsec 198
 - Authentication Header (AH) 198
 - Encapsulation Security Payload (ESP) 198
 - Internet Key Exchange (IKE) 198
- Iris-Scan 366
- ISO/IEC 27001 und 27002 1182

J

- Jailbreak (iOS) 1084
- Janus-Angriff 630
- Java 842

- Java (Buffer Overflow) 979
- JavaScript 842
- JavaScript Object Notation (JSON) 840
- Jobsuchmaschine 226
- JOESandbox 507
- John the Ripper 401, 404
- JQuery 909
- JSON 840
- Juice Shop (OWASP) 872
- JV16 Powertools 522
- JXplorer 314

K

- Kali Linux 71
 - Netzwerk-Konfiguration 119
 - Systemsprache ändern (Xfce) 78
 - Tastatur-Layout (Xfce) 77
 - Update 80
- Kali Linux - Einstellungen 101
- KARMA-Attacke 788
- Kazam 1191
- KDE 94
- Kerberos 312, 382
- Key Distribution Center 383
- Keylogger 454, 491
- Keystroke-Injection 776
- KillerBee 1124
- Klick Fraud 808
- Kontextmenü 96
- Krypto-Algorithmen 164
- Kryptoanalyse 163, 201, 202
 - Brute Force 202
 - Chosen Ciphertext 203
 - Chosen Plaintext 203
 - Dictionary Attack 201
 - Frequency Analysis 203
 - Known Ciphertext 203
 - Known Plaintext 203
 - Man-in-the-Middle-Angriff (MITM) 202
 - Probable Plaintext 203
 - Rubberhose Attack 203
 - Seitenkanal-Angriff (Side-Channel Attack) 202
 - Timing Attack 202
 - Trickery And Deceit 203
 - Wörterbuchangriff 201
- Kryptografie
 - Algorithmus 165
 - Blockchiffre 168
 - Cäsar-Chiffre 168
 - Chiffre 168
 - digitale Signaturen 187
 - Geheimtext 165
 - Klartext 165
 - Poodle-Angriff 205
 - Public Key Cryptography Standards (PKCS) 187
 - Schlüssel 165
 - Stromchiffre 168
 - symmetrische Verschlüsselung 167

VeraCrypt 172
 Vertraulichkeit 167
 Kryptosystem 164
 Kryptotrojaner 206

L

L0phtcrack 402
 Laborumgebung 70
 LAMP 845
 Lan Manager (LM) 381
 LAN Turtle 785
 Lawful interception 628
 LDAP 312
 Common Name 312
 Distinguished Name 312
 Organisationseinheit 312
 LDAP Admin 316
 libpcap 595
 Light Fidelity (Li-Fi) 1111
 Lightweight-Access Point (LAP) 1022
 LimeSDR 786
 Linset 1061
 Linux-Befehle 102
 Linux-Rechtesystem 104
 Listener 423, 426, 434
 Loadbalancing 54
 Local File Inclusion (LFI) 880, 969
 Locky 208
 Logging 561
 Lokale Sicherheitsrichtlinie 563
 Long Range Wide Area Network (LoRaWAN) 1111
 Low Orbit Ion Cannon (LOIC) 821
 LSASS 386

M

MAC-Adresstabelle 633
 macchanger 1050
 MAC-Flooding 633
 macof 647
 Magisk 1084
 Makrovirus 458
 Maltego 245
 Malware 452
 Malware-Analyse 503
 Management-Report 1192
 Man-in-the-Browser-Angriff (MIB/MITB) 696
 Man-in-the-Cloud (MITC) 1161
 Man-in-the-Middle (MITM) 629
 Man-in-the-Mobile 1072
 Man-Pages 116
 Mausezahn 286
 Maximum Transmission Unit (MTU) 729
 mdk3 1036, 1039
 Medusa 407
 Mesh-Netzwerk (WLAN) 1022
 Metagoofil 240
 Metasploit 278

Exploit für vsftpd 343
 Module 280
 Nmap in Metasploit nutzen 282
 Webscanning 855
 WMAP 855
 Workspaces 280
 Metasploitable 85, 278
 Meterpreter 435, 575
 Microdot 545
 Microsoft 365 1148
 Mimikatz 443
 Mirai 811, 1120
 Mobile Device Management (MDM) 1097
 Mobile Proxy-Tools 156
 CyberGhost 157
 Onion Browser 157
 Orbot 157
 ProxyDroid 157
 Psiphon 157
 Mona (Immunity Debugger) 997
 Most Recently Used (MRU) 571
 MouseJack-Angriff 787
 Mouse Jiggler 783
 MP3Stego 556
 MQ Telemetry Transport (MQTT) 1112
 msconfig (Autostart) 524
 msfconsole 468, 1091
 msfvenom 440, 468, 994, 1001, 1091
 Multihandler 468
 Multipartite-Virus 458
 Mutillidae II 875

N

nasm_shell.rb 998
 Nbtscan 298
 nbtstat 299
 Ncat 287, 420
 Ncrack 410
 Near-Field Communication (NFC) 1111
 Nessus 344, 856
 net-Befehle 300
 NetBEUI 297
 NetBIOS 296
 NetBIOS Enumerator 302
 Netcat 287, 420
 Netcraft 222
 Netstat 516
 net user 375
 Network Access Control 731
 Network Address Translation 131
 Netzwerkbrücke 88
 Netzwerkschnittstelle konfigurieren 121
 Netzwerk-Sniffer 593
 Neutrino 472
 Nexpose 350
 Nikto2 355, 857
 NIST 170, 338
 Nmap 263, 298

- Firewall/IDS Evasion 273
- Half-Open-Scan 267
- Host Discovery 264
- IPv6-Netzwerke scannen 288
- NSE 275
- OS Detection 273
- Ping-Scan 265
- Ports festlegen 269
- Reports 274
- Service Identification 272
- SYN-Stealth-Scan 267
- TCP-ACK-Scan 270
- TCP-Connect-Scan 268
- TCP-IDLE-Scan 271
- TCP NULL-, FIN- und Xmas-Scan 271
- TCP-SYN-Scan 267
- UDP-Scan 268
- Vulnerability-Scanning 341
- Webscanning 855
- Zenmap 277
- Noise Jamming 1037
- NOP-Byte 1003
- Notepad++ 1192
- Npcap 595
- nslookup 323, 963
- NTLM 382
- NTP 320
- ntpd 321
- ntpq 321
- ntptrace 320
- Null-Session 303
- O**
- Obfuscater 507
- Obfuscating 477
- onesixtyone 310
- OpenPuff 556
- OpenSSL 201
- OpenStego 550
- OpenVAS 349
- OSINT 218
- OSI-Referenzmodell 257
- OSSTMM 1185
- OUI (MAC-Adresse) 599
- OWASP 871, 1187
- OWASP Top 10 873, 879
- P**
- Packet Squirrel 784
- Pacu 1171
- PAM 388
- Passive Discovery 217
- Pass the Hash (PTH) 396
- passwd (Datei) 388
- Password Guessing 368
- Passwort-Richtlinie 369
- PATH-Variable 418
- pattern_create.rb 990
- pattern_offset.rb 992
- Payload 435, 452
 - staged 436
 - unstaged 436
- PCI DSS 1181
- Peer-to-Peer-Netzwerk 147
- Penetrationstest 1176
- Penetrationstester 41, 42
- Penetration Testing Execution Standard (PTES) 1187
- Pepper (Passwort-Hashes) 391
- Perimeter-Schutz 534
- Permanenter DoS-Angriff (PDoS) 805
- Personen-Suchmaschine 226
- pestudio 506
- Petya 207
- Pfadangabe 111
- Pharming 758
- Phishing 752, 758
- Phlashing 805
- PHP 842
- Ping 637
- Ping of Death 799
- Pivoting 1166
- Platform as a Service (PaaS) 1143
- Pluggable Authentication Modules 388
- Polymorphic Code 458
- Post-Exploitation 417, 430
- Potential Unwanted Application (PUA) 522
- Potential Unwanted Program (PUP) 522, 530
- Powershell 418
- Printer Exploitation Toolkit (PRET) 1126
- Private Cloud 1145
- Privilegien-Eskalation 417
- Process Explorer 513, 525
- Process Monitor 514
- Programmausführung abbrechen 113
- Promiscuous Mode 90, 594, 598
- Prompt 103
- Proxifier 146
- Proxmark 3 788
- Proxychains 135, 146
- Proxys 131
 - Arten 132
- Public Cloud 1144
- Public-Key-Infrastruktur (PKI) 189
 - Certificate Authority 190
 - Digitale Zertifikate 190
 - OCSP 196
 - Zertifikatsspeicher 192
 - Zertifikatssperlisten und OCSP 195
- Puffer (Buffer Overflow) 979
- PuTTY 140, 467, 521
- PWDump 386
- R**
- Radio-Frequency Identification (RFID) 1111
- Rainbow-Tables 391, 395

- Ransomware 206, 454
 - Raspberry Pi 788
 - reaver (WPS-Cracking) 1048
 - Reconnaissance 57, 218
 - Recon-ng 241
 - Red Teaming 1178
 - REG.exe 573
 - RegCleaner 522
 - regedit.exe 521
 - Register (Stack) 980
 - Registrierungsdatenbank (Windows) 520
 - Registrierungs-Editor 521
 - Registry 520
 - RegScanner 521
 - Regshot 521
 - Remote File Inclusion (RFI) 970
 - Remote Scan 353
 - Report
 - Management- 1192
 - technischer 1192
 - Rescue-Disk 508
 - REST-API 841
 - Retina-Scan 366
 - Retire.js 909
 - Reverse Engineering 503
 - Reverse Proxy 852
 - Reverse-Shell 426
 - RFCrack 1120
 - Rijndael *siehe* Symmetrische Algorithmen
 - RIoT 1108
 - Risk-Assessment 351
 - robots.txt 853
 - Rogue Access Point 787, 792
 - Rogue DHCP-Server 641
 - Rolling Code 1118
 - ROMMON-Modus (Cisco) 379
 - root 101
 - Rooten (Android) 1079
 - Rootkit 420, 482
 - LKM-Rootkit 484
 - Userland-Rootkit 484
 - XCP 484
 - ZeroAccess 485
 - Root-Shell 343, 427
 - Routersploit 1126
 - rpcclient 303
 - RST Hijacking 675
 - Rsyslog 566
 - Rubber Ducky 776
- S**
- SafeSEH 1008
 - SafetyNet-Service (Android) 1079
 - Salt-Wert (Passwort-Hashes) 390
 - Samba 297
 - SAM-Datenbank 381
 - Sample (Malware) 507, 537
 - Sandbox 509, 535
 - Sandboxie 535
 - Sandcat Browser 850
 - Sanitizer 900
 - Sarbanes-Oxley Act (SOX) 1182
 - Scanning 218, 256
 - Scareware 454
 - Schutzklassen 50
 - Schutzziele 49
 - SCP 613
 - Scriptkiddie 41
 - Scrubbing Center 814
 - Searchbot 808
 - Seattle Lab Mail (SLmail) 981
 - Secure Shell (SSH) 613
 - Security Audit 1177
 - Security Autorun 525
 - Security Policy 538
 - SEH Overwrite Protection (SEHOP) 1008
 - Service-Manager 526
 - Service Set Identifier (SSID) 1023
 - Session Fixation-Angriff 699
 - Session Hijacking 667
 - Active Session Hijacking 669
 - Application Level Hijacking 668
 - Application Level Session Hijacking 675
 - Network Level Hijacking 668
 - Passive Session Hijacking 669
 - Session-ID 676
 - Session Replay-Angriff 699
 - Session Token 676
 - SFTP 613
 - shadow (Datei) 388
 - Shebang-Zeile 106
 - Sheep-Dipping 533
 - Shell 418
 - Shellcode 978, 1002
 - Shellshock 858
 - Shellter 480
 - Shodan 224, 1126
 - shred 579
 - Sicherheitsstufe 50
 - Sidejacking 668, 695
 - SIEM-System 340, 353, 584, 719, 918
 - sigverif.exe 530
 - SIM-Lock 1085
 - Skipfish 857
 - Skriptvirus 458
 - SlowHTTPTest 805
 - Slowloris 804
 - Smart Home 1106, 1108
 - SMB 297
 - SMiShing 1073
 - SMTP 316
 - Smurf Attack 799
 - Snagit 1192
 - Sniffing 593, 628
 - SNMP 303
 - Community-String 304

- MIB 304
 - OID 304
 - Trap 306
 - snmpwalk 311
 - Snort 722
 - Konfiguration 723
 - Regeln 723
 - SNscan 310
 - SOAP 840
 - Social Bot 808
 - Social Engineering 230, 747
 - CEO Fraud 755
 - Computer Based Social Engineering 758
 - Dumpster Diving 757
 - Eavesdropping 756
 - Fake Websites 752
 - Human Based Social Engineering 751, 754
 - Mobile Based Social Engineering 752
 - Pharming 758
 - Phishing 752, 758
 - Piggybacking 757
 - Reverse Social Engineering 752
 - Shoulder Surfing 756
 - Spear Phishing 759, 767
 - Tailgating 757
 - Technical Support Scam 755
 - Vishing 754
 - Whaling 760
 - Social-Engineer Toolkit (SET) 762
 - Social-Media-Footprinting 229
 - SOCKS 141
 - Clientkonfiguration 142
 - Dante 142
 - vicSOCK 145
 - Software
 - entfernen 125
 - installieren 124
 - Paketlisten aktualisieren 123
 - suchen 124
 - Update (Kali Linux) 123
 - Software as a Service (SaaS) 1143
 - Software Defined Radio (SDR) 786, 1119
 - Source Routing 675
 - Spam Mimic 546
 - Spear Phishing 759
 - Spoofing 669
 - SpyAgent 494
 - Spytech SpyAgent 494
 - Spyware 454, 491
 - SQL 889
 - SQL-Injection 889, 925
 - Blind SQL-Injection 939
 - Boolean SQL-Injection 945
 - Tautology based SQL-Injection 929
 - Time based SQL-Injection 946
 - SQLMap 948
 - SSH (Secure Shell) 139, 143
 - PuTTY 140
 - SSH-Server 117
 - TCP-Verbindungen tunneln 139
 - SSL 199
 - sslstrip 1059
 - SSL-VPN 199
 - Stack 979
 - Stack Buffer Overflow 977
 - Stack Canary (Stack Cookie) 1008
 - Stack Pointer (SP) 980
 - Stapel 979
 - Steganografie 543
 - Jargon Code 548
 - Least Significant Bits 550
 - Open Code 548
 - Semagramm 547
 - Steganalyse 556
 - Steganogramm 549
 - StegoStick 554
 - Stegosuite 552
 - Strings (Sysinternals) 505
 - Stuxnet 61
 - Sudo 107
 - sudo 391
 - Suicide Hacker 42
 - Supply-Chain-Angriff 498
 - Switch 594, 633
 - Symmetrische Algorithmen 169
 - Data Encryption Standard (DES) 170
 - Rivest Cipher 171
 - Serpent 172
 - Triple-DES (3DES oder DESede) 170
 - Twofish und Blowfish 171
 - und Rijndael) 170
 - SYN-Cookies 801
 - Syn-Flood-Angriff 800
 - Syslog 564
 - Syslog-ng 566
- ## T
- Tails (Linux-Distribution) 155
 - Task-Manager 512, 524
 - TCP 262
 - desynchronized state 674
 - Initial Sequence Number (ISN) 672
 - Receive Window 671
 - RST/Reopen 674
 - SACK 799
 - Session Splicing 730
 - Sliding Window 671
 - Window Size 671
 - tcpdump 618
 - TCP-Handshake 602
 - TeamViewer 1075
 - TeamWinRecoveryProject (TWRP) 1083
 - Teardrop-Angriff 804
 - Technischer Report 1192
 - Technitium MAC Address Changer 1050
 - Telnet 287, 611

Temporal Key Integrity Protocol (TKIP) 1027
 THC Hydra 409
 Throwing Star LAN Tap Pro 785
 Ticket Granting Server 384
 Ticket Granting Ticket 383
 Tier (Architektur) 832
 Timestamp 573
 TLS 200
 Tomcat 689
 Tor-Netzwerk 147

- DuckDuckGo 149
- Hidden Wiki 151
- Onion-Adressen 149
- Onion-Proxy 148
- Onion Services 149

 touch 582
 Tracking-Pixel 130
 Transparenter Proxy 132
 Transport Layer Security (TLS 200
 Treiber prüfen 528
 Tripwire 532
 Trojaner 452, 465

- Baukasten 470
- Botnet-Trojaner 466
- CLI-Trojaner 465
- Covert-Channel-Trojaner 467
- destruktive Trojaner 467
- E-Banking-Trojaner 467
- FTP-Trojaner 466
- HTTP/HTTPS-Trojaner 466
- ICMP-Tunneling-Trojaner 467
- Proxy-Server-Trojaner 466
- Remote-Access-Trojaner 466
- VNC-Trojaner 466

 TShark 621

U

Überwachungsrichtlinien (Windows) 562
 U-Boot (Bootloader) 1122
 Ubuntu Core 1108
 UDDI 840
 UDP 261
 UDP-Flood-Angriff 798
 UDP Hijacking 675
 UNC (Uniform Naming Convention) 297
 Uniform Resource Identifier (URI) 676
 Uniform Resource Locator (URL) 676, 833
 Universal Asynchronous Receiver Transmitter (UART)
 1121
 USB-Keylogger 775
 USBNinja 782
 USB-Sticks infizieren mit SET 767
 Use-after-free (Heap Overflow) 1006
 UserLand (App) 1086
 UTF-8 834

V

Veil-Framework 479
 VeraCrypt 172
 Verzeichnis 111
 VideoGhost 783
 Viren-Baukasten 470
 Virencheck 508
 VirtualBox 67, 68

- Gasterweiterungen 79
- Hostkey 79
- Netzwerk-Konfiguration 79
- Sicherungspunkt 80
- Snapshot 80

 Virtualisierung (Cloud) 1148
 Virtualisierungssoftware 68
 Virtual Local Area Networks 732
 Virtual Private Network (VPN) 137, 197

- IPsec 137
- IPsec-VPN 198
- OpenVPN 137
- Remote-Access-VPN 197
- Site-to-Site-VPN 197
- SSL-VPN 198
- VPN-Anbieter 138
- VPN-Gateway 137

 Virus 457
 Virus Maker 471
 VirusTotal 473
 Vishing 754
 VLAN Hopping 732
 VMware 67
 Vulnerability 336
 Vulnerability Assessment 256, 351, 1177
 Vulnerability-Scanner 339
 Vysor 1075

W

Wachstafel (Steganografie) 545
 WAFW00F 852
 WannaCry 206
 Wardriving 1024
 wash (WiFi-Scanning) 1047
 Watering-Hole-Angriff 761
 WayBack Machine 223
 WDS (Wireless Distribution Set) 1022
 Wearables 1106
 Web Application Firewall (WAF) 852
 Web Bug 130
 Webcrawler 808
 WebDAV 841
 Web-Hacking 831
 Web Security Dojo 878
 Webserver 832, 843
 Webshell 969
 Website-Footprinting 239
 Web Spider (Web Crawler) 853
 Web Vulnerability Scanner (WVS) 856

WebWolf 875
weevely 969
WEP (Wired Equivalent Privacy Protocol) 1026
wevtutil.exe 571
Whaling 760
White-Box-Test 1178
White Hat 41
White Hat Hacking 1176
Whois 231
Wi-Fi Alliance 1020
WiFiKill 1088
wifiphisher 1061
WiFi Pineapple 787, 1060
Win32DiskImager 790
Windows 10 81
Windows 11 81
Windows 7 81
Wine 478
WinPcap 595
Wireless Access Point (AP) 1020
WirelessKeyView 1052
Wireless LAN (WLAN) 1017
 Frequenzen 1018
 Honeypot 787
 Phishing 1060
 Sniffing 634
Wireshark 518, 593
 Anzeigefiltern 606
 Capture Filter 601
 Display Filter 601, 606
 Ncap 595
 Pcap 595
Wiretapping 628
WordPress 861
Wörterbuch-Angriffe 393
Wortlisten (Passwort-Hacking) 394

WPA2 1027
WPA (Wi-Fi Protected Access) 1027
WPA/WPA2-Angriff 1043
WPS (Angriff) 1046
WPS (Wi-Fi Protected Setup) 1028
WPScan 866
Wrapper 468
WS-* 840
WSDL 840
wtmtp 580
Wurm 453, 459

X

XAMPP 845
XEN 68
Xfce 94
XML 839
XML-Entity 906
XSRF 899
XSS 892
XXE (XML External Entities) 906

Z

zAnti 1088
Zeitstempel 573
Zeitzone einstellen 97
Zenmap 277
Zephyr 1108
Zero-Day-Exploit 354
ZigBee 1111, 1124
Zombie (Botnetze) 809
Z-Shell 104
Zwei-Faktor-Authentifizierung (2FA) 366
Zwiebel-Routing (Tor) 147